arXiv:2601.12066v4 [cs.CV] 26 Jun 2026

Learning Stochastic Bridges for Video Object Removal via Video-to-Video Translation

Zijie Lou [*1] **Xiangwei Feng** [*1] **Jiaxin Wang** [1 2] **Jiangtao Yao** [1] **Fei Che** [1] **Tianbao Liu** [1] **Chengjing Wu** [1] **Xiaochao Qu** [1] **Luoqi Liu** [1] **Ting Liu** [1]

Abstract

Existing video object removal methods predominantly rely on diffusion models following a noise-to-data paradigm, where generation starts from uninformative Gaussian noise. This approach discards the rich structural and contextual priors present in the original input video. Consequently, such methods often lack sufficient guidance, leading to incomplete object erasure or the synthesis of implausible content that conflicts with the scene's physical logic. In this paper, we reformulate video object removal as a video-to-video translation task via a stochastic bridge model. Unlike noise-initialized methods, our framework establishes a direct stochastic path from the source video (with objects) to the target video (objects removed). This bridge formulation effectively leverages the input video as a strong structural prior, guiding the model to perform precise removal while ensuring that the filled regions are logically consistent with the surrounding environment. To address the trade-off where strong bridge priors hinder the removal of large objects, we propose a novel adaptive mask modulation strategy. This mechanism dynamically modulates input embeddings based on mask characteristics, balancing background fidelity with generative flexibility. Extensive experiments demonstrate that our approach significantly outperforms existing methods in both visual quality and temporal consistency. The project page is https://bridgeremoval.github.io/.

[*]Equal contribution [1]MT Lab, Meitu Inc., Beijing 100083, China [2]Beijing Jiaotong University, Beijing 100044, China. Correspondence to: Ting Liu <lt@meitu.com>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

1. Introduction

Video object removal (Kim et al., 2019; Li et al., 2020; Zou et al., 2021; Zhang et al., 2022; Zhou et al., 2023b; Ju et al., 2024; Li et al., 2025b) aims to erase specific objects from a video sequence while plausibly synthesizing the missing regions based on the surrounding context. The core challenge lies in reconstructing occluded areas in a manner that ensures spatial coherence within individual frames and temporal consistency across the entire sequence.

Prior works on this task have generally followed two primary directions: propagation-based methods (Kim et al., 2019; Li et al., 2020; Zou et al., 2021; Zhou et al., 2023b; Zhang et al., 2022) and generative inpainting methods (Ju et al., 2024; Li et al., 2025b; Miao et al., 2025). Propagation-based techniques typically rely on optical flow to transfer pixels from non-occluded frames to fill the missing regions. While effective for static scenes, these methods often struggle in complex scenarios involving dynamic camera movements or prolonged occlusions where valid source pixels are unavailable. Conversely, generative approaches formulate object removal as a synthesis problem. The recent emergence of large-scale diffusion models (Podell et al., 2023; Esser et al., 2024; Labs, 2024; Li et al., 2025b) has significantly advanced this domain, enabling the generation of high-fidelity textures and complex structures that were previously unattainable with traditional methods.

Despite these advancements, state-of-the-art video inpainting methods (Li et al., 2025b; Jiang et al., 2025; Miao et al., 2025), particularly those based on diffusion models, predominantly operate under a noise-to-data paradigm. In this standard framework, the generation process initiates from uninformative Gaussian noise and is tasked with denoising this stochastic state into a coherent video sequence conditioned on the masked input. We argue that this formulation is suboptimal for object removal task. Initiating generation from pure noise requires the model to synthesize the entire video content from scratch, often leading to leading to incomplete object erasure or the synthesis of implausible content that conflicts with the scene's physical logic To address these stability issues, a common strategy involves reference-guided generation, where a specialized image in-

painting model processes a keyframe (usually the first frame) that subsequently serves as a guidance signal (Ouyang et al., 2024; Gao et al., 2025). However, this approach introduces significant drawbacks. It increases system complexity by necessitating an auxiliary image edit model and creates a dependency bottleneck where artifacts in the initial frame propagate through the sequence, potentially conflicting with the natural temporal evolution of the background.

In this work, inspired by recently proposed bridge models (Chen et al., 2021; Liu et al., 2023; Chen et al., 2023; Wang et al., 2025), we propose a paradigm shift by reformulating video object removal as a video-to-video translation task. We posit that the input video, even containing the unwanted object, holds a vast amount of valid structural and environmental information that should be leveraged as a strong prior rather than discarded in favor of Gaussian noise. To realize this, we introduce BridgeRemoval, a novel framework based on the stochastic bridge formulation (Tong et al., 2024; Zhou et al., 2023a; Chen et al., 2023) . Unlike standard diffusion-based methods that traverse a path from noise to data, our approach constructs a direct stochastic bridge between the source video distribution (the input video containing the object) and the target video distribution (the clean video with the object removed). By compressing videos into continuous latent representations via a Variational Autoencoder (VAE), we establish a tractable SDE-based generation process where the input latent representation serves directly as the prior boundary distribution.

This bridge formulation anchors the generative trajectory to the source video, naturally enforcing strong spatio-temporal coherence. The model learns a transformation that selectively modifies the object region while preserving the integrity of the unmasked context. Nevertheless, a direct application of this strong prior presents a trade-off. While excellent for background preservation, an overly rigid adherence to the source video can hinder the removal of large objects, as the model may struggle to deviate sufficiently from the input to synthesize large missing regions. To mitigate this, we propose a novel adaptive mask modulation (AMM) strategy. This mechanism dynamically adjusts the influence of the input embeddings based on the spatial characteristics of the mask, allowing the model to relax structural constraints in regions with large occlusions while maintaining strict fidelity in regions with valid backgrounds.

In summary, our contributions are as follows:

- We propose BridgeRemoval, a novel video-to-video generative framework that utilizes an SDE-based bridge model to reformulate video object removal. By establishing a direct stochastic path from the source video to the target, our method effectively leverages input priors to achieve better object removal.
- We introduce an adaptive mask modulation strategy that dynamically balances the trade-off between background fidelity and generative flexibility, ensuring robust performance across objects of varying scales.
- To provide a comprehensive evaluation of video object removal models, we propose BridgeRemoval-Bench, which encompasses a wide variety of scenarios and provides meticulously annotated masks.
- Extensive experiments demonstrate the effectiveness of our approach in diverse real-world scenarios, outperforming existing methods in both visual quality and temporal coherence, while also highlighting the potential of our framework to be extended to other video editing tasks.

2. Related Works

Diffusion-based Video Object Removal Diffusion-based methods (Li et al., 2025b; Zi et al., 2025a; Miao et al., 2025; Jiang et al., 2025; Lee et al., 2025) have achieved state-of-the-art performance in video object removal. DiffuEraser (Li et al., 2025b) adapts the image inpainting architecture of BrushNet (Ju et al., 2024) to the video domain via a progressive two-stage training strategy involving prior propagation and temporal refinement. Beyond simple occlusion handling, ROSE (Miao et al., 2025) addresses complex environmental interactions by leveraging a synthetic dataset with comprehensive supervision to simultaneously remove objects and their collateral visual effects, such as shadows and reflections. Furthermore, unified editing frameworks have also shown promise in this area. VACE (Jiang et al., 2025) proposes a universal backbone for diverse visual conditions, while GenOmnimatte (Lee et al., 2025) employs layered video decomposition, both of which support object removal as a downstream application.

Bridge Models Recently, bridge models (Chen et al., 2021; Tong et al., 2024; Liu et al., 2023; Zhou et al., 2023a; Chen et al., 2023; Zheng et al., 2025; He et al., 2024; De Bortoli et al., 2021; Peluchetti, 2023) have garnered significant attention for their ability to transcend the limitations of the Gaussian prior inherent in standard diffusion models. By establishing a direct stochastic path between two arbitrary distributions, bridge models have demonstrated the clear superiority of a data-to-data generation paradigm over traditional noise-to-data approaches, particularly in tasks such as image-to-image translation (Liu et al., 2023; Zhou et al., 2023a) and speech synthesis (Chen et al., 2023; Li et al., 2025a). Notably, FrameBridge (Wang et al., 2025) successfully extended this methodology to image-to-video (I2V) synthesis by introducing SNR-aligned fine-tuning to stabilize the learning process.

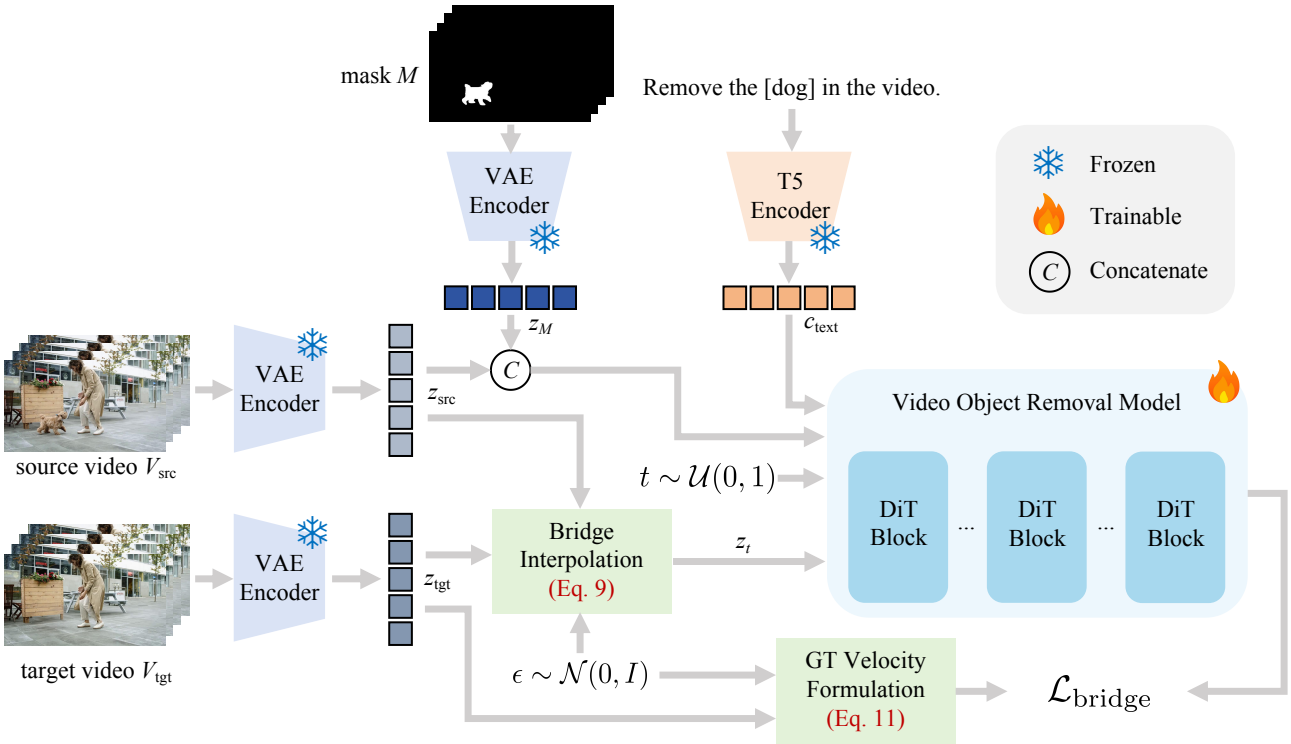

Figure 1. The framework of BridgeRemoval. The video inputs are projected into latent space using a frozen VAE encoder. Unlike standard diffusion, we employ a VP-SDE Bridge formulation to interpolate a trajectory (z_t) directly from the source video prior (z_{src}) to the clean target (z_{tgt}). The DiT-based model is conditioned on text embeddings (c_{text}) and a spatial input formed by concatenating the mask (z_{M}) with the source latent (z_{src}), optimized via a velocity-matching objective ($\mathcal{L}_{\text{bridge}}$).

3. Preliminaries

Probability path modeling (De Bortoli et al., 2021; Lipman et al., 2022; Liu et al., 2022) defines a class of generative models that describe continuous-time processes transporting mass from a prior distribution p_1 (source) to a target distribution p_0 (target). Generally, these models are governed by a stochastic differential equation (SDE) (Song et al., 2021):

$$\mathrm{d}X_t = v(X_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}W_t, \quad t \in [0, 1], \tag{1}$$

where $v(X_t, t)$ is the velocity field driving the deterministic trajectory, $g(t)$ controls the stochastic diffusion, and W_t is a standard Brownian motion.

3.1. Brownian Bridge

The standard Brownian Bridge (Albergo et al., 2025; Li et al., 2023) serves as a foundational instance of such processes, conditioned to start at a specific point x_1 and end at x_0. Unlike standard diffusion which maps data to Gaussian noise, a Brownian Bridge interpolates between two data points. Assuming a constant diffusion coefficient, the intermediate state X_t follows a simple linear interpolation perturbed by noise:

$$X_t = (1-t)x_0 + tx_1 + \sqrt{t(1-t)}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{2}$$

While intuitive, this symmetric variance schedule $t(1-t)$ is rigid. It lacks the flexibility to handle complex high-dimensional data like video, where the signal-to-noise ratio (SNR) needs to be carefully scheduled to ensure stable training and generation.

3.2. VP-SDE Bridge

To overcome the limitations of the standard Brownian Bridge, we adopt a Variance-Preserving (VP) formulation inspired by the VP-SDE typically used in score-based generative models (Song et al., 2021). This formulation generalizes the bridge process by introducing time-dependent coefficients derived from a flexible noise schedule $\beta(t) \in [\beta_{\min}, \beta_{\max}]$.

Unlike the heuristic linear interpolation, the VP-SDE Bridge is theoretically grounded. As derived in Appendix B.1, the marginal distribution of this process remains Gaussian, allowing us to sample the intermediate state X_t directly via a reparameterized ternary interpolation:

$$X_t = a_t x_0 + b_t x_1 + c_t \epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{3}$$

Crucially, the coefficients a_t, b_t, c_t are not arbitrary but are strictly determined by the cumulative variance schedule σ_t

of the VP-SDE:

$$a_t = \frac{\bar{\sigma}_t^2}{\sigma_1^2}, \quad b_t = \frac{\sigma_t^2}{\sigma_1^2}, \quad c_t = \frac{\bar{\sigma}_t \sigma_t}{\sigma_1}, \tag{4}$$

where σ_1^2 is the total variance at $t = 1$, and $\bar{\sigma}_t^2 = \sigma_1^2 - \sigma_t^2$. This rigorous formulation guarantees that boundary conditions $X_0 = x_0$ and $X_1 = x_1$ are satisfied, while providing a tunable trajectory well-suited for high-fidelity video generation.

4. BridgeRemoval

As shown in Figure 1, we propose BridgeRemoval, a novel framework for video object removal. Unlike standard diffusion models that generate content from a generic Gaussian prior, our method leverages the VP-SDE Bridge formulation to construct a direct probability path from the source video (with objects) to the target video (object removed). This approach ensures higher fidelity and better preservation of non-masked regions.

4.1. Latent Encoding and Dual-Role Conditioning

Given a training triplet consisting of a source video V_{src}, a target video V_{tgt}, and a binary mask M, we first project the visual data into a compressed latent space using a pre-trained VAE encoder $\mathcal{E}$ (Wan et al., 2025):

$$z_{\text{src}} = \mathcal{E}(V_{\text{src}}), \quad z_{\text{tgt}} = \mathcal{E}(V_{\text{tgt}}), \quad z_M = \mathcal{E}(M). \tag{5}$$

Simultaneously, the text prompt P is encoded into embeddings c_{text} via a T5 encoder (Raffel et al., 2020).

To provide precise spatial guidance for inpainting, we construct a conditional input y by concatenating the encoded mask with the source latent:

$$y = \text{Concat}(z_M, z_{\text{src}}). \tag{6}$$

It is crucial to note that z_{src} plays a dual role in our framework. First, it acts as the *prior distribution* (the starting point of the bridge at $t = 1$). Second, it serves as a *spatial condition* in y, providing the network with contextual information about the background that should be preserved. This dual utilization effectively anchors the generation process, ensuring seamless blending between the inpainted region and the original background.

4.2. Velocity-Driven Bridge Training

To learn the transition trajectory between the source and target distributions, we frame the video removal task as a stochastic process governed by a Variance-Preserving Stochastic Differential Equation (VP-SDE) (Song et al., 2021).

Bridge Process. Standard diffusion models typically assume an uninformative Gaussian prior $\mathcal{N}(0, I)$. However, our task provides the source video z_{src} as a strong structural prior. To leverage this, we adopt the Denoising Diffusion Bridge Model (DDBM) (Zhou et al., 2023a), essentially replacing the Gaussian prior with a Dirac prior $\delta_{z_{\text{src}}}$.

Specifically, the bridge process is defined by a forward SDE that drifts towards the source video z_{src}:

$$\begin{aligned} \mathrm{d}z_t = &\left[f(t)z_t + g(t)^2 h(z_t, t, z_{\text{src}})\right] \mathrm{d}t \\ &+ g(t)\mathrm{d}w, \quad z_0 \sim p_{\text{tgt}},\ z_T = z_{\text{src}}, \end{aligned} \tag{7}$$

where $f(t)$ and $g(t)$ are the drift and diffusion coefficients of the VP-SDE. The term $h(z_t, t, z_{\text{src}}) \triangleq \nabla_{z_t} \log p(z_{\text{src}}|z_t)$ acts as a guidance force, pulling the trajectory towards z_{src}.

Correspondingly, the generative reverse process is governed by:

$$\begin{aligned} \mathrm{d}z_t = &[f(t)z_t - g(t)^2(s_\theta(z_t, t, z_{\text{src}}) \\ &- h(z_t, t, z_{\text{src}}))]\mathrm{d}t + g(t)\mathrm{d}\bar{w}, \end{aligned} \tag{8}$$

where $s_\theta(z_t, t, z_{\text{src}})$ approximates the score function. Since the transition kernel remains Gaussian, the term h is analytically tractable. We provide the proof of the Gaussian nature of the bridge marginal (see Appendix B.1) and the derivation of h (see Appendix B.2).

Training Objective. Instead of simulating stochastic paths directly, we leverage the analytic property of the bridge marginal distribution. As derived in Appendix B.1, the intermediate state z_t can be sampled directly via a weighted ternary interpolation:

$$z_t = a_t z_{\text{tgt}} + b_t z_{\text{src}} + c_t \epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{9}$$

Here, z_t is a mixture of the clean target z_{tgt}, the source prior z_{src}, and noise ϵ. The scalar coefficients a_t, b_t, c_t are rigorously derived from the VP-SDE variance schedule σ_t (see Appendix B.1):

$$a_t = \frac{\bar{\sigma}_t^2}{\sigma_1^2}, \quad b_t = \frac{\sigma_t^2}{\sigma_1^2}, \quad c_t = \frac{\bar{\sigma}_t \sigma_t}{\sigma_1}. \tag{10}$$

These coefficients ensure the boundary conditions $z_0 = z_{\text{tgt}}$ and $z_1 = z_{\text{src}}$.

To train the network v_θ parameterized by θ, we define a ground-truth velocity target u_t. As shown in Appendix B.3, minimizing the velocity matching error is equivalent to score matching:

$$u_t = \frac{a_t}{\rho_t}\epsilon - \frac{c_t}{\rho_t} z_{\text{tgt}}, \quad \text{where } \rho_t = \sqrt{a_t^2 + c_t^2}. \tag{11}$$

The network is optimized by minimizing the mean squared error:

$$\mathcal{L}_{\text{bridge}} = \mathbb{E}_{t, \epsilon, z_{\text{src}}, z_{\text{tgt}}} \left[\| v_\theta(z_t, t, y, c_{\text{text}}) - u_t \|^2 \right]. \tag{12}$$

The complete training procedure is detailed in Algorithm 1.

Algorithm 1 Training

Input: source video V_{src}, target video V_{tgt}, mask M, text prompt P, VAE encoder $\mathcal{E}$, T5 encoder, model v_θ, variance schedule σ_t

repeat
- Encode latents $z_{\text{src}} \leftarrow \mathcal{E}(V_{\text{src}})$, $z_{\text{tgt}} \leftarrow \mathcal{E}(V_{\text{tgt}})$, and $z_M \leftarrow \mathcal{E}(M)$
- Encode text embeddings $c_{\text{text}} \leftarrow \text{T5}(P)$
- Construct conditional input $y \leftarrow \text{Concat}(z_M, z_{\text{src}})$
- Sample interpolation time $t \sim \mathcal{U}(0,1)$ and noise $\epsilon \sim \mathcal{N}(0, I)$
- Compute SDE coefficients a_t, b_t, c_t based on the variance schedule σ_t
- Construct intermediate bridge state $z_t = a_t z_{\text{tgt}} + b_t z_{\text{src}} + c_t \epsilon$
- Compute normalization factor $\rho_t = \sqrt{a_t^2 + c_t^2}$
- Compute velocity target $u_t = \frac{a_t}{\rho_t}\epsilon - \frac{c_t}{\rho_t} z_{\text{tgt}}$
- Update parameters θ by gradient descent on $\|v_\theta(z_t, t, y, c_{\text{text}}) - u_t\|^2$

until *convergence*;

Algorithm 2 Inference

Input: source video V_{src}, mask M, text prompt P, trained model v_θ, steps N, variance schedule σ_t, T5 encoder, VAE encoder $\mathcal{E}$, VAE decoder $\mathcal{D}$

Encode input latents $z_{\text{src}} \leftarrow \mathcal{E}(V_{\text{src}})$ and condition $y \leftarrow \text{Concat}(\mathcal{E}(M), z_{\text{src}})$
Encode text embeddings $c_{\text{text}} \leftarrow \text{T5}(P)$
Initialize state $z \leftarrow z_{\text{src}}$ using the source video as the prior
Define time schedule $\{t_k\}_{k=N}^{0}$ from 1 to 0 **for** $k = N, \ldots, 1$ **do**
- Set current time $t \leftarrow t_k$ and next time $t' \leftarrow t_{k-1}$
- Predict velocity field $\hat{v}_t \leftarrow v_\theta(z, t, y, c_{\text{text}})$
- Recover predicted target $\hat{z}_{0|t}$ via Eq. 13
- **if** $k > 1$ **then**
  - Compute SDE solver weights w_1, w_2, w_3 based on σ_t
  - Sample noise $\epsilon' \sim \mathcal{N}(0, I)$
  - Update state $z \leftarrow w_1 z + w_2 \hat{z}_{0|t} + w_3 \epsilon'$
- **else**
  - Set final state $z \leftarrow \hat{z}_{0|t}$
- **end**

end
Decode output video $V_{\text{out}} \leftarrow \mathcal{D}(z)$

Output: Inpainted video V_{out}

4.3. Generative Inference via SDE Solver

During inference, we aim to reverse the bridge process to traverse from the source latent z_{src} back to the clean target z_{tgt}. Theoretically, this corresponds to solving the backward SDE (Eq. 8) derived in Sec. 4.2. Since the continuous backward SDE involves the unknown score function s_θ, we employ a numerical SDE solver to approximate the trajectory. We adopt a variance-corrected sampling strategy that acts as a discretization of the backward SDE, ensuring the generation strictly follows the statistics of the Brownian bridge.

Given the current state z_t at timestep t, the network predicts the velocity field $\hat{v}_t = v_\theta(z_t, t, y)$, which approximates the drift term of the reverse process. We first estimate the clean target latent $\hat{z}_{0|t}$ by inverting the velocity equation (Eq. 11), utilizing the normalization factor $\rho_t = \sqrt{a_t^2 + c_t^2}$ defined previously:

$$\hat{z}_{0|t} = \frac{1}{\rho_t^2 / c_t} \left(\frac{a_t}{c_t} z_t - \frac{a_t b_t}{c_t} z_{\text{src}} - \rho_t \hat{v}_t \right). \tag{13}$$

This formula explicitly removes the predicted noise and the weighted source component from z_t, recovering an approximation of the target z_{tgt}. To progress to the next timestep $t' < t$, we perform an SDE solver step that combines the current state z_t, the predicted target $\hat{z}_{0|t}$, and Gaussian noise injection:

$$z_{t'} = w_1 z_t + w_2 \hat{z}_{0|t} + w_3 \epsilon', \quad \epsilon' \sim \mathcal{N}(0, I). \tag{14}$$

Here, w_1, w_2, w_3 are scalar weights analytically determined by the bridge schedule to ensure the trajectory maintains the correct conditional variance. Specifically, w_3 modulates the contribution of the freshly sampled noise ϵ' to prevent variance collapse near the target. The detailed inference algorithm is summarized in Algorithm 2.

4.4. Adaptive Mask Modulation

While the bridge formulation effectively anchors the generation trajectory to the source video, this rigid adherence to the source distribution can become detrimental when removing large objects. To mitigate this limitation, we propose Adaptive Mask Modulation (AMM), a spatially-aware feature modulation mechanism inspired by FiLM (Perez et al., 2018). The mask $M \in \mathbb{R}^{C_m \times F \times H \times W}$ is then projected into a latent feature space through a learnable patch embedding:

$$F_M = \phi_M(M) \in \mathbb{R}^{D \times F' \times H' \times W'} \tag{15}$$

where ϕ_M is a 3D convolutional layer. From the mask features F_M, we predict spatially-varying modulation parameters through two $1 \times 1 \times 1$ convolutional projections:

$$\gamma = f_\gamma(F_M), \quad \beta = f_\beta(F_M) \tag{16}$$

where $\gamma, \beta \in \mathbb{R}^{D \times F' \times H' \times W'}$ represent the scale and shift parameters, respectively.

Let $h \in \mathbb{R}^{D \times F' \times H' \times W'}$ denote the input patch embeddings after the initial projection. The modulated embeddings h' are computed as:

$$h' = h \odot (1 + \gamma) + \beta \tag{17}$$

Scale modulation $(1+\gamma)$ adaptively amplifies or attenuates features based on their spatial relationship to the mask, enabling the model to learn where to preserve original information. Shift modulation (β) injects position-aware guidance into the feature, providing explicit spatial cues about the inpainting region boundaries and structure.

By incorporating rich spatial mask information through learnable modulation, AMM enables the model to adaptively adjust its behavior across different spatial regions, effectively counterbalancing the strong source prior within masked regions while maintaining strict fidelity to the surrounding environment.

5. Experiments

5.1. Experimental Setup

Training Data. We construct a training dataset totaling approximately 47,000 video pairs from two complementary sources. First, we utilize the ROSE dataset (Miao et al., 2025), which consists of 16,000 synthetically generated samples. By leveraging 3D engines to toggle object visibility, ROSE provides perfectly aligned triplets of source videos, target backgrounds, and masks. While this offers high-quality supervision, our statistical analysis indicates a distribution bias towards static, rigid objects, limiting the model's generalization to complex motion. To address this limitation, we introduce a supplementary real-world composite dataset comprising 31,000 video pairs. This subset is explicitly designed to improve robustness on dynamic agents, containing 15,000 human-centric videos and 16,000 videos of common dynamic objects (e.g., animals, vehicles). The data pipeline adopts a composition-based approach: we employ SAM2 (Ravi et al., 2025) to extract high-fidelity foregrounds from real-world videos and composite them onto clean backgrounds. A Qwen3-VL (Bai et al., 2025b) based filtering mechanism is then applied to ensure visual coherence and discard low-quality samples.

Testing Data. Existing video object removal benchmarks suffer from limited scale and insufficient scene diversity. For instance, DAVIS (Perazzi et al., 2016) contains only 50 videos, all in landscape orientation, with some challenging cases involving fast motion. ROSE-Bench (Miao et al., 2025) includes 60 videos, but most depict static scenes, and its publicly released dataset does not specify a test split, making it unsuitable for standardized evaluation.

To enable comprehensive assessment of video object removal models, we introduce BridgeRemoval-Bench, a new benchmark comprising 150 high-quality videos with rich scene variety, including both landscape and portrait orientations, as well as annotations for both single and multiple objects. The dataset is constructed from three distinct sources:

- 50 real-world videos collected from stock platforms such as Pexels (pex), featuring aesthetically pleasing visuals and clean backgrounds.
- 50 synthetic videos generated by first using Qwen3 (Yang et al., 2025a) to produce diverse text prompts and then synthesizing videos with Veo (veo), covering common categories including humans, animals, and vehicles.
- 50 in-the-wild videos sourced from social media platforms like TikTok (tik), exhibiting highly complex and challenging scenarios.

For all 150 videos, object masks are generated using SAM2 (Ravi et al., 2025). Furthermore, to support methods such as VACE (Jiang et al., 2025) that require a textual description of the target video (object removed), we employ Qwen3-VL (Bai et al., 2025b) to generate a detailed caption for each video (object removed), and performed the same process on the DAVIS dataset. In summary, the DAVIS dataset and BridgeRemoval-Bench are used to evaluate the performance of various algorithms.

Metrics. To comprehensively evaluate the performance of each model, we employ the following metrics:

- CLIP-T (Bai et al., 2025a) is used to calculate the similarity between the result video and the text description to verify whether the object has been removed. And CLIP-F (Bai et al., 2025a) calculates the average inter-frame CLIP similarity to gauge temporal consistency.
- Following prior works (Bian et al., 2025), we evaluate the fidelity of the background using standard metrics, including PSNR, MSE, and LPIPS (Zhang et al., 2018). These metrics are computed exclusively on the unmasked areas.
- We also report results from VBench (Huang et al., 2024). However, as these metrics from VBench exhibit limited discriminative power in our specific task and may not fully reflect the nuanced performance of the algorithms, we provide these results in the Appendix D for reference only.

Implementation Details. Our model is initialized from the pre-trained Wan2.1-1.3B (Wan et al., 2025). During training, we utilize video clips consisting of 81 frames. To accommodate varying aspect ratios, we employ a bucket-based multi-resolution strategy, where videos are resized to resolutions such as 576×1024, 1024×576, or 768×768 based on their original orientation. The bridge process is parameterized with 1000 discrete timesteps during training, utilizing a linear variance schedule with $\beta_{\min} = 0.01$ and $\beta_{\max} = 50.0$. We employ the AdamW optimizer (Loshchilov & Hutter,

Table 1. Quantitative comparisons among BridgeRemoval and other video object removal models in DAVIS dataset. **Red** stands for the best, **Blue** stands for the second best.

Method	Year-Venue	Base Model	Video Quality		Unmasked Region Preservation			Runtime (s/frame) ↓
			CLIP-T ↑	CLIP-F ↑	PSNR ↑	MSE ↓	LPIPS ↓	
Senoria (Zi et al., 2025b)	2025-arxiv	CogVideoX-5B (Yang et al., 2025b)	0.2618	0.9654	17.4752	2057.1134	0.4654	2.143
Ditto (Bai et al., 2025a)	2025-arxiv	Wan2.1-14B (Wan et al., 2025)	0.2196	**0.9712**	11.7304	4945.1432	0.5226	6.402
ICVE (Liao et al., 2025)	2025-arxiv	Hunyuanvideo-13B (Kong et al., 2024)	0.2361	0.9670	25.9675	244.8118	0.1116	12.571
ROSE (Miao et al., 2025)	2025-NeurIPS	Wan2.1-1.3B (Wan et al., 2025)	0.2353	0.9676	26.1202	204.0729	0.1046	**0.741**
VACE (Jiang et al., 2025)	2025-ICCV	Wan2.1-1.3B (Wan et al., 2025)	0.2629	0.9654	**27.3758**	**138.2393**	**0.0796**	1.765
GenOmnimatte (Lee et al., 2025)	2025-CVPR	Wan2.1-1.3B (Wan et al., 2025)	**0.2814**	0.9701	26.6812	177.3141	0.125	3.914
BridgeRemoval (Ours)	2026	Wan2.1-1.3B (Wan et al., 2025)	**0.2788**	**0.9768**	**28.2154**	**133.3605**	**0.0977**	**1.111**

Table 2. Quantitative comparisons among BridgeRemoval and other video object removal models in BridgeRemoval-Bench. **Red** stands for the best, **Blue** stands for the second best.

Method	Year-Venue	Base Model	Video Quality		Unmasked Region Preservation			Runtime (s/frame) ↓
			CLIP-T ↑	CLIP-F ↑	PSNR ↑	MSE ↓	LPIPS ↓	
Senoria (Zi et al., 2025b)	2025-arxiv	CogVideoX-5B (Yang et al., 2025b)	0.2920	0.9843	18.9123	1553.9859	0.3365	2.143
Ditto (Bai et al., 2025a)	2025-arxiv	Wan2.1-14B (Wan et al., 2025)	0.2218	0.9873	10.5340	7523.6043	0.5262	6.402
ICVE (Liao et al., 2025)	2025-arxiv	Hunyuanvideo-13B (Kong et al., 2024)	0.2842	0.9887	17.4098	1589.3646	0.2740	12.571
ROSE (Miao et al., 2025)	2025-NeurIPS	Wan2.1-1.3B (Wan et al., 2025)	0.2219	**0.9879**	30.5175	84.6950	0.0470	**0.741**
VACE (Jiang et al., 2025)	2025-ICCV	Wan2.1-1.3B (Wan et al., 2025)	0.2875	0.9859	31.0548	**61.7623**	**0.0335**	1.765
GenOmnimatte (Lee et al., 2025)	2025-CVPR	Wan2.1-1.3B (Wan et al., 2025)	**0.2966**	0.9878	**31.5999**	63.3032	0.0506	3.914
BridgeRemoval (Ours)	2026	Wan2.1-1.3B (Wan et al., 2025)	**0.2922**	**0.9917**	**33.2394**	**46.3525**	**0.0415**	**1.111**

2019) with a constant learning rate of 2×10^{-5} and train the model for 20000 steps on 8 NVIDIA H100 GPUs. For inference, we use a fast sampling strategy with 50 steps to ensure efficient generation.

5.2. Quantitative Evaluation

We compared BridgeRemoval with six state-of-the-art methods including Senoria (Zi et al., 2025b), Ditto (Bai et al., 2025a), ICVE (Liao et al., 2025), ROSE (Miao et al., 2025), VACE (Jiang et al., 2025), and GenOmnimatte (Lee et al., 2025) on both DAVIS (Perazzi et al., 2016) and BridgeRemoval-Bench. CLIP-T and CLIP-F measure the model's capability to remove the target object, while PSNR, MSE, and LPIPS evaluate its ability to preserve the background content. A high-performing video object removal method must excel in both aspects. Thanks to the stochastic bridge formulation, BridgeRemoval achieves strong performance in both object removal and background preservation.

As shown in Table 1 and 2, our method attains the best or second-best results across all quantitative metrics. Notably, on BridgeRemoval-Bench, our approach outperforms the current state-of-the-art method by 1.64 dB in PSNR. In contrast, other methods exhibit a clear trade-off. Some successfully remove the object but fail to preserve the background, while others maintain the background well but inadequately remove the target. For instance, VACE (Jiang et al., 2025) tends to generate a new object within the masked region rather than seamlessly inpainting the background. Consequently, it achieves relatively high scores in PSNR, MSE, and LPIPS but performs poorly in CLIP-T and CLIP-F, indicating ineffective object removal. In addition, we compared the inference speed of different methods. All methods are evaluated on L20X GPUs, our method runs at 1.111 seconds per frame, striking a favorable balance between generation quality and inference efficiency.

5.3. Qualitative Comparison

For the visual comparison, we compare our method with ROSE (Miao et al., 2025), VACE (Jiang et al., 2025), and GenOmnimatte (Lee et al., 2025), which are representative methods of diffusion-based approaches. Figure 2 presents four comparison results for video object removal. Our method starts from the source video rather than random noise and leverages a strong prior to effectively fill the masked regions. It completes masked regions with coherence and clear contents, while other compared methods tend to fail or produce unpleasant inpainting results such as texture distortions and black hazy region in ROSE results, as well as artifacts in VACE and GenOmnimatte. More samples can be visited at: `https://bridgeremoval.github.io/`.

5.4. Ablation Study

We conduct comprehensive ablation studies to validate the design choices of our framework. Detailed results are provided in Appendix C.

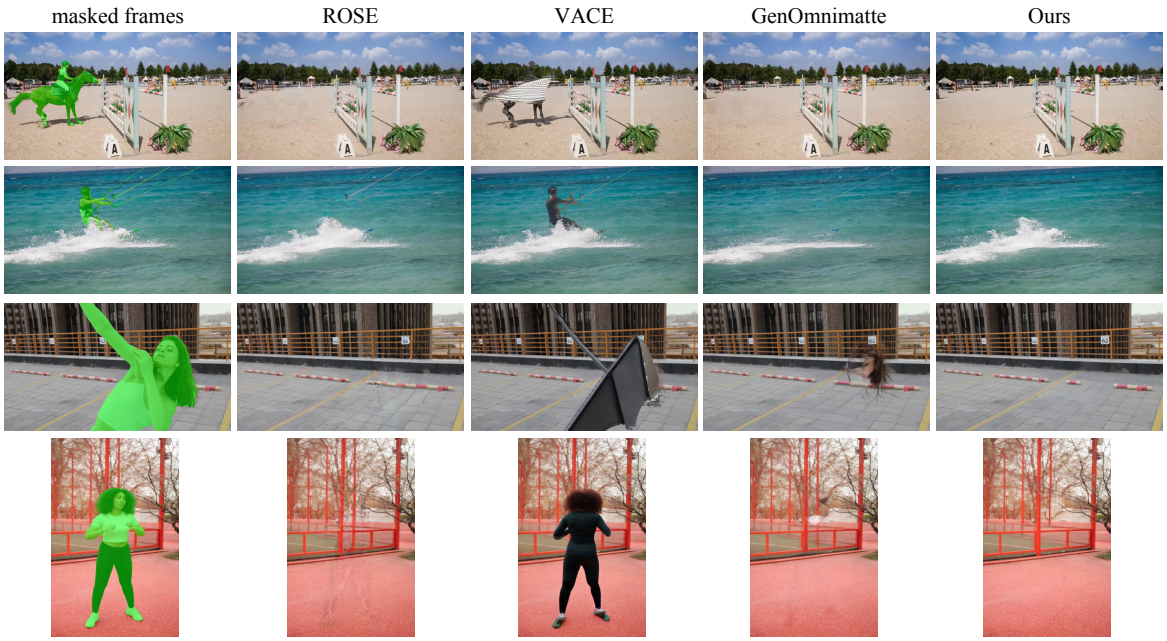

Figure 2. Qualitative comparisons. The input videos in the first two rows are from DAVIS dataset, and those in the last two rows are from BridgeRemoval-Bench.

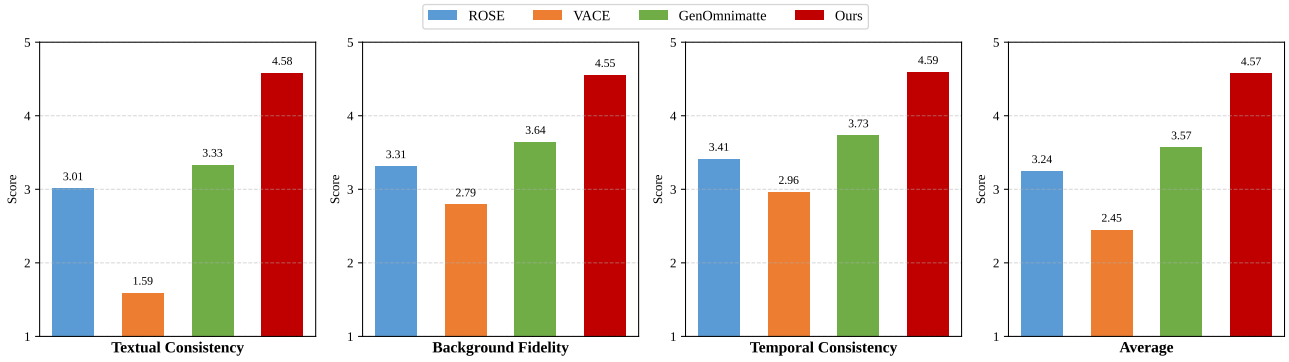

Figure 3. User study. Users rated BridgeRemoval and other video object removal methods based on three criteria: text consistency, background fidelity, and temporal consistency.

5.5. User Study

We conducted a user study involving 24 participants, including researchers and students specializing in image and video editing, to evaluate randomly selected cases. The assessment focuses on three dimensions: Textual Consistency (whether the target is effectively removed), Background Fidelity (how well the background is preserved), and Temporal Consistency (the presence of flickering or artifacts). Each dimension is rated on a 5-point scale (1: lowest, 5: highest). A total of 1440 valid evaluations were collected. As shown in Figure 3, BridgeRemoval significantly outperformed existing baselines, achieving higher preference rates across all evaluation criteria.

6. Conclusion

This study introduces BridgeRemoval, a stochastic bridge framework that reformulates video object removal as a video-to-video translation task. By leveraging structural priors from the input video and employing an adaptive mask modulation strategy, our method effectively balances background fidelity with generative flexibility. Extensive experiments demonstrate that our approach outperforms existing methods in visual quality and temporal consistency. This work validates the efficacy of bridge-based models for video object removal and we believe that the designs in BridgeRemoval will provide valuable insights to the video edit community.

Impact Statement

This paper introduces BridgeRemoval, a Bridge-based framework advancing high-fidelity video object removal. While enhancing content creation and post-production, this technology poses potential risks for manipulating authentic footage or altering context. We emphasize the importance of responsible deployment and detection methods to mitigate the misuse of generative video editing tools.

References

https://www.pexels.com/.

https://www.tiktok.com/.

https://aistudio.google.com/models/veo-3.

Albergo, M. S., Boffi, N. M., and Vanden-Eijnden, E. Stochastic interpolants: A unifying framework for flows and diffusions. *JMLR*, 2025.

Bai, Q., Wang, Q., Ouyang, H., Yu, Y., Wang, H., Wang, W., Cheng, K. L., Ma, S., Zeng, Y., Liu, Z., Xu, Y., Shen, Y., and Chen, Q. Scaling instruction-based video editing with a high-quality synthetic dataset. *arXiv preprint arXiv:2510.15742*, 2025a.

Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al. Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*, 2025b.

Bian, Y., Zhang, Z., Ju, X., Cao, M., Xie, L., Shan, Y., and Xu, Q. VideoPainter: Any-length video inpainting and editing with plug-and-play context control. In *SIGGRAPH*, 2025.

Chen, T., Liu, G.-H., and Theodorou, E. Likelihood training of schrödinger bridge using forward-backward sdes theory. In *ICLR*, 2021.

Chen, Z., He, G., Zheng, K., Tan, X., and Zhu, J. Schrodinger bridges beat diffusion models on text-to-speech synthesis. *arXiv preprint arXiv:2312.03491*, 2023.

De Bortoli, V., Thornton, J., Heng, J., and Doucet, A. Diffusion schrödinger bridge with applications to score-based generative modeling. In *NeurIPS*, 2021.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024.

Gao, C., Ding, L., Cai, X., Huang, Z., Wang, Z., and Xue, T. LoRA-Edit: Controllable first-frame-guided video editing via mask-aware lora fine-tuning. *arXiv preprint arXiv:2506.10082*, 2025.

He, G., Zheng, K., Chen, J., Bao, F., and Zhu, J. Consistency diffusion bridge models. In *NeurIPS*, 2024.

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

Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., Wang, Y., Chen, X., Wang, L., Lin, D., Qiao, Y., and Liu, Z. VBench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024.

Jiang, Z., Han, Z., Mao, C., Zhang, J., Pan, Y., and Liu, Y. VACE: All-in-one video creation and editing. In *ICCV*, 2025.

Ju, X., Liu, X., Wang, X., Bian, Y., Shan, Y., and Xu, Q. BrushNet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In *ECCV*, 2024.

Kim, D., Woo, S., Lee, J.-Y., and Kweon, I. S. Deep video inpainting. In *CVPR*, 2019.

Kingma, D., Salimans, T., Poole, B., and Ho, J. Variational diffusion models. In *NeurIPS*, 2021.

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

Labs, B. F. Flux. https://github.com/black-forest-labs/flux, 2024.

Lee, Y.-C., Lu, E., Rumbley, S., Geyer, M., Huang, J.-B., Dekel, T., and Cole, F. Generative Omnimatte: Learning to decompose video into layers. In *CVPR*, 2025.

Li, A., Zhao, S., Ma, X., Gong, M., Qi, J., Zhang, R., Tao, D., and Kotagiri, R. Short-term and long-term context aggregation network for video inpainting. In *ECCV*, 2020.

Li, B., Xue, K., Liu, B., and Lai, Y.-K. BBDM: Image-to-image translation with brownian bridge diffusion models. In *CVPR*, 2023.

Li, C., Chen, Z., Bao, F., and Zhu, J. Bridge-SR: Schrödinger bridge for efficient sr. In *ICASSP*, 2025a.

Li, X., Xue, H., Ren, P., and Bo, L. DiffuEraser: A diffusion model for video inpainting. *arXiv preprint arXiv:2501.10018*, 2025b.

Liao, X., Zeng, X., Song, Z., Fu, Z., Yu, G., and Lin, G. In-context learning with unpaired clips for instruction-based video editing. *arXiv preprint arXiv:2510.14648*, 2025.

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

Liu, G.-H., Vahdat, A., Huang, D.-A., Theodorou, E. A., Nie, W., and Anandkumar, A. I ^{2} SB: Image-to-image schrödinger bridge. In *ICML*, 2023.

Liu, X., Gong, C., and Liu, Q. Flow Straight and Fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *ICLR*, 2019.

Miao, C., Feng, Y., Zeng, J., Gao, Z., Liu, H., Yan, Y., Qi, D., Chen, X., Wang, B., and Zhao, H. ROSE: Remove objects with side effects in videos. In *NeurIPS*, 2025.

Ouyang, W., Dong, Y., Yang, L., Si, J., and Pan, X. I2VEedit: First-frame-guided video editing via image-to-video diffusion models. In *SIGGRAPH Asia*, 2024.

Peluchetti, S. Non-denoising forward-time diffusions. *arXiv preprint arXiv:2312.14589*, 2023.

Perazzi, F., Pont-Tuset, J., McWilliams, B., Gool, L. V., Gross, M., and Sorkine-Hornung, A. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016.

Perez, E., Strub, F., De Vries, H., Dumoulin, V., and Courville, A. FiLM: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., and Rombach, R. SDXL: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020.

Ravi, N., Gabeur, V., Hu, Y.-T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., et al. SAM 2: Segment anything in images and videos. In *ICLR*, 2025.

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

Tong, A., Malkin, N., Fatras, K., Atanackovic, L., Zhang, Y., Huguet, G., Wolf, G., and Bengio, Y. Simulation-free schrödinger bridges via score and flow matching. In *ICAIS*, 2024.

Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.-W., Chen, D., Yu, F., Zhao, H., Yang, J., et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Wang, Y., Chen, Z., Chen, X., Wei, Y., Zhu, J., and Chen, J. FrameBridge: Improving image-to-video generation with bridge models. In *ICML*, 2025.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, 2025a.

Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D., Zhang, Y., Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., and Tang, J. CogVideoX: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025b.

Zhang, K., Fu, J., and Liu, D. Flow-guided transformer for video inpainting. In *ECCV*, 2022.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

Zheng, K., He, G., Chen, J., Bao, F., and Zhu, J. Diffusion bridge implicit models. In *ICLR*, 2025.

Zhou, L., Lou, A., Khanna, S., and Ermon, S. Denoising diffusion bridge models. In *ICLR*, 2023a.

Zhou, S., Li, C., Chan, K. C. K., and Loy, C. C. ProPainter: Improving propagation and transformer for video inpainting. In *ICCV*, 2023b.

Zi, B., Peng, W., Qi, X., Wang, J., Zhao, S., Xiao, R., and Wong, K.-F. MiniMax-Remover: Taming bad noise helps video object removal. *arXiv preprint arXiv:2505.24873*, 2025a.

Zi, B., Ruan, P., Chen, M., Qi, X., Hao, S., Zhao, S., Huang, Y., Liang, B., Xiao, R., and Wong, K.-F. Señorita-2M: A high-quality instruction-based dataset for general video editing by video specialists. *arXiv preprint arXiv:2502.06734*, 2025b.

Zou, X., Yang, L., Liu, D., and Lee, Y. J. Progressive temporal feature alignment network for video inpainting. In *CVPR*, 2021.

A. Pseudo Code for the Training and inference of BridgeRemoval

To clearly delineate the distinctions between BridgeRemoval and other models (such as ROSE (Miao et al., 2025), which is trained with the Flow Matching framework.) in video object removal, we present a detailed comparison of their algorithmic formulations. The detailed configurations are summarized in Table 3.

Table 3. Comparison of algorithmic settings between the Flow Matching Baseline and BridgeRemoval (Ours).

	Flow Matching Baseline	**BridgeRemoval (Ours)**
Input Data	source video V_{src}, target video V_{tgt} (during training), mask M	source video V_{src}, target video V_{tgt} (during training), mask M
Condition	$y = \text{Concat}(z_M, z_{\text{src}})$	$y = \text{Concat}(z_M, z_{\text{src}})$
Training	$z_t = (1-t)z_{\text{tgt}} + t\epsilon$	$z_t = a_t z_{\text{tgt}} + b_t z_{\text{src}} + c_t \epsilon$
Prediction Target	Velocity $v = \epsilon - z_{\text{tgt}}$	Velocity $u_t = \frac{a_t}{\rho_t}\epsilon - \frac{c_t}{\rho_t} z_{\text{tgt}}$
Inference Init	Gaussian Noise $\mathcal{N}(0, I)$	Source Prior z_{src}
Trajectory	ODE Integration ($z_1 \to z_0$)	Bridging ($z_{\text{src}} \to z_{\text{tgt}}$)

We provide the pseudo code for the training and inference process of BridgeRemoval (See Algorithm 4 and 6). Meanwhile, we also provide that of Flow Matching-based video object removal models (See Algorithm 3 and 5) to show the distinctions between BridgeRemoval and Flow Matching-based models. In practical implementation, time-dependent coefficients (e.g., $\bar{\alpha}t$ or a_t, b_t, c_t) are typically precomputed and retrieved from a schedule table for computational efficiency.

Algorithm 3 Flow Matching Training for Video Object Removal

Input: Source Video V_{src}, Target Video V_{tgt}, Mask M, Text Prompt P, Model v_θ, VAE Encoder $\mathcal{E}$, T5 Encoder
Output: Optimized parameters θ

Encode inputs to latent space: $z_{\text{src}} \leftarrow \mathcal{E}(V_{\text{src}})$, $z_0 \leftarrow \mathcal{E}(V_{\text{tgt}})$, and $z_M \leftarrow \mathcal{E}(M)$
Construct conditional input by concatenating mask and source video: $y \leftarrow \text{Concat}(z_M, z_{\text{src}})$
Encode text prompt to embeddings: $c_{\text{text}} \leftarrow \text{T5}(P)$
repeat
  Sample continuous time $t \sim \mathcal{U}(0, 1)$ and Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$
  Construct interpolated state via linear interpolation: $z_t = (1-t)z_0 + t\epsilon$
  Compute ground-truth velocity target: $v = \epsilon - z_0$
  Predict velocity field using the model: $\hat{v} \leftarrow v_\theta(z_t, t, y, c_{\text{text}})$
  Compute velocity matching loss: $\mathcal{L} = \|\hat{v} - v\|^2$
  Update parameters θ by gradient descent
until *convergence*;

Algorithm 4 BridgeRemoval Training for Video Object Removal

Input: Source Video V_{src}, Target Video V_{tgt}, Mask M, Text Prompt P, Model v_θ, Variance Schedule σ_t, VAE Encoder $\mathcal{E}$, T5 Encoder
Output: Optimized parameters θ

Encode inputs to latent space: $z_{\text{src}} \leftarrow \mathcal{E}(V_{\text{src}})$, $z_{\text{tgt}} \leftarrow \mathcal{E}(V_{\text{tgt}})$, and $z_M \leftarrow \mathcal{E}(M)$
Construct conditional input: $y \leftarrow \text{Concat}(z_M, z_{\text{src}})$
Encode text prompt to embeddings: $c_{\text{text}} \leftarrow \text{T5}(P)$
repeat
  Sample interpolation time $t \sim \mathcal{U}(0, 1)$ and noise $\epsilon \sim \mathcal{N}(0, I)$
  Compute SDE coefficients a_t, b_t, c_t derived from the variance schedule σ_t
  Construct intermediate bridge state via ternary interpolation: $z_t = a_t z_{\text{tgt}} + b_t z_{\text{src}} + c_t \epsilon$
  Compute ground-truth velocity target u_t (V-Prediction parameterization): $\rho_t = \sqrt{a_t^2 + c_t^2}, \quad u_t = \frac{a_t}{\rho_t}\epsilon - \frac{c_t}{\rho_t} z_{\text{tgt}}$
  Predict velocity field using the model: $\hat{v}_t \leftarrow v_\theta(z_t, t, y, c_{\text{text}})$
  Compute velocity matching loss: $\mathcal{L} = \|\hat{v}_t - u_t\|^2$
  Update parameters θ by gradient descent
until *convergence*;

Algorithm 5 Flow Matching Inference for Video Object Removal

Input: Source Video V_{src}, Mask M, Text Prompt P, Model v_θ, Steps N, VAE Encoder $\mathcal{E}$, VAE Decoder $\mathcal{D}$
Output: Inpainted Video V_{out}

Encode inputs: $z_{\text{src}} \leftarrow \mathcal{E}(V_{\text{src}})$, $z_M \leftarrow \mathcal{E}(M)$ and condition $y \leftarrow \text{Concat}(z_M, z_{\text{src}})$
Encode text prompt: $c_{\text{text}} \leftarrow \text{T5}(P)$
Initialize latent state from pure Gaussian noise: $z \sim \mathcal{N}(0, I)$
Define time schedule $\{t_k\}_{k=N}^{0}$ from 1 to 0
for $k = N, \ldots, 1$ **do**
  Set current time $t \leftarrow t_k$ and next time $t' \leftarrow t_{k-1}$
  Predict velocity field: $\hat{v} \leftarrow v_\theta(z, t, y, c_{\text{text}})$
  Compute step size: $\Delta t = t' - t$
  Euler ODE integration step: $z \leftarrow z + \hat{v} \cdot \Delta t$
end
Decode output latent to pixel space: $V_{\text{out}} \leftarrow \mathcal{D}(z)$

Algorithm 6 BridgeRemoval Inference for Video Object Removal

Input: Source Video V_{src}, Mask M, Text Prompt P, Model v_θ, Steps N, Variance Schedule σ_t, VAE Encoder $\mathcal{E}$, VAE Decoder $\mathcal{D}$
Output: Inpainted Video V_{out}

Encode inputs: $z_{\text{src}} \leftarrow \mathcal{E}(V_{\text{src}})$, $z_M \leftarrow \mathcal{E}(M)$ and condition $y \leftarrow \text{Concat}(z_M, z_{\text{src}})$
Encode text prompt: $c_{\text{text}} \leftarrow \text{T5}(P)$
Initialize state $z \leftarrow z_{\text{src}}$ using the source video as the prior
Define time schedule $\{t_k\}_{k=N}^{0}$ from 1 to 0
for $k = N, \ldots, 1$ **do**
  Set current time $t \leftarrow t_k$ and next time $t' \leftarrow t_{k-1}$
  Predict velocity field: $\hat{v}_t \leftarrow v_\theta(z, t, y, c_{\text{text}})$
  Recover predicted target $\hat{z}_{0|t}$ from velocity (analytic inversion): $\rho_t = \sqrt{a_t^2 + c_t^2}, \quad \hat{z}_{0|t} = \frac{a_t}{\rho_t^2} z - \frac{c_t}{\rho_t} \hat{v}_t - \frac{a_t b_t}{\rho_t^2} z_{\text{src}}$
  if $k > 1$ **then**
    Compute SDE solver weights w_1, w_2, w_3 based on σ_t
    Sample noise $\epsilon' \sim \mathcal{N}(0, I)$
    Update state via Euler solver: $z \leftarrow w_1 z + w_2 \hat{z}_{0|t} + w_3 \epsilon'$
  else
    Set final state directly: $z \leftarrow \hat{z}_{0|t}$
  end
end
Decode output video: $V_{\text{out}} \leftarrow \mathcal{D}(z)$

B. Mathematical Proofs

B.1. Derivation of Bridge Marginal Distribution

To provide a theoretical foundation for the proposed BridgeRemoval and justify the coefficients used in Eq. 10, we first derive the marginal distribution of a general bridge process $p_{t,\text{bridge}}(\mathbf{z}_t|\mathbf{z}_0, \mathbf{z}_T)$ following the framework of DDBM (Zhou et al., 2023a).

In this section, we adopt standard diffusion notation (Ho et al., 2020), letting $\mathbf{z}_0$ represent the clean data (Target) and $\mathbf{z}_T$ represent the prior (Source). We explicitly use α_t, σ_t to denote the noise schedule coefficients of the underlying process. We will later map these general theoretical results to the specific σ_t-based parameterization used in our main text.

Using Bayes' rule and the Markov property of diffusion (Kingma et al., 2021) :

$$p_{t,\text{bridge}}(\mathbf{z}_t|\mathbf{z}_0, \mathbf{z}_T) = \frac{p(\mathbf{z}_T|\mathbf{z}_t)p(\mathbf{z}_t|\mathbf{z}_0)}{p(\mathbf{z}_T|\mathbf{z}_0)}. \tag{18}$$

Assuming the underlying process is a VP-SDE with Gaussian transition kernels:

$$\begin{aligned} p(\mathbf{z}_t|\mathbf{z}_0) &= \mathcal{N}(\mathbf{z}_t; \alpha_t \mathbf{z}_0, \sigma_t^2 \mathbf{I}), \\ p(\mathbf{z}_T|\mathbf{z}_t) &= \mathcal{N}\left(\mathbf{z}_T; \frac{\alpha_T}{\alpha_t}\mathbf{z}_t, \left(\sigma_T^2 - \frac{\alpha_T^2}{\alpha_t^2}\sigma_t^2\right)\mathbf{I}\right). \end{aligned} \tag{19}$$

where $\text{SNR}_t = \alpha_t^2/\sigma_t^2$ (Kingma et al., 2021).

The resulting bridge marginal is also Gaussian (Zhou et al., 2023a) :

$$p_{t,\text{bridge}}(\mathbf{z}_t|\mathbf{z}_0, \mathbf{z}_T) = \mathcal{N}(\mathbf{z}_t; \boldsymbol{\mu}_t, \Sigma_t \mathbf{I}), \tag{20}$$

where the reparameterized state can be written as:

$$\mathbf{z}_t = A_t \mathbf{z}_0 + B_t \mathbf{z}_T + C_t \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \tag{21}$$

with general coefficients:

$$A_t = \alpha_t \left(1 - \frac{\text{SNR}_T}{\text{SNR}_t}\right), \quad B_t = \frac{\text{SNR}_T}{\text{SNR}_t}\frac{\alpha_t}{\alpha_T}, \quad C_t = \sigma_t \sqrt{1 - \frac{\text{SNR}_T}{\text{SNR}_t}}. \tag{22}$$

Here we prove that the coefficients a_t, b_t, c_t defined in Eq. 10 are a valid instance of the general theory derived above.

In the main text, we define a schedule variable σ_t^2 (let's denote it $\hat{\sigma}_t^2$ here to distinguish from diffusion variance) representing the cumulative variance contribution from the source. Specifically, we set:

$$b_t = \frac{\hat{\sigma}_t^2}{\hat{\sigma}_1^2}$$

Comparing this to the general form of B_t in Eq. 22:

$$\frac{\hat{\sigma}_t^2}{\hat{\sigma}_1^2} = \frac{\text{SNR}_T}{\text{SNR}_t}\frac{\alpha_t}{\alpha_T}$$

This equality holds if we define the underlying diffusion schedule such that the SNR ratio matches our variance schedule. Furthermore, for a standard VP-SDE where $\alpha_t^2 + \sigma_t^2 = 1$, we typically have $\alpha_T \approx 0$ (so $\text{SNR}_T \approx 0$) and $\alpha_0 = 1$.

By simplifying the general coefficients under the condition that our bridge schedule is consistent with the underlying diffusion physics, we recover the forms:

$$a_t = \frac{\bar{\sigma}_t^2}{\sigma_1^2}, \quad b_t = \frac{\sigma_t^2}{\sigma_1^2}, \quad c_t = \frac{\bar{\sigma}_t \sigma_t}{\sigma_1}. \tag{23}$$

where $\bar{\sigma}_t^2 = \sigma_1^2 - \sigma_t^2$. Specifically, notice that at endpoints:

$t = 0(\text{Target}) \implies \sigma_0 = 0 \implies a_0 = 1, b_0 = 0, c_0 = 0$, recovering $\mathbf{z}_0$.

$t = 1(\text{Source}) \implies \sigma_1 = \sigma_1 \implies a_1 = 0, b_1 = 1, c_1 = 0$, recovering $\mathbf{z}_T$.

Thus, Eq. 10 in the main text is theoretically grounded in the DDBM framework, providing a specific, tractable parameterization for video removal.

B.2. Derivation of the Guidance Term h

In the SDE formulation (Eq. 7 in main text), we introduced the guidance term $h(\mathbf{z}_t, t, \mathbf{z}_{\text{src}}) = \nabla_{\mathbf{z}_t} \log p(\mathbf{z}_{\text{src}}|\mathbf{z}_t)$. Since $p_{\text{diff}}(\mathbf{z}_{\text{src}}|\mathbf{z}_t)$ is Gaussian (as shown above), we can derive h analytically.

Recall that:

$$p_{\text{diff}}(\mathbf{z}_{\text{src}}|\mathbf{z}_t) = \mathcal{N}\left(\mathbf{z}_{\text{src}}; \frac{\alpha_1}{\alpha_t}\mathbf{z}_t, \Sigma_{1|t}\mathbf{I}\right)$$

where $\Sigma_{1|t} = \sigma_1^2 - \frac{\alpha_1^2}{\alpha_t^2}\sigma_t^2$.

Taking the logarithm of the Gaussian density:

$$\log p(\mathbf{z}_{\text{src}}|\mathbf{z}_t) = -\frac{\left\|\mathbf{z}_{\text{src}} - \frac{\alpha_1}{\alpha_t}\mathbf{z}_t\right\|^2}{2\Sigma_{1|t}} + C$$

Computing the gradient with respect to $\mathbf{z}_t$:

$$\begin{aligned} h(\mathbf{z}_t, t, \mathbf{z}_{\text{src}}) &= \nabla_{\mathbf{z}_t}\left(-\frac{\left\|\mathbf{z}_{\text{src}} - \frac{\alpha_1}{\alpha_t}\mathbf{z}_t\right\|^2}{2\Sigma_{1|t}}\right) \\ &= \frac{\frac{\alpha_1}{\alpha_t}}{\Sigma_{1|t}}\left(\mathbf{z}_{\text{src}} - \frac{\alpha_1}{\alpha_t}\mathbf{z}_t\right) \end{aligned}$$

This explicit form confirms that the guidance term required for the SDE formulation is indeed analytically tractable.

The coefficients expressions given in Eq. 10 of the main text ($a_t = \bar{\sigma}_t^2/\sigma_1^2$, etc.) are instances of the general form derived above under a specific variance schedule. Specifically, when the bridge variance schedule satisfies $\text{SNR}_t/\text{SNR}_1 = \bar{\sigma}_t^2/\sigma_t^2$, the general formulas simplify to the specific forms used in our implementation. This demonstrates that our method strictly adheres to the DDBM theoretical framework.

B.3. Equivalence of Training Objective

In the main text, we define the training objective as a velocity matching loss:

$$\mathcal{L} = \mathbb{E}\left[\|\mathbf{v}_\theta - \mathbf{u}_t\|^2\right]$$

where the target velocity is given by $\mathbf{u}_t = \frac{d\mathbf{z}_t}{dt}$. Here, we prove that this objective is theoretically equivalent to Denoising Bridge Score Matching.

The true score function of the bridge process is defined as $\nabla_{\mathbf{z}_t}\log p_t(\mathbf{z}_t|\mathbf{z}_{\text{tgt}}, \mathbf{z}_{\text{src}})$ (Zhou et al., 2023a). Using the Gaussian conclusion from Appendix B.1, we can explicitly write:

$$\nabla_{\mathbf{z}_t}\log p_t = -\frac{\mathbf{z}_t - \boldsymbol{\mu}_t}{\sigma_{t|\text{bridge}}^2} = -\frac{\mathbf{z}_t - (a_t\mathbf{z}_{\text{tgt}} + b_t\mathbf{z}_{\text{src}})}{\sigma_{t|\text{bridge}}^2}$$

Let us define the normalized residual term $\boldsymbol{\epsilon}_{\text{res}} = \frac{\mathbf{z}_t - a_t\mathbf{z}_{\text{tgt}} - b_t\mathbf{z}_{\text{src}}}{c_t}$, where $c_t = \sigma_{t|\text{bridge}}$.

If we parameterize the model to predict this normalized residual (which is equivalent to predicting velocity under the V-Prediction framework), i.e., letting the model output $\mathbf{v}_\theta \propto \boldsymbol{\epsilon}_{\text{res}}$, then the Score Matching loss:

$$\mathcal{L}_{\text{SM}} = \mathbb{E}\left[\|\mathbf{s}_\theta - \nabla_{\mathbf{z}_t}\log p_t\|^2\right]$$

can be rewritten in the Mean Squared Error (MSE) form:

$$\mathcal{L}_{\text{SM}} \propto \mathbb{E}\left[\left\|\mathbf{v}_\theta(\mathbf{z}_t) - \underbrace{\left(\frac{a_t}{\rho_t}\boldsymbol{\epsilon} - \frac{c_t}{\rho_t}\mathbf{z}_{\text{tgt}}\right)}_{\text{Ground Truth Velocity } \mathbf{u}_t}\right\|^2\right]$$

This transformation utilizes the reparameterization formula $\mathbf{z}_t = a_t\mathbf{z}_{\text{tgt}} + b_t\mathbf{z}_{\text{src}} + c_t\boldsymbol{\epsilon}$.

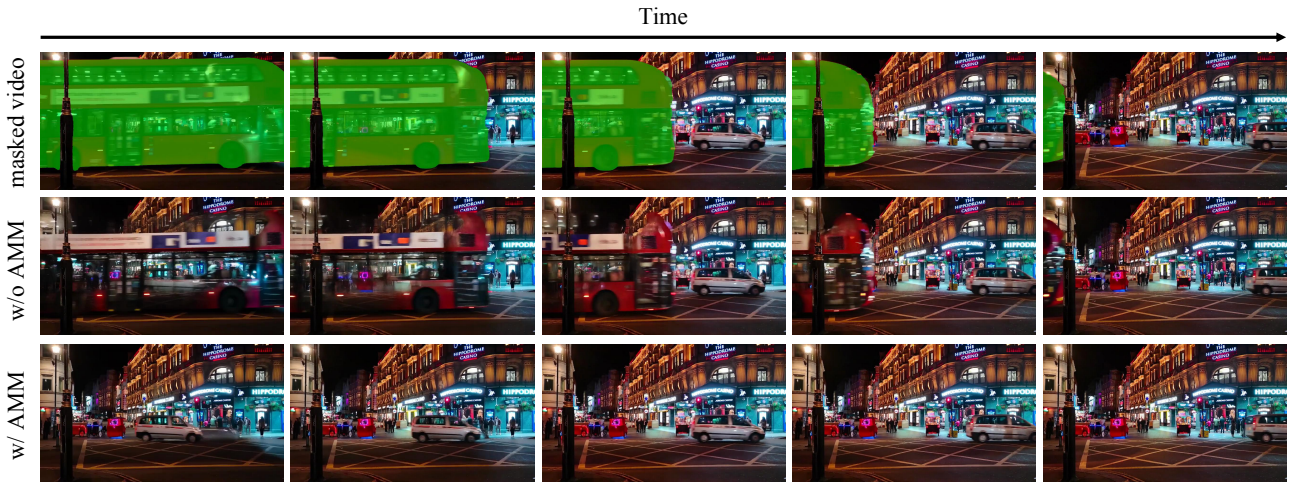

Figure 4. Comparison of w/ and w/o adaptive mask modulation.

Minimizing our proposed velocity regression loss $\mathcal{L}$ is mathematically equivalent to minimizing the Score Matching error of the bridge process. This ensures that the vector field learned by the model can correctly push the distribution from the source video to the target video. This proves that our MSE training objective is mathematically grounded in Score Matching theory, guaranteeing that the model learns the correct gradient field for the bridge process.

C. Ablation Study

C.1. Influence of AMM

We conduct an ablation study to evaluate the impact of AMM. As shown in Figure 4, a bus occupies approximately 80 percent of the video frame. When AMM is disabled, our BridgeRemoval fails to fully erase the large object. The reconstructed region retains visible artifacts and structural remnants of the bus, indicating that the strong structural prior from the source video impedes sufficient deviation needed for plausible inpainting of extensive occlusions. In contrast, with AMM enabled, the model successfully removes the entire bus and synthesizes a coherent background that aligns with the surrounding scene context and maintains temporal consistency across frames. This demonstrates that AMM effectively mitigates the rigidity induced by the bridge prior. By amplifying the embeddings in unmasked background regions, AMM enhances the model's reliance on reliable contextual cues while granting greater generative flexibility within the masked area.

C.2. Influence of Inference Steps

Diffusion-based video object removal methods (Miao et al., 2025; Jiang et al., 2025; Lee et al., 2025) typically initialize inference from random noise and often require 50 or more denoising steps. In contrast, our BridgeRemoval starts inference directly from the source video. To investigate the impact of inference steps, we evaluate our method on the DAVIS dataset using 10, 20, 30, 40, and 50 steps, as reported in Table 4. The results show that with only 10 steps, BridgeRemoval already achieves the best CLIP-F, PSNR, and MSE. As the number of inference steps increases, CLIP-T gradually improves while PSNR steadily decreases. This trend is intuitive: more steps lead to cleaner object removal, reflected in higher CLIP-T, but at the cost of background fidelity, resulting in lower PSNR.

Figure 5 illustrates this behavior on a representative example. At 10 steps, the masked region lacks fine texture details. As the step count increases, the inpainted content becomes progressively more realistic and coherent with the scene semantics. To balance object removal quality and background preservation, we adopt 50 inference steps in our main experiments.

D. More Quantitative Results

Some methods, such as ROSE (Miao et al., 2025), employ VBench (Huang et al., 2024) to evaluate the performance of video object removal. However, this may be inappropriate. VBench was specifically designed for assessing video generation models, decomposing video generation quality into multiple dimensions including Motion Smoothness, Background

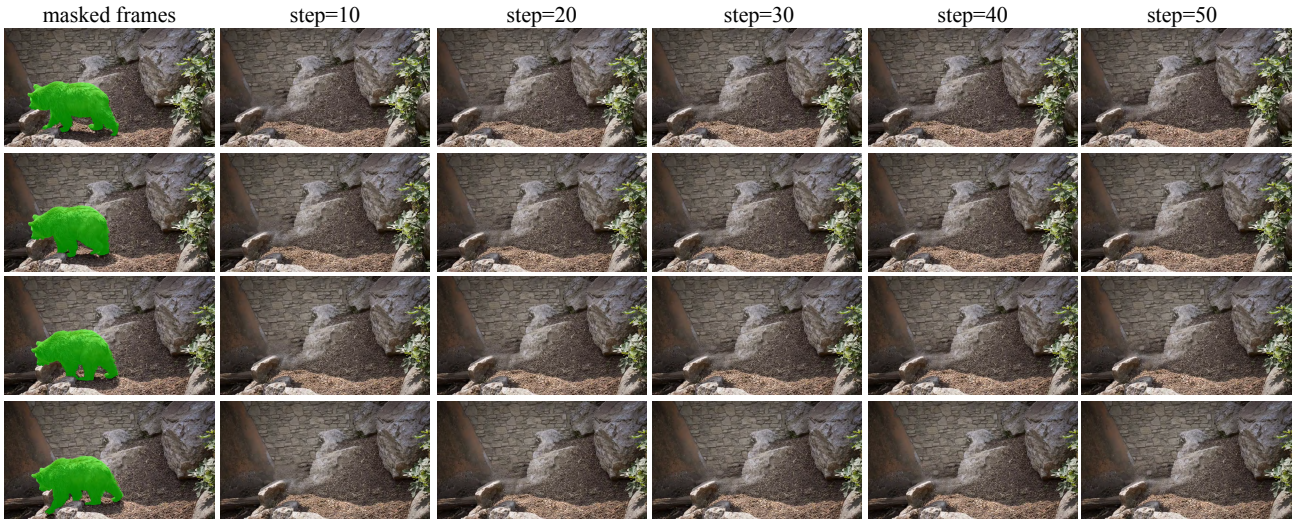

Figure 5. Visualization results of BridgeRemoval at different inference steps. This video is from the DAVIS dataset.

Table 4. Comparison of different inference steps on DAVIS dataset. **Red** stands for the best.

	Video Quality		Unmasked Region Preservation		
Inference steps	CLIP-T ↑	CLIP-F ↑	PSNR ↑	MSE ↓	LPIPS ↓
10	0.2765	**0.9771**	**28.4155**	**128.6873**	0.0985
20	0.2779	0.9767	28.3085	131.6830	0.0977
30	0.2784	0.9769	28.2897	132.2086	0.0975
40	0.2786	0.9767	28.2555	132.7415	**0.0973**
50	**0.2788**	0.9768	28.2154	133.3605	0.0977

Table 5. VBench-based evaluation on DAVIS dataset. **Red** stands for the best, **Blue** stands for the second best.

Method	VBench				
	Motion Smoothness ↑	Background Consistency ↑	Temporal Flickering ↓	Subject Consistency ↑	Imaging Quality ↑
Senorita (Zi et al., 2025b)	0.9724	0.8885	0.9437	0.8152	0.5187
Ditto (Bai et al., 2025a)	0.9748	0.9166	**0.9345**	**0.9119**	**0.6948**
ICVE (Liao et al., 2025)	**0.9803**	0.9173	0.9445	0.8983	0.5325
ROSE (Miao et al., 2025)	0.9750	**0.9245**	0.9384	0.9051	0.6239
VACE (Jiang et al., 2025)	0.9716	0.9173	**0.9329**	0.8974	**0.6618**
GenOmnimatte (Lee et al., 2025)	**0.9757**	0.9216	0.9384	**0.9119**	0.6179
Ours	0.9736	**0.9305**	0.9358	**0.9111**	0.6376

Table 6. VBench-based evaluation on BridgeRemoval-Bench. **Red** stands for the best, **Blue** stands for the second best.

Method	VBench				
	Motion Smoothness ↑	Background Consistency ↑	Temporal Flickering ↓	Subject Consistency ↑	Imaging Quality ↑
Senorita (Zi et al., 2025b)	0.9901	0.9162	0.9770	0.8771	0.5470
Ditto (Bai et al., 2025a)	0.9892	0.9479	**0.9715**	**0.9589**	**0.7423**
ICVE (Liao et al., 2025)	**0.9935**	0.9402	0.9791	**0.9561**	0.5803
ROSE (Miao et al., 2025)	0.9918	0.9470	0.9764	0.9536	0.6257
VACE (Jiang et al., 2025)	0.9907	**0.9532**	**0.9740**	0.9436	**0.6462**
GenOmnimatte (Lee et al., 2025)	**0.9925**	0.9417	0.9769	0.9558	0.6373
Ours	0.9924	**0.9554**	0.9758	0.9557	0.6429

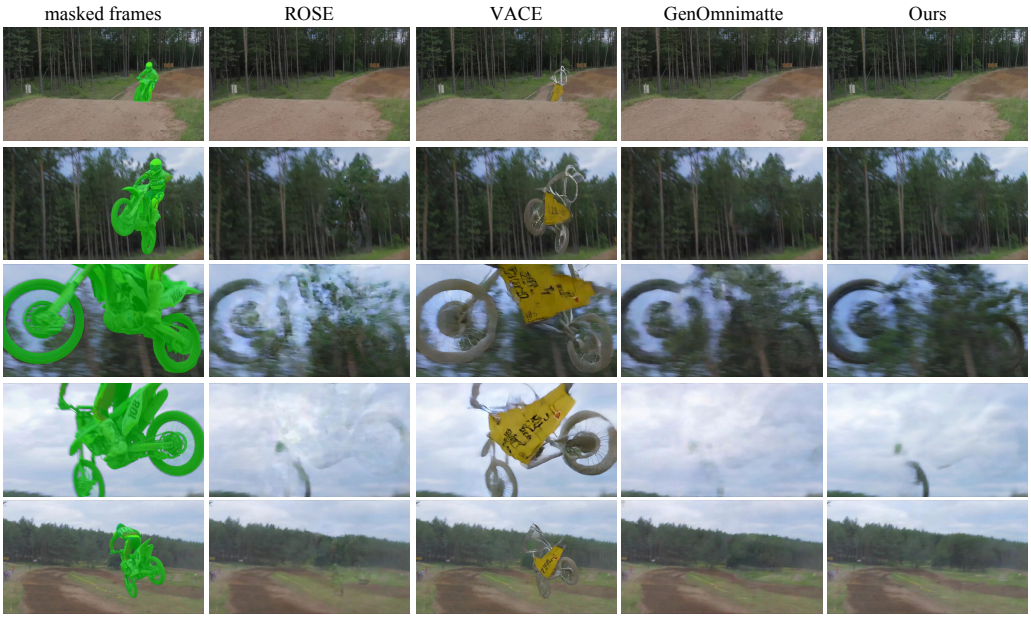

Figure 6. Limitations in removing fast moving objects. This video is from the DAVIS dataset.

Consistency, Temporal Flickering, Subject Consistency, and Imaging Quality. These metrics have limitations when applied to the video object removal task because they cannot determine whether the target object has been genuinely removed. For instance, VACE (Jiang et al., 2025) often generates a new object that conforms to the shape of the mask. As shown in Tables 5 and 6, VACE achieves relatively high scores on Temporal Flickering and Imaging Quality, yet this does not indicate effective object removal. Furthermore, the scores across different methods on VBench metrics exhibit very low discriminability, which fails to reflect meaningful performance gaps. For example, all methods achieve Motion Smoothness scores above 0.97, suggesting limited utility of this metric for distinguishing object removal quality.

E. Limitation

One limitation of BridgeRemoval lies in its performance on videos containing objects with rapid motion. As illustrated in Figure 6, when the target object moves quickly across frames, our method may not fully remove it and can leave visible artifacts or ghosting effects in the inpainted region. Importantly, this difficulty is not specific to our approach but appears to be a general challenge in the field of video object removal. We observe that state of the art methods such as ROSE (Miao et al., 2025), VACE (Jiang et al., 2025), and GenOmnimatte (Lee et al., 2025) also exhibit similar failure modes on the same types of fast moving objects. In future work, we plan to explore hybrid architectures that integrate explicit motion estimation with stochastic bridge models to enhance robustness in high speed scenarios.

F. More Qualitative Results

More qualitative samples can be visited at: `https://bridgeremoval.github.io/`