Unified Long Video Inpainting and Outpainting via Overlapping High-Order Co-Denoising

Shuangquan Lyu
Carnegie Mellon University

Jian Mao
Jilin University

Yue Ma
Tsinghua University

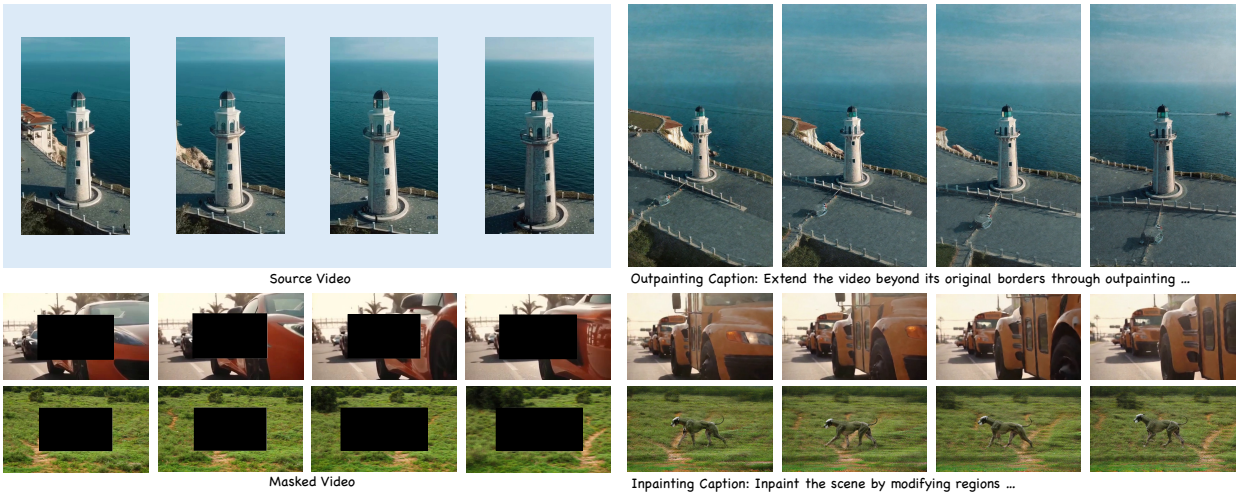


Fig. 1: Showcase of our methods. We present a unified approach for long video inpainting and outpainting that extends a text-to-video diffusion model to generate long-horizon, spatially edited videos with high fidelity under memory bounded by window size.‘

***Abstract*—Diffusion-based text-to-video models are increasingly capable, but mask-based editing over hundreds of frames remains challenging: naïve long-video generation suffers from memory blow-up, window seams, and temporal drift, while existing editors often require specialized modules or heavy fine-tuning. We present Overlapping High-Order Co-Denoising, a lightweight framework that turns a single pre-trained text-to-video model into a unified inpainting–outpainting editor. We train only LoRA adapters using mixed interior and border masks together with a dual-region loss that improves synthesis inside the mask while explicitly preserving known content. At inference, we denoise long latent sequences using overlapping windows, apply second-order Heun sampling within each window, and fuse overlaps with Hamming-weighted blending to reduce boundary artifacts and improve temporal coherence. On InpaintBench (30 real-world videos, 81–300 frames), our method outperforms Wan 2.1 variants and VACE in background faithfulness (SSIM/LPIPS), temporal consistency (tLPIPS), and text alignment (CLIP), and scales to long horizons, demonstrated up to 800 frames, with memory bounded by the chosen window size.**

***Index Terms*—Video diffusion models, mask-based video editing, video inpainting and outpainting, long-form video editing, low-rank adaptation (LoRA), high-order diffusion sampling**

I. Introduction

Generating videos from textual descriptions is a fundamental problem in generative modeling. Recent foundation text-to-video diffusion models [1], [2] have made remarkable progress in generating short video clips from textual descriptions. With the large amount of training data, large-scale training and architectural refinements, they naturally possess video editing capabilities. Inpainting and outpainting [3], [4] are two video editing tasks which have been widely studied. Building on these foundation models, recent work has explored text-guided masked editing.

However, two major challenges remain when adapting a single model to long-video editing. First, extending diffusion models to substantially longer videos is difficult due to memory limits and temporal consistency issues—naïve scaling often leads to drift or stitching artifacts over time. Second, enabling fine-grained mask-based inpainting and outpainting within one unified framework is non-trivial. While recent works have begun exploring controllable video editing, current foundation models still offer limited spatial control without additional specialized modules. Enhancing a pre-trained model's controllability for selective region editing, especially over long durations, remains an open problem.

Built on Wan [2], we adapt a single pre-trained text-to-video model for *mask-based* inpainting and outpainting by inserting LoRA adapters and training on randomly masked clips with a dual-region MSE loss. For long-horizon inference, we denoise overlapping temporal windows using a second-order Heun update and fuse overlaps with Hamming-weighted blending to reduce seams and drift while keeping memory bounded by window size. Our design improves long-video fidelity and temporal stability while keeping GPU memory bounded. Experiments show consistent gains on CLIP [5], SSIM [6], LPIPS [7] and tLPIPS over strong baselines (e.g.,

Wan 2.1 14B).

Our contributions are summarized as follows:

- We adapt a single pre-trained text-to-video diffusion model to both video inpainting and outpainting using mask-conditioned LoRA fine-tuning with a dual-region reconstruction loss.
- We develop a sliding-window diffusion inference scheme that combines a Heun solver with Hamming-weighted blending, reducing seam artifacts and temporal drift while keeping memory bounded by window size.
- We build and evaluate on InpaintBench (30 videos, 81–300 frames), achieving consistent improvements in CLIP, SSIM, LPIPS, and tLPIPS over strong baselines while enabling long-horizon editing, demonstrated up to 800 frames, with bounded memory.

We unify mask-conditioned LoRA adaptation and long-horizon co-denoising in a single pipeline. The dataset and code will be released publicly.

II. Related Work

Foundation text-to-video diffusion models. Recent large-scale text-to-video diffusion models produce high-quality short clips with strong motion priors and text alignment [1], [2]. However, they are typically trained on fixed temporal horizons and offer limited *native* support for long-horizon, mask-based region edits, making naive extensions prone to drift and boundary artifacts.

Long video generation and extension. Training diffusion models directly on long videos is computationally expensive, so most text-to-video models are limited to short clips and suffer quality degradation when naively extended. Autoregressive approaches generate variable-length videos but often accumulate errors over time [8]–[19]. A large body of work instead explores *tuning-free* strategies to extend off-the-shelf diffusion models at inference time, including temporal co-denoising, noise rescheduling, and attention or latent blending [20]–[23]. Notably, Gen-L-Video [22] introduces a windowed temporal co-denoising framework, while FreeNoise [21] and DiTCtrl [23] improve long-range consistency via noise and attention manipulation. In this paper, we build on the windowed co-denoising paradigm and incorporate high-order integration together with Hamming-weighted blending to further reduce seam artifacts and temporal drift.

Diffusion-based video inpainting and outpainting. Classical video inpainting benchmarks (e.g., DAVIS, YouTube-VOS) assume pixel-level ground truth for missing regions and typically emphasize temporal propagation. More recent diffusion-based approaches incorporate text guidance and masked editing. VACE [24] provides a unified video creation/editing framework that supports masked edits, making it a strong baseline for our V2V setting. AVID [25] introduces a MultiDiffusion-style strategy to support any-length video inpainting, and Follow-Your-Canvas [26] and Be-Your-Outpainter [27] focus on large-extent/high-resolution outpainting via windowed fusion and input-specific adaptation. VideoPainter [28] enables plug-and-play mask-based video editing with pre-trained diffusion models. In contrast to methods that introduce specialized modules or focus on a single edit mode, we emphasize a lightweight *unified* pipeline that adapts a single T2V backbone to both interior-hole inpainting and border outpainting, while scaling to long videos through a sampler designed to suppress window seams and temporal drift.

Parameter-efficient adaptation with LoRA. Adapting billion-parameter diffusion backbones via full fine-tuning is costly. LoRA [29] enables parameter-efficient specialization by learning low-rank updates to frozen weights, and has become a standard tool for quickly adapting foundation models. We leverage LoRA to inject *mask-controllable* behavior into a pre-trained T2V model with minimal trainable parameters, enabling both inpainting and outpainting under a shared training objective.

High-order diffusion sampling. Diffusion sampling can be interpreted as numerically integrating a reverse-time process, where solver choice affects stability and error accumulation [30]. Higher-order updates such as Heun-style steps can improve sample quality per function evaluation and reduce discretization error [31]. Building on these insights, we integrate a two-stage Heun update into windowed temporal co-denoising and combine it with Hamming-weighted overlap blending to mitigate seam artifacts and long-horizon drift in edited videos.

III. Methodology

Our approach comprises three components: (1) LoRA-based spatial mask fine-tuning, (2) inference-time mask conditioning, and (3) long-horizon temporal co-denoising with a high-order solver. Together, these elements enable a single foundation model (Wan 2.1 14B) to perform both inpainting and outpainting within a unified framework. Figure 2 illustrates the overall pipeline.

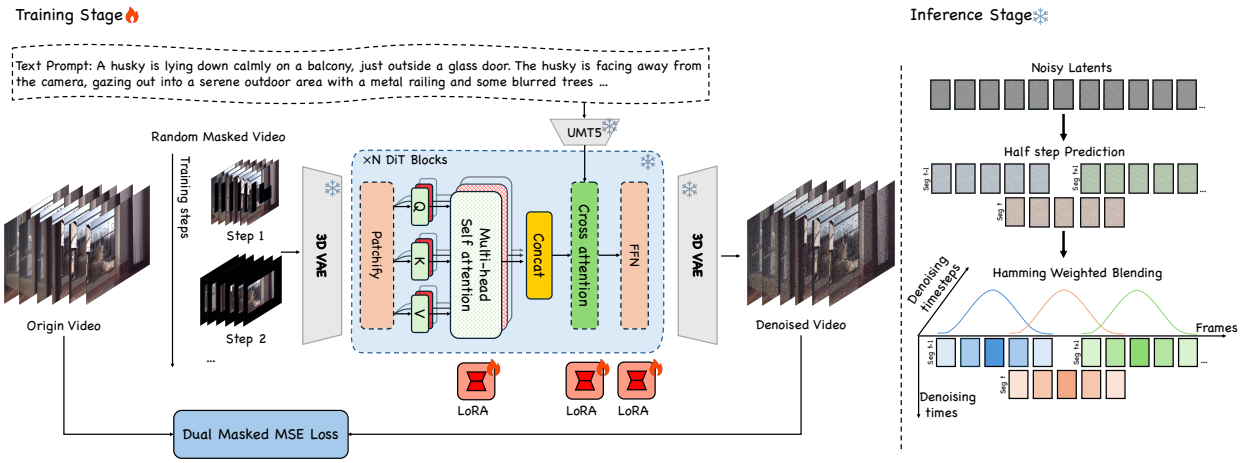
A. LoRA-based Spatial Mask Fine-Tuning

Although the original Wan model can perform basic video editing, its limited controllability makes more complex mask-guided edits difficult. To enable mask-conditioned generation without modifying the core DiT architecture, we inject LoRA adapters into self-attention, cross-attention, and feed-forward blocks. Concretely, for every weight matrix $W \in \mathbb{R}^{d\times k}$ in the frozen DiT, we learn a residual update:

$$\Delta W = BA, \quad B \in \mathbb{R}^{d\times r},\ A \in \mathbb{R}^{r\times k}, \tag{1}$$

where $r \ll \min(d, k)$ is the LoRA rank. The adapted weight is $W^* = W + \Delta W$, and thus we optimize only A and B during training time instead of the entire DiT blocks.

During each iteration, we sample a video clip $\{x_f\}_{f=1}^{T}$ and randomly apply one of two mask types to all frames: (1) *Border masks*, where pixels outside a central rectangle covering $\alpha \in [0.5, 0.8]$ of each spatial dimension are masked out to simulate outpainting; and (2) *Interior masks*, where $m \sim \mathrm{Uniform}\{1, 4\}$ rectangular regions within each frame are masked to simulate inpainting. Sampling both mask types



(a) Random Spatial Mask LoRA Finetuning (b) Temporal Co-Denoising

Fig. 2: **Overview.** We introduce a unified LoRA-based fine-tuning pipeline for both video inpainting and outpainting on InpaintBench. During training, each clip is randomly masked with either (i) *border masks*, which remove frame boundaries, or (ii) *interior masks*, which occlude regions inside the frame; a dual-region MSE loss then encourages accurate hole filling while preserving unmasked content. At inference, we partition long sequences into overlapping windows and perform temporal co-denoising using a two-stage Heun sampler with Hamming-weighted blending, yielding smoother long-video editing with reduced boundary artifacts.

during training enables a single model to learn both inpainting and outpainting within one unified framework.

We distinguish the video frame index $f \in \{1, \ldots, T\}$ from diffusion timesteps. Let $x_f \in \mathbb{R}^{H\times W\times 3}$ denote the ground-truth frame and $\hat{x}_f$ denote the reconstructed output frame after VAE decoding. Let $M_f \in \{0,1\}^{H\times W}$ be the binary mask for frame f, where $M_f(u,v) = 1$ indicates pixels to be synthesized and $M_f(u,v) = 0$ indicates known pixels.

We define masked-region and unmasked-region reconstruction losses as

$$L_{\text{masked}} = \sum_{f=1}^{T} \|M_f \odot (\hat{x}_f - x_f)\|_2^2, \tag{2}$$

$$L_{\text{unmasked}} = \sum_{f=1}^{T} \|(1 - M_f) \odot (\hat{x}_f - x_f)\|_2^2. \tag{3}$$

The final dual-region loss is a weighted sum:

$$L_{\text{Dual}} = \lambda\, L_{\text{masked}} + (1-\lambda)\, L_{\text{unmasked}}, \tag{4}$$

where $\lambda \in [0,1]$ balances hole-filling fidelity and context preservation. We empirically set $\lambda = 0.9$ in all experiments unless noted otherwise.

B. Inference-Time Mask Conditioning

At inference, we use the same masking operator as in training: known pixels are provided through masked latents, while the model synthesizes masked regions guided by the text prompt and LoRA adaptation. This preserves known content outside the edit region while allowing synthesis inside the mask.

a) Inpainting: Given an input video $\{x_f\}_{f=1}^{T}$ and a user-specified mask $\{M_f\}_{f=1}^{T}$, we form masked frames

$$\tilde{x}_f = (1 - M_f) \odot x_f, \qquad f = 1, \ldots, T, \tag{5}$$

and encode them with the Wan VAE encoder $E(\cdot)$:

$$z'_f = E(\tilde{x}_f). \tag{6}$$

The LoRA-adapted diffusion model then performs conditional denoising using $\{z'_f\}$ and the text prompt.

We do *not* explicitly clamp (re-impose) known pixels at each denoising step. Instead, preservation of unmasked content is encouraged by the dual-region loss (Sec. III-A), which penalizes deviations in $(1 - M_f)$ regions during fine-tuning. This avoids hard transitions that can arise from repeatedly overwriting known pixels during denoising.

b) Outpainting: To outpaint beyond the original field of view, we expand the latent canvas by padding. Let $P(\cdot)$ denote zero-padding to a user-specified spatial size. For each frame latent z_f, we form

$$z'_f = P(z_f), \tag{7}$$

and apply the same conditional diffusion process to synthesize content in the padded area. The padded latent canvas exposes new spatial regions for synthesis while retaining the original video content as context for coherent scene extension.

C. Arbitrary-Length Temporal Co-Denoising

Since Wan is trained on fixed short sequences, naively processing a longer video of $T \gg W$ frames incurs severe memory growth and temporal consistency issues. We therefore adopt a sliding window co-denoising strategy with adjustable overlap length O and Hamming-weighted blending at each diffusion time step t. The process is illustrated in Figure 2 (b). The key idea is to process a long video as overlapping temporal segments, denoise each segment locally, and then fuse the overlapping predictions smoothly.

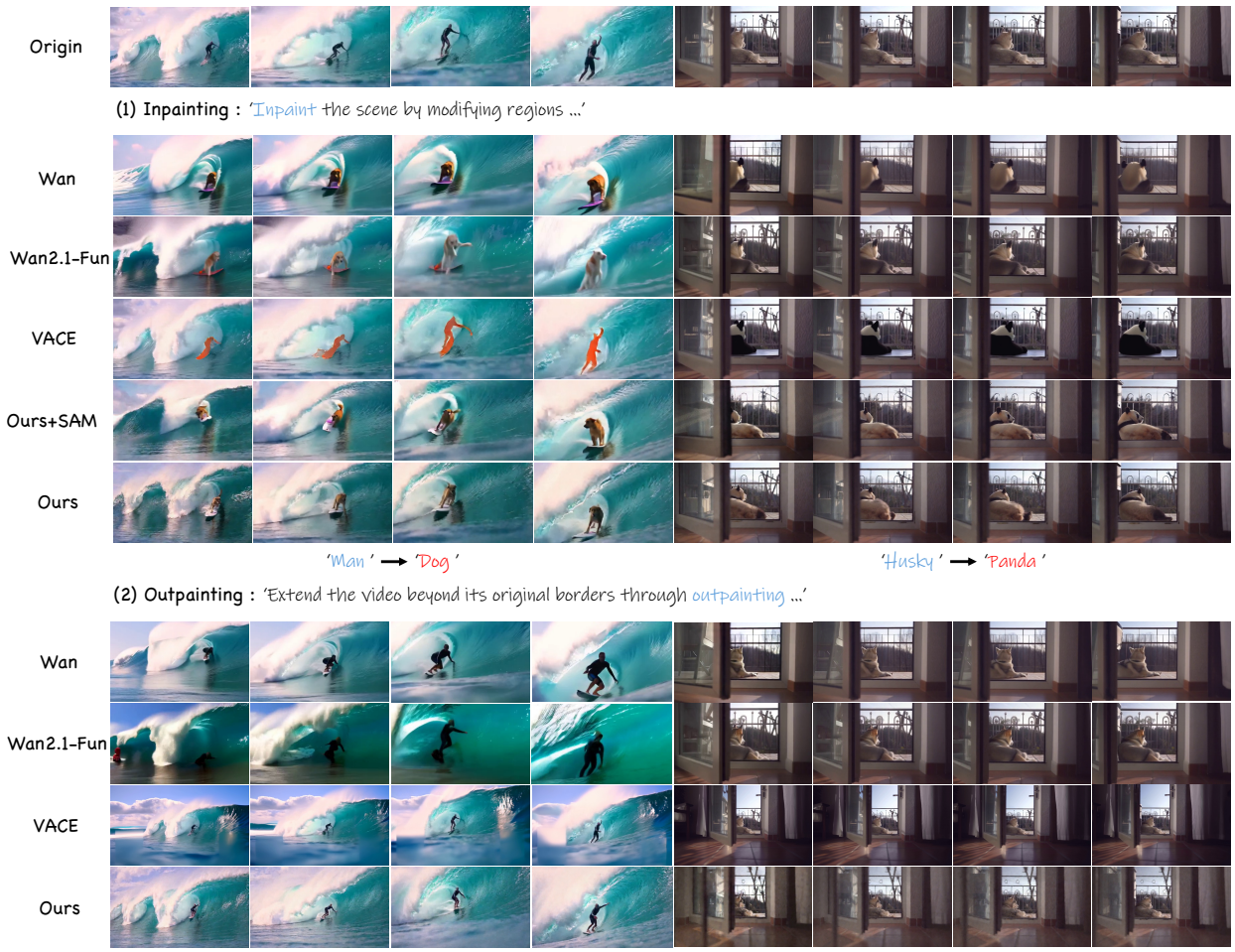


Fig. 3: **Qualitative results.** Inpainting (top) and outpainting (bottom) on long videos. Our method better preserves structure and motion while reducing boundary seams and temporal drift relative to Wan 2.1 variants and VACE.

a) Window extraction: Let $Z^{(n)} \in \mathbb{R}^{T\times d}$ denote the full latent sequence at diffusion step n (with $n = N$ the initial noisy latent and $n = 0$ the final latent). We extract overlapping windows using start indices

$$s_i = 1 + (i-1)(W-O), \qquad i = 1, \dots, I, \tag{8}$$

where

$$I = \max\left(1, \left\lceil \frac{T-W}{W-O} \right\rceil + 1\right). \tag{9}$$

The i-th window at step n is

$$Z_i^{(n)} = Z^{(n)}[\, s_i : s_i + W - 1\,] \in \mathbb{R}^{W\times d}. \tag{10}$$

b) Two-stage Heun update (within each window): Let $\{\sigma_n\}_{n=0}^{N}$ be the discrete noise schedule and $\Delta\sigma_n = \sigma_{n-1} - \sigma_n$. We denote by $\Phi(\cdot, \sigma)$ the diffusion-model-derived update direction (e.g., the ODE derivative in EDM-style sampling). In our implementation, $\Phi(\cdot, \sigma)$ is computed from the model's noise-prediction output under the standard EDM ODE formulation. For each window i, we perform a second-order Heun step:

$$K_1 = \Phi\left(Z_i^{(n)}, \sigma_n\right), \tag{11}$$

$$\tilde{Z}_i = Z_i^{(n)} + \Delta\sigma_n\, K_1, \tag{12}$$

$$K_2 = \Phi\left(\tilde{Z}_i, \sigma_{n-1}\right), \tag{13}$$

$$\hat{Z}_i^{(n-1)} = Z_i^{(n)} + \Delta\sigma_n\, \frac{K_1 + K_2}{2}. \tag{14}$$

This requires two model evaluations per step and reduces error accumulation over long sequences. Compared with a first-order update, Heun sampling reduces local discretization error and improves temporal stability.

c) Overlap blending with Hamming weights: We blend overlapping windows using a 1D Hamming window of length W:

$$w_j = \alpha - \beta \cos\left(\frac{2\pi(j-1)}{W-1}\right), \qquad j = 1, \dots, W, \tag{15}$$

with $\alpha = 0.54$ and $\beta = 0.46$. For each frame index $f \in \{1, \dots, T\}$, we aggregate all windows covering f:

$$Z^{(n-1)}[f] = \frac{\sum_{i:\, f\in[s_i,\, s_i+W-1]} w_{f-s_i+1}\, \hat{Z}_i^{(n-1)}[\, f - s_i + 1\,]}{\sum_{i:\, f\in[s_i,\, s_i+W-1]} w_{f-s_i+1}}. \tag{16}$$

Hamming weights down-weight window boundaries and emphasize segment centers, where predictions are typically more stable, reducing seams relative to naïve stitching or uniform averaging.

d) Complexity and Parallelism: At each diffusion step, we process windows sequentially (or in parallel if multiple devices are available) and accumulate their contributions into a global buffer. Memory is bounded by the window size W, and runtime scales linearly with video length T up to the constant factor from overlapping windows and the two-stage solver.

After all steps to $t = 0$, we decode X_0 using the VAE decoder to obtain the final video frames. Our ablation study (Section IV-D) confirms the importance of LoRA fine-tuning and Heun sampling, while qualitative comparisons suggest that Hamming blending further helps suppress boundary artifacts in long-video editing.

IV. Experiments

A. Datasets

Existing benchmarks such as DAVIS [3] and YouTube-VOS [4] focus on short clips. To evaluate long-form editing, we construct **InpaintBench**, a collection of 30 publicly available real-world videos spanning human actions, animals, outdoor scenes, and indoor environments, with lengths from 81 to 300 frames. For each video, we provide a base caption, an *inpainting* prompt, an *outpainting* prompt, and a mask specification used consistently across methods. We release the videos, prompts, and mask generators for reproducible evaluation. Despite its modest size, the benchmark covers diverse content and long-horizon editing scenarios underrepresented in short-clip benchmarks.

B. Implementation Details

Our method is built on the official DiffSynth-Studio [32] codebase and the publicly available 14B-parameter Wan 2.1 text-to-video model. We inject LoRA adapters (rank 16) into all self-attention layers, cross-attention layers, and feed-forward sublayers. Training is performed on a single NVIDIA H100 GPU using the AdamW optimizer with a fixed learning rate of 1×10^{-4}. We train for 2000 steps on a curated set of 25 video clips spanning human activities, animals, indoor scenes, and outdoor scenes. Each sample is temporally resampled to 81 frames at 416×240. Border and interior masks are generated on the fly to simulate outpainting and inpainting. Training takes about one hour. Text prompts are generated with ChatGPT, then manually reviewed and refined before pairing with each clip. At inference, we adopt a two-stage Heun solver and Hamming-weighted blending, together with classifier-free guidance, for text-guided video-to-video generation.

C. Evaluation Setup

Tasks. We evaluate two editing tasks: (1) Object inpainting: adding or replacing objects within videos, and (2) Scene outpainting: extending content beyond original frame boundaries.

Baselines. We compare against Wan 2.1 14B [2], Wan2.1-Fun-Control [33], and VACE [24], which represent strong recent text-to-video and video-to-video diffusion systems supporting conditional generation. While systems like AVID [25] and VideoPainter [28] address video completion, they rely on architectures or inference mechanisms that differ substantially from our lightweight LoRA-adaptation setting. In contrast, our method creates a lightweight, trainable adapter that is structurally simpler and more scalable. We compare against VACE and Wan-Control as they represent the state-of-the-art in tuning-based and control-based foundation models, which are the direct counterparts to our proposed LoRA approach. Unless explicitly modified, each baseline uses its default inference sampler and schedule from the corresponding public implementation. The effect of our two-stage Heun solver is isolated separately in the ablation study.

Evaluation Protocol. In our object replacement and scene extension tasks, edited regions have no pixel-level ground truth, so we separate *background preservation* from *edit realism/alignment*. We therefore evaluate edited content using prompt alignment and temporal consistency rather than pixel reconstruction. (1) **Background preservation / leakage:** we compute SSIM (Structural Similarity) [6] and LPIPS (Learned Perceptual Image Patch Similarity) [7] *only on the unmasked region* $(1 - M)$ between the input and edited videos, which directly measures whether methods inadvertently alter known pixels outside the target area. (2) **Prompt alignment in edited regions:** we report CLIP (Contrastive Language–Image Pre-training) similarity [5] between the text prompt and generated frames, computed on the *masked region crop* (bounding box of M) to focus on the intended edit rather than the preserved background.[1] (3) **Temporal stability:** we report tLPIPS as the average LPIPS between consecutive generated frames, which captures flicker and short-term inconsistency (lower is better).

We report metrics averaged over videos and frames; for Table I we average each metric across both tasks with equal weight unless stated otherwise. This protocol avoids invalid pixel metrics inside the edited region while still quantifying (i) faithfulness to preserved content, (ii) semantic alignment of the synthesized content, and (iii) temporal smoothness.

Qualitative results. Figure 3 shows representative inpainting and outpainting examples (77–181 frames). Compared with Wan 2.1 variants and VACE, our method better maintains motion continuity, reduces boundary seams, and produces more coherent prompt-aligned edits. See the supplementary videos for full temporal comparisons.

D. Ablation Studies

We conduct ablations to measure the contribution of each component and the selection of hyperparameters.

*a) **Selection of** λ **in** L_{Dual}:* We evaluated $\lambda \in \{0.1, 0.5, 0.9, 1.0\}$ and report the results in Table II. Increasing λ places greater emphasis on masked-region reconstruction,

[1] If the masked region is disconnected (multi-rectangle), we take the tight union bounding box; details in code.

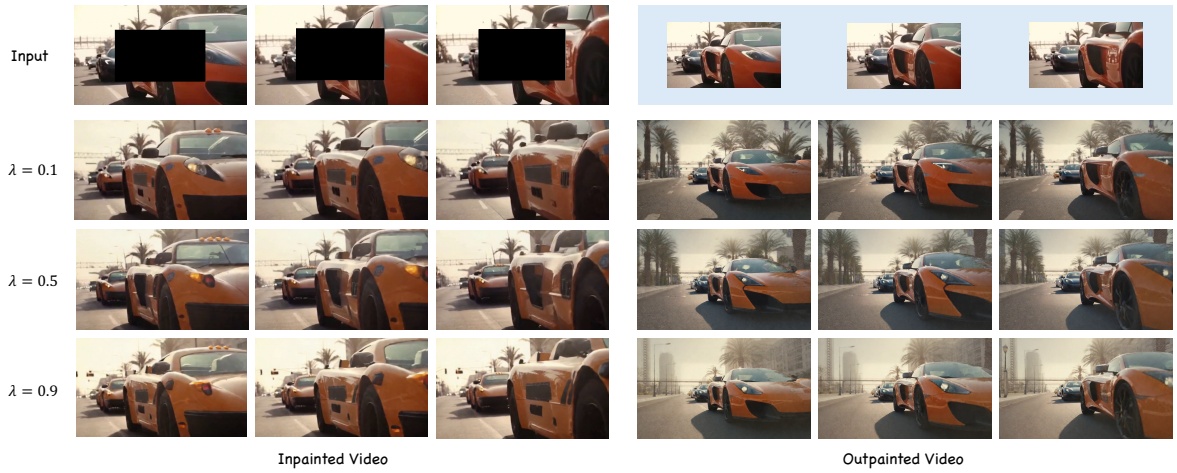


Fig. 4: **Effect of λ in L_{Dual}.** Smaller λ preserves context but under-fills; larger λ improves completion but may perturb known regions. λ=0.9 gives the best trade-off.

Method	CLIP↑	SSIM↑	LPIPS↓	tLPIPS↓
Wan2.1 14B	29.2	0.563	0.130	0.225
Wan2.1-Fun-Control 14B	27.1	0.589	0.162	0.228
VACE	28.4	0.612	0.125	0.227
Ours	**29.4**	**0.613**	**0.121**	**0.224**

TABLE I: Quantitative comparisons on InpaintBench. Results are reported as [mean ± standard deviation across videos / mean with 95% bootstrap confidence intervals]. ↑ indicates higher is better and ↓ indicates lower is better. SSIM and LPIPS are computed on the unmasked region $(1-M)$ for both inpainting and outpainting, measuring background preservation. CLIP evaluates prompt alignment of edited regions, and tLPIPS measures temporal consistency.

	Inpainting		Outpainting	
λ	CLIP↑	SSIM↑	CLIP↑	SSIM↑
0.1	23.4	0.695	22.9	0.672
0.5	26.7	0.661	25.8	0.645
0.9	29.8	0.636	28.1	0.592
1.0	29.6	0.628	28.0	0.587

TABLE II: L_{masked} weight λ impact on performance

which improves hole filling but introduces slightly larger deviations in the unmasked areas. The best balance of CLIP and SSIM is achieved at $\lambda = 0.9$, as confirmed by both quantitative metrics and the qualitative examples in Figure 4. In practice, λ can be tuned along with other hyperparameters to match specific application requirements.

Arbitrary-length Temporal Co-denoising. To assess the limitations of other approaches that encode and generate entire videos simultaneously, we conducted experiments to test the upper limit of video lengths on 1 NVIDIA H100 80GB GPU. We encounter out-of-memory errors when the generation video of the same size is longer than the frames' upper limit.

Two-stage Heun solver. Table IV compares video generation with and without our two-stage Heun solver. Incorporating Heun's method raises SSIM, reduces LPIPS and tLPIPS. These

Method	Max number of frames
VACE	fixed 81
Wan 2.1 14B	max 245
Ours	bounded by window size; demonstrated up to 800 frames

TABLE III: Supported maximum video length at 1600×800 resolution.

gains demonstrate that the high-order solver substantially improves both reconstruction fidelity, perceptual quality and temporal consistency.

Method	SSIM↑	LPIPS↓	tLPIPS↓
Without two-stage Heun solver	0.586	0.135	0.238
With two-stage Heun solver	**0.603**	**0.121**	**0.224**

TABLE IV: Effect of two-stage Heun solver

Effect of Window Length. We experimented with shorter window lengths (e.g. 50 frames) processed by the model by artificially limiting the temporal attention. Shorter windows mean more frequent blending, which could accumulate error but also could refresh context. We found 80–100 frame windows to be optimal for our model; very short windows (30 frames) hurt global consistency (some long-term context was lost). Using the model's suggestion (81) was a safe choice.

These ablations confirm that each design choice contributes to the robust performance of our system.

V. Limitations

Our approach inherits limitations of the underlying T2V backbone: extremely long-range identity persistence and fine-grained geometry may still drift under large edits or very high-motion scenes, and the two-stage solver roughly doubles per-step compute compared to first-order sampling. In practice, the improved stability allows fewer total diffusion steps, partially offsetting this cost. In addition, because edited regions lack ground-truth, our quantitative protocol focuses on background

preservation, prompt alignment, and temporal stability; richer human preference studies are left to future work.

VI. Conclusion

We presented a unified framework to achieve long-form video inpainting and outpainting by combining LoRA-based fine-tuning with an overlapping high-order diffusion sampling strategy. Starting from the Wan 2.1 foundation model, we turned it into a flexible, high-quality video editor capable of filling in or extending content over hundreds of frames. Through the dual-region loss and mask conditioning, the LoRA adaptation preserves the original content while seamlessly painting missing regions – all without modifying the original model architecture. Through overlapping window denoising and second-order solver integration, we scale the generation to substantially longer durations with smoother transitions and reduced boundary artifacts between segments. Our experiments show improved fidelity, perceptual quality, and temporal stability over strong baselines on long-form inpainting and outpainting. Beyond these tasks, the same LoRA adaptation and overlapping high-order sampling strategy can be applied to other mask-guided or control-guided video edits with minimal additional training.

References

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

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

[3] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] N. Xu, L. Yang, Y. Fan, D. Yue, Y. Liang, J. Yang, and T. Huang, "Youtube-vos: A large-scale video object segmentation benchmark," *arXiv preprint arXiv:1809.03327*, 2018.

[5] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[6] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

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

[8] Y. He, T. Yang, Y. Zhang, Y. Shan, and Q. Chen, "Latent video diffusion models for high-fidelity long video generation," *arXiv preprint arXiv:2211.13221*, 2022.

[9] Y. Ma, Y. He, X. Cun, X. Wang, S. Chen, X. Li, and Q. Chen, "Follow your pose: Pose-guided text-to-video generation using pose-free videos," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 5, 2024, pp. 4117–4125.

[10] K. Feng, Y. Ma, B. Wang, C. Qi, H. Chen, Q. Chen, and Z. Wang, "Dit4edit: Diffusion transformer for image editing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 3, 2025, pp. 2969–2977.

[11] Y. Ma, K. Feng, X. Zhang, H. Liu, D. J. Zhang, J. Xing, Y. Zhang, A. Yang, Z. Wang, and Q. Chen, "Follow-your-creation: Empowering 4d creation through video inpainting," *arXiv preprint arXiv:2506.04590*, 2025.

[12] Y. Shen, J. Yuan, T. Aonishi, H. Nakayama, and Y. Ma, "Follow-your-preference: Towards preference-aligned image inpainting," *arXiv preprint arXiv:2509.23082*, 2025.

[13] Z. Long, M. Zheng, K. Feng, X. Zhang, H. Liu, H. Yang, L. Zhang, Q. Chen, and Y. Ma, "Follow-your-shape: Shape-aware image editing via trajectory-guided region control," *arXiv preprint arXiv:2508.08134*, 2025.

[14] Y. Ma, Z. Wang, T. Ren, M. Zheng, H. Liu, J. Guo, M. Fong, Y. Xue, Z. Zhao, K. Schindler *et al.*, "Fastvmt: Eliminating redundancy in video motion transfer," *arXiv preprint arXiv:2602.05551*, 2026.

[15] Y. Ma, Y. Liu, Q. Zhu, A. Yang, K. Feng, X. Zhang, Z. Li, S. Han, C. Qi, and Q. Chen, "Follow-your-motion: Video motion transfer via efficient spatial-temporal decoupled finetuning," *arXiv preprint arXiv:2506.05207*, 2025.

[16] Y. Ma, K. Feng, Z. Hu, X. Wang, Y. Wang, M. Zheng, X. He, C. Zhu, H. Liu, Y. He *et al.*, "Controllable video generation: A survey," *arXiv preprint arXiv:2507.16869*, 2025.

[17] Y. Ma, X. Wang, Q. Ma, Q. Wang, M. Zheng, X. Yang, H. Li, C. Zhao, J. Ying, H. Yang *et al.*, "Group editing: Edit multiple images in one go," *arXiv preprint arXiv:2603.22883*, 2026.

[18] R. Henschel, L. Khachatryan, H. Poghosyan, D. Hayrapetyan, V. Tadevosyan, Z. Wang, S. Navasardyan, and H. Shi, "Streamingt2v: Consistent, dynamic, and extendable long video generation from text," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 2568–2577.

[19] R. Villegas, M. Babaeizadeh, P.-J. Kindermans, H. Moraldo, H. Zhang, M. T. Saffar, S. Castro, J. Kunze, and D. Erhan, "Phenaki: Variable length video generation from open domain textual description," *arXiv preprint arXiv:2210.02399*, 2022.

[20] J. Kim, J. Kang, J. Choi, and B. Han, "Fifo-diffusion: Generating infinite videos from text without training," *Advances in Neural Information Processing Systems*, vol. 37, pp. 89 834–89 868, 2024.

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

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

[23] M. Cai, X. Cun, X. Li, W. Liu, Z. Zhang, Y. Zhang, Y. Shan, and X. Yue, "Ditctrl: Exploring attention control in multi-modal diffusion transformer for tuning-free multi-prompt longer video generation," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 7763–7772.

[24] Z. Jiang, Z. Han, C. Mao, J. Zhang, Y. Pan, and Y. Liu, "Vace: All-in-one video creation and editing," *arXiv preprint arXiv:2503.07598*, 2025.

[25] Z. Zhang, B. Wu, X. Wang, Y. Luo, L. Zhang, Y. Zhao, P. Vajda, D. Metaxas, and L. Yu, "Avid: Any-length video inpainting with diffusion model," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 7162–7172.

[26] Q. Chen, Y. Ma, H. Wang, J. Yuan, W. Zhao, Q. Tian, H. Wang, S. Min, Q. Chen, and W. Liu, "Follow-your-canvas: Higher-resolution video outpainting with extensive content generation," *arXiv preprint arXiv:2409.01055*, 2024.

[27] F.-Y. Wang, X. Wu, Z. Huang, X. Shi, D. Shen, G. Song, Y. Liu, and H. Li, "Be-your-outpainter: Mastering video outpainting through input-specific adaptation," in *European Conference on Computer Vision*. Springer, 2024, pp. 153–168.

[28] Y. Bian, Z. Zhang, X. Ju, M. Cao, L. Xie, Y. Shan, and Q. Xu, "Videopainter: Any-length video inpainting and editing with plug-and-play context control," in *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, 2025, pp. 1–12.

[29] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

[30] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[31] T. Karras, M. Aittala, T. Aila, and S. Laine, "Elucidating the design space of diffusion-based generative models," *Advances in neural information processing systems*, vol. 35, pp. 26 565–26 577, 2022.

[32] M. Team, "Diffsynth-studio," https://github.com/modelscope/DiffSynth-Studio, 2025, accessed: 2025-08-02.

[33] Alibaba PAI, "Wan2.1-Fun-14B-Control," https://huggingface.co/alibaba-pai/Wan2.1-Fun-14B-Control, Aug. 2025, accessed: 2025-08-02.