LongSCOP: Semantically Consistent Long Video Outpainting

Yu Tang[13*], **Zhiyuan You**[23*], **Zhixiao Wang**[1], **Fanghua Yu**[4], **Qingyu Zhang**[1], **Xiang Yin**[3], **Chao Dong**[35†], **Jinjin Gu**[6†]
[1]Nanjing University [2]The Chinese University of Hong Kong
[3]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
[4]The University of Hong Kong [5]Shenzhen University of Advanced Technology
[6]INSAIT, Sofia University "St. Kliment Ohridski"
* Equal Contribution † Corresponding Author

Abstract

Video out-painting is an important problem, yet existing research focuses primarily on short clips, leaving long video out-painting largely underexplored. Extending video out-painting to long videos poses substantial challenges. A natural solution is iterative out-painting, *i.e.*, generating videos chunk by chunk, but this introduces two major issues: generation instability or even failure caused by inter-chunk inconsistencies and error accumulation, and semantic drift over long temporal horizons. In this paper, we propose **LongSCOP**, short for **S**emantically **C**onsistent long Video **O**ut-**P**ainting. First, we introduce a training recipe with three components for smooth and stable long-horizon generation: self-forcing training, latent calibration, and future references. Beyond generation stability, we recognize that the rich temporal context of long videos provides valuable references for maintaining semantic consistency. We therefore develop a VLM-based agent that automatically selects the most informative reference frames from the entire video to maintain consistent subjects, scenes, and backgrounds throughout long-horizon generation. We also construct a high-quality training dataset of 62K video clips. To facilitate research on this underexplored problem, we introduce **LVO-Bench**, a benchmark curated by human experts to assess long-horizon stability and semantic consistency. Extensive experiments demonstrate that our method outperforms existing approaches in both generation stability and semantic coherence.

1 Introduction

Long video out-painting is an important yet underexplored problem. Video out-painting aims to expand the spatial field of view of a video beyond its originally observed boundaries [38], and has many applications [48, 8]. Benefiting from the rapid progress of video generation models [27, 37], recent video out-painting methods have achieved encouraging results in synthesizing visually realistic and temporally smooth content [50, 7]. Meanwhile, extending out-painting from short clips (*e.g.*, a few seconds) to substantially longer videos (*e.g.*, 20 seconds or more) is increasingly important in real-world usage. Such long-form content has become a dominant format on modern video platforms [11], and it usually contains richer narratives and more dynamic scene transitions compared to short clips; thus, it is more valuable for practical applications [48].

Despite its practical importance, long video out-painting remains highly challenging. A direct solution would be to process an entire long video jointly. However, due to the high computational and memory cost of video generation models, handling long sequences in a single pass is often impractical. A more natural and scalable alternative is iterative out-painting, *i.e.*, dividing the video into temporal chunks and generating them sequentially [50]. This strategy enables existing short-video models to be extended to longer videos without fundamentally changing their architectures.

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



Figure 1: Illustration of the two main challenges in long video out-painting: (a) unstable generation including abrupt transitions, brightness flickering, and even complete generation collapse caused by accumulated errors, and (b) semantic drift like changed scene layouts, unstable object identities, or inconsistent subject appearances over long time. Both cases are generated by VACE model [22].

However, simply applying chunk-by-chunk generation introduces two major challenges over long horizons. First, consistency across neighboring chunks is difficult to preserve, often causing abrupt transitions, brightness flickering, and progressively accumulated errors that may eventually lead to complete generation collapse. As shown in Fig. 1 (a), the out-painting on the left and right sides gradually collapses due to accumulated errors, resulting in excessive color spill and completely abstract objects. Second, as the temporal horizon grows longer, the generated content tends to drift semantically, resulting in changed scene layouts, unstable object identities, or inconsistent subject appearances over time. For example, in Fig. 1 (b), the model initially out-painted a human hand, but the hand gradually transforms into a lizard body over a long period.

To address the above problems, we introduce **LongSCOP**, a **S**emantically **C**onsistent long video **O**ut-**P**ainting method, tackling long video out-painting from two aspects: stable long-horizon generation and semantic consistency preservation.

One fundamental requirement for long video out-painting is to maintain smooth and stable long-horizon generation, avoiding abrupt transitions, brightness flickering, and generation collapse (*i.e.*, caused by accumulated errors). To this end, we propose a long-horizon stabilization strategy for iterative out-painting from three aspects. First, we adopt a self-forcing training scheme [21], where the model uses its own predictions as references during training, so that the training process is better aligned with inference-time iterative generation. Second, we introduce latent calibration by normalizing the predicted latents, which constrains their distribution from drifting too far away and thus reduces the risk of error accumulation and generation collapse. Third, instead of relying only on the out-painted results from the previous chunk as a reference, we further incorporate future references to provide richer temporal guidance and improve the stability of long-range generation.

Another, more advanced requirement is semantic consistency, *i.e.*, subjects, scenes, and backgrounds should remain consistent over long temporal horizons. Even with smooth and stable long-horizon generation, individual subjects may gradually and seamlessly transform into entirely different ones as generation proceeds, as illustrated in Fig. 1 (b). To address this issue, we recognize that the rich context of a long video provides a valuable source of visual references for maintaining semantic consistency. While temporally adjacent reference frames help maintain local continuity and smooth transitions between consecutive chunks, such local references become insufficient over long temporal horizons. They mainly provide short-range motion and appearance cues, but often fail to preserve subject identity and scene semantics across distant chunks. We therefore build an agent based on Vision Language Models (VLMs) [2, 28, 32, 39, 5, 52, 14] that searches the entire video and automatically selects reference frames with the most informative semantic cues, providing global guidance for semantically consistent long-horizon generation.

To train and evaluate our LongSCOP, we collect a large-scale, high-quality training dataset and carefully construct a benchmark specifically for long-video out-painting. Our training dataset covers a wide range of video genres, consisting of 62K video clips, and ensures high-quality and diverse content that is essential for robust training across complex, dynamic scenes. The LVO-Bench benchmark, curated by human experts, includes representative video clips selected for their complexity and

diversity, providing a comprehensive evaluation platform for testing the model's ability to maintain semantic consistency and temporal coherence in challenging long-video generation scenarios.

Extensive experimental results show the performance advantages of our LongSCOP. Specifically, in the LVO-Bench benchmark, we improve the FVD [36] by 28.19 points and PSNR by 0.92 dB compared to the previous best model. The ablation experiments show the effectiveness of our proposed two main modules and the design choice of our key reference frame retrieval method.

2 Related Works

Video Generation Models. Early video diffusion models extend image diffusion architectures to the temporal domain by introducing temporal attention [42], 3D convolutions, or latent video representations, enabling the synthesis of visually realistic and temporally coherent short clips [19, 3, 51]. Subsequent works improve generation quality and controllability by scaling training data [6], model capacity, and conditioning signals, leading to increasingly capable text-to-video and image-to-video systems [18, 34, 16]. Recently, open large-scale video generation models such as Open-Sora Plan [27] and Wan [37] demonstrate strong generalization across diverse video creation tasks, and provide powerful foundation models for downstream video editing and completion. In addition, to overcome the challenges of long-video generation and reduce the gap between training and inference, recent methods have explored auto-regressive generation, such as Diffusion Forcing [4], Self Forcing [21], and Rolling Forcing [29]. These methods have been shown to be effective in improving long-horizon video generation.

Video Out-painting Video out-painting seeks to extend the spatial contents of a video beyond its initial boundaries. Existing methods usually focus on temporal consistency or stronger conditioning. Complete and Temporally Consistent Video Out-painting builds background estimates, uses optical flow for temporal consistency, and introduces image shifting to improve frame completion [8]. M3DDM is particularly relevant to our setting: it uses multiple guide frames to connect clip-level inference and extracts global frames of the video as prompts via cross-attention [12]. More recent out-painting methods strengthen guidance in other ways. MOTIA performs input-specific adaptation to learn source-video patterns [38]. Infinite-Canvas injects the source video and its relative positional relations into each spatial window [7]; and Unboxed combines 3D-supported out-painting for static regions with dynamic object completion and guided video synthesis [48]. These methods show that stronger guidance beyond the current frame is beneficial, but they still do not formulate reference selection as a query-dependent retrieval problem over the full video timeline. Moreover, VACE [22] as a general video editing method could achieve video out-painting through specific mask design.

Reference-driven Video Completion Recent work has also explored richer reference signals for video completion [53]. HomoGen [9] propagates contextual pixels via homography and injects them as priors into a video diffusion model. VideoRepainter [17] uses a user-edited key frame as an explicit reference and propagates its semantics to the remaining frames. These methods confirm the value of strong external guidance, but the guidance is either geometrically propagated or manually specified, rather than automatically retrieved from the source video according to the semantic needs.

3 Methods

3.1 Preliminaries

VACE Model. VACE [22] is a general video editing model with unified support for various video editing tasks. The inputs of VACE are termed as video condition units, which could be expressed as:

$$I = \{T, V_{in}, M\}, \tag{1}$$

where T is the textual prompt, V_{in} is the video sequence $V_{in} = \{v_0, v_1, ..., v_{n-1}\}$ with each $v_i \in \mathbb{R}^{H\times W\times 3}$, and M is the mask sequence $M = \{m_0, m_1, ..., m_{n-1}\}$ with each $m_i \in \{0, 1\}^{H\times W}$. In M, 0 and 1 indicate the pixels that should be maintained and edited, respectively. The VACE model f takes $I = \{T, V_{in}, M\}$ as inputs and then outputs an edited video V_{out} as:

$$V_{out} = f(I = \{T, V_{in}, M\}). \tag{2}$$

VACE is trained on extensive and large-scale datasets, thus could generalize to different masks very well. By designing different masks, VACE could edit regions at various scales and positions.

VACE for Video Out-painting We could design inputs $I = \{T, V_{in}, M\}$ to achieve video out-painting using the VACE model. For input video V_{in}, we zero-pad the original video to the out-painting target resolution, thus the input video V_{in} will have black borders as in Fig. 2. For masks M, we assign the mask values of the original region as 0, meaning these pixels should be kept. The mask values of the out-painted region (*i.e.*, the black border) are assigned as 1, indicating that they should be generated. The textual prompts T are obtained by VLM captioning on the original video. With above designed inputs, VACE could serve as a baseline model for video out-painting.

VACE Out-painting with Reference Frames If we have an out-painted frame with the target resolution, an intuitive idea is to treat the out-painted frame as the reference frame to guide a smooth and continuous editing. We could achieve this simply by appending an out-painted frame before the input video V_{in} and an all 0 mask before the masks M, since the all 0 mask means that all pixels should be kept and unchanged. Multiple reference frames could be added in the same way.

VACE Chunk-by-Chunk Out-painting With the above preliminaries, we could implement a chunk-by-chunk video out-painting pipeline. For a long input video, we first divide it into several consecutive chunks and apply VACE out-painting to each chunk in temporal order. The generated results from previous chunks are then used as reference frames for subsequent chunks, forming a simple chunk-by-chunk baseline for long video out-painting.

3.2 Foundation: Smooth and Stable Long-Horizon Generation

A fundamental requirement for long video out-painting is to ensure smooth and stable long-horizon generation, avoiding abrupt transitions, brightness flickering, and generation collapse caused by accumulated errors. However, as the generation proceeds over time, the out-painted regions gradually degrade due to error accumulation, leading to severe color spillover and distorted structures (See Fig. 1(a)). To address this, we propose a long-horizon stabilization strategy for iterative out-painting.

Self-forcing Training with Model-predicted References Chunk-by-chunk video out-painting requires auto-regressive generation across multiple chunks, making training-inference mismatch a major source of error accumulation. During training, if the model always conditions on ground-truth history frames, it only observes clean reference inputs. At inference time, however, later chunks depend on previously generated outputs, which inevitably contain noise and bias. This discrepancy grows over time and leads to unstable long-horizon generation.

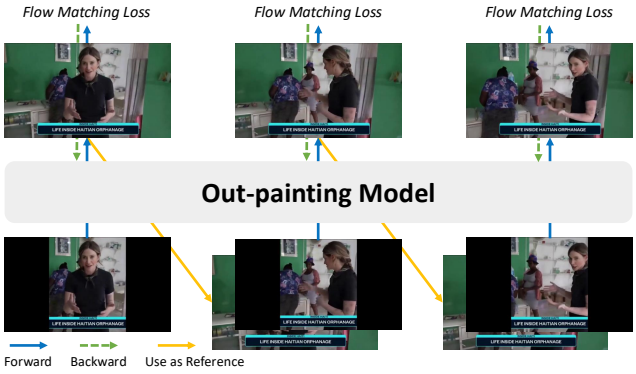


Figure 2: Illustration of self-forcing training. During training, we use the model-predicted frames as reference frames, which could reduce the gap between training and inference.

To reduce this gap, inspired by [21], as depicted in Fig. 2, we introduce a self-forcing training strategy. During training, we construct two consecutive chunks. The model first performs online out-painting on the first chunk and produces predicted latent representations. These latents are then decoded into pixel space through the VAE decoder, and the last several generated frames are pre-pended to the second chunk as prefix conditions. Importantly, these prefix frames are not out-painted again; instead, they serve only as temporal conditioning signals. By exposing the model to its own predictions during training, self-forcing better aligns training with auto-regressive inference and improves long-range stability.

Latent Calibration A second source of error accumulation comes from the latent distribution shift over time. In long-video chunk-wise inference, generated frames must be repeatedly written back and re-encoded into latent space. Such repeated operations can progressively distort latent statistics, causing degradation in brightness, saturation, contrast, and texture fidelity.

To address this issue, we introduce latent calibration. We treat the latent statistics of several real frames from the first chunk as an anchor distribution and compute their mean μ_{real} and variance σ^2_{real}. During the out-painting process of next chunk, we align the reference latent z_{raw} to the anchor

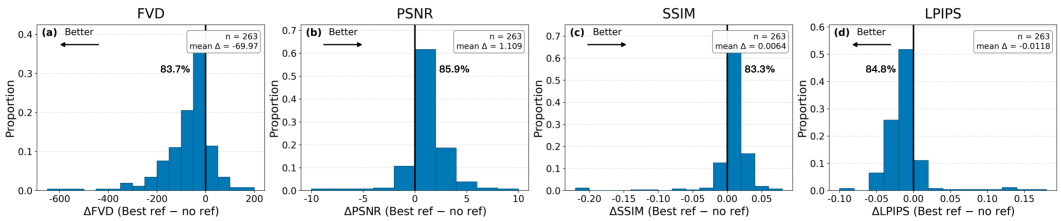


Figure 3: Illustration of the importance of suitable reference frame. Adding only one reference frame could significantly improve the out-painting performance of the whole video by nearly 70 FVD points and more than 1.1 dB PSNR on average.

distribution to obtain the final reference latent z as:

$$\hat{z} = \left(\frac{z_{\text{raw}} - \mu_{\text{raw}}}{\sigma_{\text{raw}} + \epsilon} \right) \sigma_{\text{real}} + \mu_{\text{real}}, \qquad z = \alpha \hat{z} + (1 - \alpha) z_{\text{raw}}, \tag{3}$$

where μ_{raw} and σ_{raw} are the mean and standard deviation of the raw reference latent z_{raw}, respectively, and ϵ is a small constant 10^{-6} to prevent numerical instability caused by division by zero, α is a weighted hyper-parameter and is set as 0.9. In this way, the latent space remains close to the original real-video statistics throughout long-horizon generation, which reduces long-term quality degradation and lowers the risk of generation collapse. Note that the latent calibration is conducted during both self-forcing training and inference process.

Bidirectional Reference In standard chunk-wise inference, the current chunk usually conditions only on past frames. Although such one-sided conditioning helps maintain local temporal smoothness, it does not provide any information about future content. As a result, the content synthesized in the current chunk may become inconsistent with future chunks, leading to flickering and semantic discontinuities near chunk boundaries.

We therefore introduce bidirectional reference chunk inference. When out-painting the current chunk, we not only condition on historical frames but also include a small set of future frames as auxiliary references. The future reference could be implemented by a quite simple inference trick **without the need of retraining**. Specifically, during inference, we increase the input video length to cover a longer time than the current chunk. After out-painting, we keep only the outputs corresponding to the current chunk and discard the results of future frames. Although simple, this strategy allows the model to access a broader temporal context and anticipate scene evolution during generation, effectively reducing boundary inconsistency.

3.3 Advance: Semantic Consistency via Key Reference Frame Retrieval

Even with smooth and stable long-horizon generation, some subjects may gradually drift into totally different ones as generation proceeds, as shown in Fig. 1(b). Therefore, a more advanced requirement is semantic consistency, *i.e.*, the subjects, scenes, and backgrounds should remain consistent over long temporal horizons. Here we show that good reference frames could provide semantically consistent information, and design a method to automatically select the most suitable reference frames.

What a Good Reference Can Do We restrict reference frames to real frames sampled from the input video, rather than using externally generated images. This design has two advantages. First, it avoids introducing synthetic artifacts or additional distribution mismatch. Second, these frames naturally share the same subjects, scene layout, and visual style as the target video, making them more suitable as conditioning signals for out-painting. For long videos, relying only on temporally adjacent history frames provides local continuity but is often insufficient for preserving global semantics over long horizons. In contrast, well-chosen reference frames can provide complementary structural cues, appearance information, and contextual semantics, thereby mitigating facial distortion, background inconsistency, and semantic drift across chunks.

We demonstrate the effectiveness of reference frames on LVO-Bench benchmark by comparing each sample with and without reference frames across multiple metrics in Fig. 3. Here the reference frames are sampled from the input video at one frame per second. We report the gains obtained by the best reference frames of each sample. As shown in Fig. 3, most samples (around 83% to 86%) achieve better results with reference frames than without them across all metrics. In particular, adding only one reference frame could significantly improve the out-painting performance of the whole video by nearly 70 FVD and more than 1.1 dB PSNR on average, indicating a substantial gain in video quality.

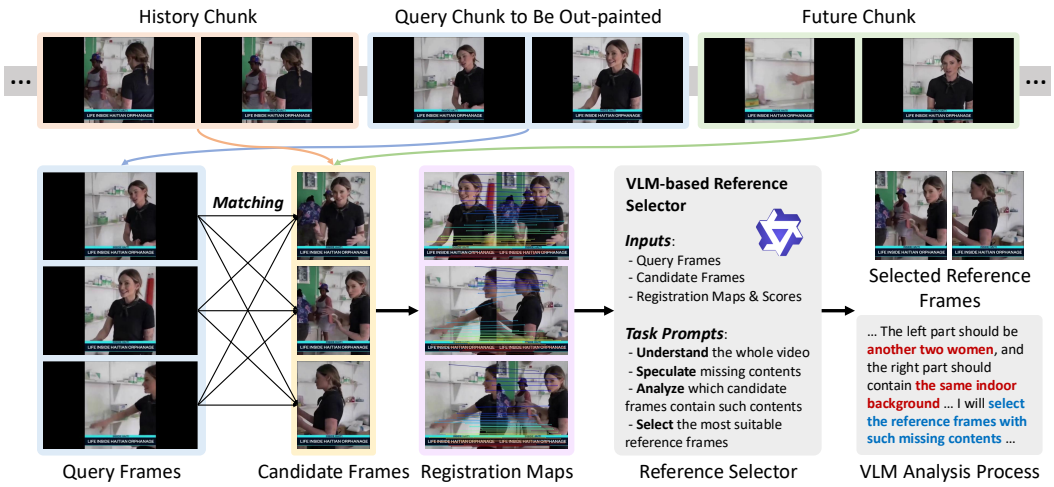


Figure 4: Illustration of key reference frame retrieval framework. Given query frames from current query chunk to be out-painted and candidate frames from both history chunk and future chunk, we aim to find the most suitable reference frames to out-paint current chunk. Specifically, we calculate the registration maps and scores for all query-candidate pairs, and then input them into a VLM, Qwen3.5-VL-27B [32]. The VLM is requested to understand the whole video, speculate the missing contents (*i.e.*, contents to out-paint like the two women and the indoor background), analyze which candidate frames may contain such missing contents, finally select the most suitable reference frames.

What Makes a Good Reference Frame Our analysis suggests that a good reference frame should satisfy two key properties: **appropriate spatial overlap** and **information usefulness**. Spatial overlap determines whether the candidate frame can be reliably aligned with the query chunk, while information usefulness indicates whether the candidate provides content that is truly helpful for recovering the missing out-painting region. These two properties are complementary: the former ensures usable geometric correspondence, and the latter ensures effective contextual guidance.

Spatial overlap reflects the degree of shared visual content between the candidate reference frame and the query chunk. If the overlap is too large, the candidate frame mostly repeats the already visible region and contributes limited new information for out-painting. If the overlap is too small, the candidate may become difficult to align with the query chunk, making its structural guidance unreliable or even misleading. Therefore, an effective reference frame should maintain a moderate overlap, balancing reliable spatial correspondence with sufficient complementary context.

Information usefulness measures whether the complementary content provided by the reference frame is actually beneficial for the missing region. Even when a candidate frame has an appropriate spatial overlap, it may still be less useful if its newly introduced content is irrelevant to the area to be out-painted. A good reference should not only introduce additional content, but also provide the right contextual cues needed for out-painting. Thus, the most valuable reference frames are those that are both spatially compatible with the query chunk and informative for the target missing region.

How to Get a Good Reference Frame Based on the above observations, we introduce a geometry-guided and semantics-aware reference retrieval method. For each query chunk, we uniformly sample four query frames to capture its missing-region context, and then extract candidate frames from the full video at fixed temporal intervals and rank them through a two-stage procedure.

In the first stage, we focus on the **appropriate spatial overlap**. As depicted in Fig. 4, we first use the image registration model Mast3R [26] to estimate the spatial relationship between each query frame and candidate frame. Mast3R produces registration scores that reflect the degree of geometric match. We use these scores to remove poorly aligned candidates (*i.e.*, bottom 30% of poorly aligned candidates) before semantic reasoning. This geometric filtering stage substantially reduces the search space and ensures that the remaining candidates satisfy a basic level of spatial compatibility.

In the second stage, the main target is **information usefulness**. Since information usefulness is hard to measure by a simple metric, we resort to a VLM, Qwen3.5-VL-27B [32]. Unlike registration scores, which mainly reflect geometric similarity, the VLM can reason about which visual cues are missing in the out-painting region and whether a candidate frame can provide those cues [1, 44–47].

Specifically, as shown in Fig. 4, the VLM jointly examines the query frames, candidate frames, and registration maps and scores to determine whether a candidate can supply informative appearance, structural, or contextual content for out-painting. This cascaded design, geometry-guided filtering followed by semantics-aware reasoning, improves both retrieval efficiency and selection accuracy. The selected key reference frame is then fed into the model as an additional condition for generation.

Joint Training with Selected Key Reference Frame Rather than using the selected key reference frames only as a test-time plug-in, we incorporate them into the training stage to make the out-painting model better adapted to reference-guided generation. Specifically, we sample a subset of training videos and construct training chunks with their retrieved key reference frames, following the same reference-format design used in inference. The selected frame is appended to the input video as an additional reference condition with an all-zero mask, so that the model learns to exploit its visual cues while keeping the reference content unchanged. This joint training strategy exposes the model to the same type of selected reference conditions during optimization and inference, reducing the mismatch between training and inference.

4 Experiments

4.1 Dataset and Benchmark Construction

Training Dataset Collection (a) *Source video collection*. To construct a high-quality training dataset, we collect a large amount of high-resolution videos covering diverse categories, including portraits, architecture, animals, plants, and natural scenes. After removing videos with black borders, low effective resolution, insufficient bitrate, or visible compression artifacts, we segment videos into scenes using PySceneDetect [31] and retain scenes longer than 64 frames at 24 FPS. We then evaluate each clip by frame quality and scene dynamics, where frame quality is measured by averaged MUSIQ [23] and LAION-Aesthetics [25] scores from frames sampled at 3 FPS, and scene dynamics are measured by RAFT optical-flow magnitude [35]. By jointly ranking these criteria, we select 62K high-quality clips. All selected clips are down-sampled by a factor of two and extracted at 24 FPS as individual frames for subsequent training. (b) *Video captioning*. For each clip, we further obtain a textual caption using VLM captioning [32], which is used as the textual prompt for training. (c) *Random mask for video out-painting*. Based on these high-quality videos, we construct video out-painting training samples by applying random exterior masks. Specifically, we randomly apply the random mask strategy using a mask ratio between 0.25 and 0.66. The unmasked region is always constrained to be a complete rectangle, corresponding to the original visible video content, while the masked region only appears outside this rectangle. This design matches the practical setting of video out-painting and allows the model to generalize to diverse out-painting ratios and spatial layouts.

Existing Benchmarks To verify the effectiveness of LongSCOP, we conduct evaluations on DAVIS [30] and YouTube-VOS [43], which are widely used benchmarks for video out-painting. Following M3DDM [12], we compare the results of different methods in the horizontal direction, using mask ratios of 0.25 and 0.66.

LVO-Bench Benchmark Construction We construct our LVO-Bench benchmark (short for **L**ong **V**ideo **O**ut-painting) from existing LongVideoBench [41] and Video-MME [13] benchmarks using a multi-stage filtering pipeline designed specifically for reference-based long-video out-painting. We first apply scene boundary detection to split each source video into temporally coherent shots, and retain only sufficiently long segments so that temporally distant reference retrieval is meaningful. We then remove degenerate cases with little visual change over time, such as news-style or talking-head footage, by sampling frames, uniformly within each segment and measuring content variation together with motion statistics. Segments with insufficient temporal change are discarded, as are clips with overly violent motion that would make the task ill-posed.

To ensure suitability for left-right out-painting, we further apply geometric validation. We estimate inter-frame motion using optical flow and robust point tracking, and retain clips with stable horizontal camera motion, limited rotation, and small scale drift. We then verify whether temporally distant frames can align with target frames and provide sufficient recoverable content near the left or right borders. Clips with unreliable geometry or weak recoverable regions are discarded. As a result, LVO-Bench focuses on long, dynamic, and geometrically valid videos where reference-frame retrieval is both necessary and useful.

Table 1: Results of video out-painting performance on the DAVIS [30] and YouTube-VOS [43] benchmarks across multiple metrics. The best and second best results are **bold** and underlined. Note that the two benchmarks mainly contain *short clips* where **most samples only have one chunk** (*i.e.*, $\leq$ 3 seconds for DAVIS and 3 to 5 seconds for YouTube-VOS). Though specifically designed for long videos with multiple chunks, we still achieve comparable results on such short clips.

	DAVIS [30]				YouTube-VOS [43]			
	SSIM ↑	PSNR ↑	LPIPS ↓	FVD ↓	SSIM ↑	PSNR ↑	LPIPS ↓	FVD ↓
Dehan [8]	0.627	17.96	0.233	363.1	0.720	18.25	0.228	149.7
SDM [33]	0.708	20.02	0.217	334.6	0.728	19.91	0.200	94.8
M3DDM [12]	0.708	20.26	0.203	300.0	0.731	20.20	0.185	66.6
Infinite-Canvas [7]	0.726	**20.80**	**0.160**	242.8	-	-	-	-
OutDreamer [50]	**0.757**	20.30	0.174	268.9	**0.764**	**20.21**	**0.183**	56.0
VACE [22]	0.696	18.83	0.257	**195.7**	0.739	18.97	0.234	**45.8**
LongSCOP (Ours)	0.754	20.04	**0.160**	197.1	0.748	19.31	0.231	53.9

Benchmark Composition and Scene Diversity The final LVO-Bench contains 52 video cases, with durations ranging from approximately 5 seconds to more than 20 seconds. We examine its diversity along two complementary dimensions: scene environment and camera shot scale. The benchmark contains 26 indoor and 26 outdoor videos. It also includes 3 close-up, 36 medium-shot, and 13 long-shot videos. These counts describe two categorizations of the same 52 cases. The combination of different environments, framing scales, and temporal lengths provides varied conditions for evaluating long-range reference retrieval and video out-painting.

4.2 Metrics and Implementation Details

Metrics Our evaluation approach utilizes four well-established metrics: Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [40], Learned Perceptual Image Patch Similarity (LPIPS) [49], and Frechet Video Distance (FVD) [36]. For assessing PSNR, SSIM, and FVD, the generated videos are converted into frames within a normalized value range of [0, 1]. LPIPS is evaluated over a range of [-1, 1]. About the FVD metric, we adopt a uniform frame sampling, with 16 frames per video for evaluation following M3DDM [12, 15].

Implementation Details Our model is built upon and initialized from VACE-1.3B [22]. The LoRA [20] technique is adopted. The LoRA rank r and LoRA α are set to 16 and 8, respectively. The low-rank adapters are trained using the Adam optimizer [24]. We set the learning rate to 0.0001, and set the weight decay to 0.01. The model is trained on 8 NVIDIA A6000 GPUs for 20 hours.

4.3 Results on Benchmarks

Quantitative Results on DAVIS and YouTube-VOS Benchmarks In Tab. 1, we evaluate the video out-painting performance of our LongSCOP on the DAVIS [30] and YouTube-VOS [43] benchmarks. Note that the two benchmarks mainly contain *short clips* where **most samples only have one chunk** (*i.e.*, $\leq$ 3 seconds for DAVIS and 3 to 5 seconds for YouTube-VOS). Although LongSCOP is specifically designed for long video out-painting with multiple chunks, it still achieves competitive results. Compared to the base VACE, the performance is improved stably in SSIM, PSNR, and LPIPS. For video-level distribution similarity, LongSCOP achieves a comparable FVD to VACE, and clearly better than other task-specific video out-painting methods.

Quantitative Results on our LVO-Bench Benchmark The video out-painting results on our LVO-Bench benchmark are illustrated in Tab. 2. Our LongSCOP improves the FVD by 28.19 points (197.28 *v.s.* 225.47) and PSNR by 0.92 dB (19.87 *v.s.* 18.95) compared to the previous best model. These clear advantages prove the superiority of our LongSCOP.

Results across Scene Categories. We further evaluate LongSCOP across scene categories in LVO-Bench, beyond the aggregate results in Tab. 2. The benchmark includes 26 indoor and 26 outdoor videos, as well as 3 close-up, 36 medium-shot, and 13 long-shot videos. As shown in Tab. 3, LongSCOP improves FVD in every category. The outdoor subset improves from 306.50 to 236.64. Close-up videos show a larger numerical gain, but this subset contains only three cases.

Qualitative Results As shown in Fig. 5, our method is significantly superior to other methods in terms of both stability and semantic consistency. First, our LongSCOP keeps a stable generation over the whole video, while all baseline methods suffer from a degree of generation collapse. Moreover, in

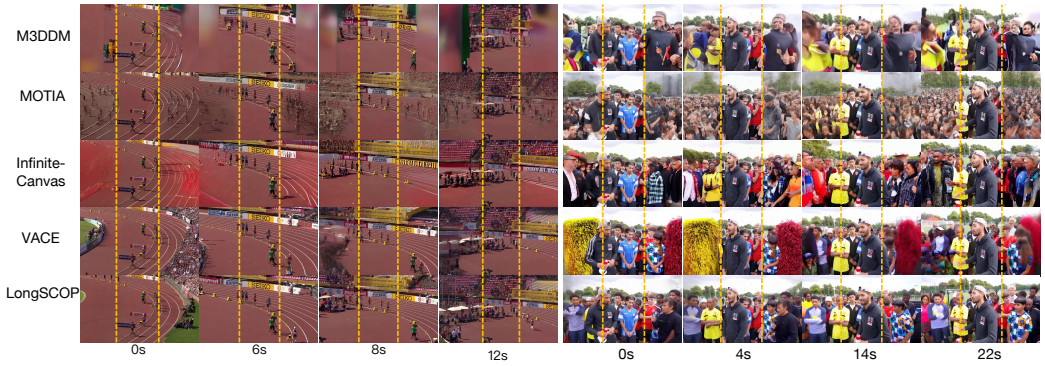


Figure 5: Qualitative comparison. Our LongSCOP could achieve stable video out-painting results and maintain the semantic consistency over a long time horizon. The region circled by the yellow box indicates the original input region. Zoom in for better views.

Table 2: Video out-painting results on LVO-Bench.

Method	SSIM↑	PSNR↑	LPIPS↓	FVD↓
M3DDM	0.505	17.16	0.220	270.3
MOTIA	0.647	16.64	0.320	419.4
Infinite-Canvas	0.731	18.95	0.237	299.3
VACE	0.725	18.59	0.238	225.5
LongSCOP	**0.749**	**19.87**	**0.207**	**197.3**

Table 3: Scene-wise FVD on LVO-Bench.

Category	N	VACE	LongSCOP
Indoor	26	306.17	**299.65**
Outdoor	26	306.50	**236.64**
Close-up	3	421.53	**296.03**
Medium-shot	36	256.12	**217.00**
Long-shot	13	445.53	**401.26**

Table 4: Ablation of stable generation and reference retrieval on LVO-Bench.

Stable Gen.	Refer. Retrieval	SSIM↑	PSNR↑	LPIPS↓	FVD↓
✗	✗	0.725	18.59	0.238	225.5
✓	✗	**0.751**	19.47	0.208	208.6
✓	✓	0.749	**19.87**	**0.207**	**197.3**

Fig. 5(right), LongSCOP maintains almost the same characteristics of the man in yellow and the man in purple throughout all 22 seconds, while all baselines fail to maintain such semantic consistency.

4.4 Ablation Studies

Ablation of Our Proposed Two Main Modules We ablate the two main modules, including stable generation strategies and reference frame retrieval method, in Tab. 4. The stable generation strategies improve SSIM from 0.725 to 0.751, PSNR from 18.59 to 19.47, and FVD from 225.5 to 208.6, confirming their effectiveness in reducing long-horizon degradation. Adding reference frame retrieval further improves PSNR to 19.87 and FVD to 197.3, demonstrating better semantic consistency.

Ablation of Key Reference Retrieval Methods To systematically evaluate the design choices in the retrieval module, we construct a set of implementation variants about candidate filtering strategy, the use of registration-derived auxiliary information, and the organization of VLM reasoning. Specifically, we consider the following variants: (1) using manually selected reference frames as an upper-bound baseline; (2) feeding all candidate frames and the four query frames into the VLM for direct retrieval; (3) filtering out the bottom 30% of candidates according to registration scores before passing the remaining candidates and query frames to the VLM; (4) filtering out the bottom 50% of candidates according to registration scores before passing the remaining candidates and query frames to the VLM; (5) filtering out the bottom 30% of candidates before VLM reasoning, and input registration scores; (6) filtering out the bottom 50% of candidates before VLM reasoning, and input registration scores and rank information; (7) filtering out the bottom 30% of candidates and additionally providing registration scores, rank information, and registration visualizations to the VLM. These variants span a spectrum from purely geometry-based selection to deep integration of geometric filtering and semantic reasoning, allowing us to isolate the contribution of each design choice.

As demonstrated in Tab. 5, Variant (7) achieves the best agreement with the reference-frame rankings derived from objective generation metrics, yielding the strongest performance on HiT and Recall. We therefore adopt Variant (7) as the final design: we first filter out the bottom 30% of candidates using registration scores, and then provide the remaining candidates, query frames, registration ranking information and registration visualizations to the VLM. This result suggests that moderate geometric filtering is effective at removing clearly invalid candidates, while explicit rank-aware registration cues help the VLM focus on references that are both spatially alignable and semantically informative.

Table 5: Quantitative results of various implementation variants of reference frame retrieval methods.

Method	Hit@1↑	Hit@3 ↑	Recall@3 ↑	Recall@5 ↑	GT rank@1 ↓	GT rank@3 ↓	nDCG@3 ↑
Variant (1)	0.1394	0.6653	0.3320	0.5252	6.7849	7.2363	0.6066
Variant (2)	0.1163	0.5814	0.3036	0.4774	7.5349	7.4496	0.5612
Variant (3)	0.1395	0.6162	0.3023	0.4942	7.0076	7.4199	0.5815
Variant (4)	0.1240	0.5890	0.3056	0.4941	6.8992	7.4670	0.5590
Variant (5)	0.1434	0.6550	0.3198	0.5110	6.9729	7.4477	**0.5931**
Variant (6)	0.1589	0.6279	0.3114	0.4994	6.9884	7.4625	0.5768
Variant (7)	0.1473	**0.6744**	**0.3256**	**0.5265**	**6.5969**	**7.0853**	**0.6115**

Table 6: Retrieval generalization across registration models and Qwen3.5 VLMs. 9B and 27B represent the number of parameters used in the Qwen model.

Registration + VLM	Hit@3↑	Rec@5↑	nDCG@3↑
Mast3R + 9B	0.6202	0.4929	0.5890
RoMa + 9B	0.6473	0.5136	**0.6127**
RoMa + 27B	0.6512	0.4845	0.5965
Mast3R + 27B	**0.6744**	**0.5265**	0.6115

Table 7: Effect of registration-based candidate filtering ratios.

Removed	Hit@3↑	Rec@5↑	nDCG@3↑
20%	**0.6744**	0.5142	0.5980
30%	**0.6744**	**0.5265**	**0.6115**
40%	0.6550	0.5032	0.5814
50%	0.6705	0.5174	0.5877
60%	0.6317	0.5065	0.5670
70%	0.5968	0.5065	0.5318
80%	0.5310	0.5045	0.4805

Inference time. For approximately 15-second videos, VACE requires 13.5 minutes on an NVIDIA A6000. Our full pipeline with Mast3R and Qwen3.5-27B takes 20.1 minutes, incurring about 49% overhead. This additional cost comes from reference retrieval and bidirectional generation. Replacing the 27B VLM with 9B reduces end-to-end latency to 16.5 minutes (22% overhead), saving 3.6 minutes per video. These totals are end-to-end measurements; individual stages were not separately timed.

Table 8: Average inference time per 15-second video (Mast3R retrieval).

Method	Time (min)↓	Overhead
VACE	13.5	–
LongSCOP (9B)	16.5	+22%
LongSCOP (27B)	20.1	+49%

Generality across VLMs and registration models. Our reference retrieval framework separates geometric filtering from semantic reasoning. Neither component requires out-painting-specific fine-tuning, allowing pretrained backbones to be replaced directly. We evaluate Mast3R and RoMa [10] for registration, together with Qwen3.5-9B and Qwen3.5-27B for semantic reasoning. As shown in Tab. 6, all four combinations achieve Hit@3 above 0.62 and nDCG@3 above 0.58. Mast3R + 27B gives the highest Hit@3 (0.6744), while RoMa + 9B remains competitive (0.6473). These results indicate flexibility across the tested pretrained backbones. Two real-world videos outside the benchmark further support qualitative robustness.

Sensitivity to geometric filtering. We vary the fraction of registration-filtered candidates from 20% to 80%, keeping other components fixed. As shown in Tab. 7, removing 30% achieves the best Recall@5 (0.5265) and nDCG@3 (0.6115), while tying the best Hit@3 (0.6744). Aggressive filtering removes useful candidates and gradually reduces retrieval accuracy.

Memory cost. Retrieval and generation run in separate stages without modifying the VACE DiT architecture. The video-generation stage retains approximately the same peak memory requirement as VACE. Image registration uses about 8 GB, while Qwen3.5-27B requires approximately 40 GB (distributed across two servers), compared with about 13 GB for Qwen3.5-9B during retrieval. Thus, the smaller VLM reduces retrieval memory cost without changing the generator.

5 Conclusions and Limitations

We introduce LongSCOP, a long video out-painting framework, built upon stable generation strategies, key reference frame retrieval method, large-scale training datasets, and a long video out-painting benchmark, showing clear improvement on both generation stability and semantic coherence.

Limitations Despite its effectiveness, LongSCOP still has some limitations. The VLM-based retrieval module introduces additional inference cost, and its benefit depends on the availability of informative reference frames in the input video. Performance may degrade when missing regions never appear in other frames. How to reduce the retrieval time and how to enhance the robustness of the reference frame remain important directions for future work. In addition, it is important to improve the speed of self-forcing training, which is also left as our future works.

Acknowledgment. This work was partially supported by the National Natural Science Foundation of China (Grant No.62276251) and the Joint Lab of CAS-HK. This research was partially funded by the Ministry of Education and Science of Bulgaria (support for INSAIT, part of the Bulgarian National Roadmap for Research Infrastructure).

References

[1] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.

[2] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[3] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023.

[4] B. Chen, D. Martí Monsó, Y. Du, M. Simchowitz, R. Tedrake, and V. Sitzmann. Diffusion Forcing: Next-token prediction meets full-sequence diffusion. In *NeurIPS*, 2024.

[5] H. Chen, W. Li, J. Gu, J. Ren, S. Chen, T. Ye, R. Pei, K. Zhou, F. Song, and L. Zhu. Restoreagent: Autonomous image restoration agent via multimodal large language models. In *Advances in Neural Information Processing Systems*, 2024.

[6] H. Chen, Y. Zhang, X. Cun, M. Xia, X. Wang, C. Weng, and Y. Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *CVPR*, 2024.

[7] Q. Chen, Y. Ma, H. Wang, J. Yuan, W. Zhao, Q. Tian, H. Wang, S. Min, Q. Chen, and W. Liu. Infinite-Canvas: Higher-resolution video outpainting with extensive content generation. In *AAAI*, 2025.

[8] L. Dehan, W. V. Ranst, P. Vandewalle, and T. Goedemé. Complete and temporally consistent video outpainting. In *CVPRW*, 2022.

[9] D. Ding, Y. Pan, R. Feng, Q. Dai, K. Qiu, J. Bao, C. Luo, and Z. Chen. HomoGen: Enhanced video inpainting via homography propagation and diffusion. In *CVPR*, 2025.

[10] J. Edstedt, Q. Sun, G. Bökman, M. Wadenbäck, and M. Felsberg. RoMa: Robust Dense Feature Matching. In *CVPR*, 2024.

[11] M. Elmoghany, R. Rossi, S. Yoon, S. Mukherjee, E. M. Bakr, P. Mathur, G. Wu, V. D. Lai, N. Lipka, R. Zhang, et al. A survey on long-video storytelling generation: architectures, consistency, and cinematic quality. In *ICCV*, 2025.

[12] F. Fan, C. Guo, L. Gong, B. Wang, T. Ge, Y. Jiang, C. Luo, and J. Zhan. Hierarchical masked 3d diffusion model for video outpainting. In *ACM MM*, 2023.

[13] C. Fu, P. Chen, Y. Shen, Y. Qin, M. Zhang, X. Lin, J. Yang, X. Zheng, K. Li, X. Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. In *NeurIPS*, 2025.

[14] J. Gu. Position: Agentic systems constitute a key component of next-generation intelligent image processing. *arXiv preprint arXiv:2505.16007*, 2025.

[15] J. Gu, H. Cai, H. Chen, X. Ye, J. S. Ren, and C. Dong. PIPAL: A large-scale image quality assessment dataset for perceptual image restoration. In *ECCV*, 2020.

[16] Y. Guo, C. Yang, A. Rao, Z. Liang, Y. Wang, Y. Qiao, M. Agrawala, D. Lin, and B. Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

[17] Y. Guo, C. Yang, A. Rao, C. Meng, O. Bar-Tal, S. Ding, M. Agrawala, D. Lin, and B. Dai. Keyframe-guided creative video inpainting. In *CVPR*, 2025.

[18] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[19] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet. Video diffusion models. In *NeurIPS*, 2022.

[20] 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. In *ICLR*, 2022.

[21] X. Huang, Z. Li, G. He, M. Zhou, and E. Shechtman. Self Forcing: Bridging the train-test gap in autoregressive video diffusion. In *NeurIPS*, 2025.

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

[23] J. Ke, Q. Wang, Y. Wang, P. Milanfar, and F. Yang. MUSIQ: Multi-scale image quality transformer. In *ICCV*, 2021.

[24] D. P. Kingma. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[25] LAION-Aesthetics Team. LAION-Aesthetics, 2022. URL https://laion.ai/blog/laion-aesthetics.

[26] V. Leroy, Y. Cabon, and J. Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024.

[27] B. Lin, Y. Ge, X. Cheng, Z. Li, B. Zhu, S. Wang, X. He, Y. Ye, S. Yuan, L. Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024.

[28] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. In *NeurIPS*, 2024.

[29] K. Liu, W. Hu, J. Xu, Y. Shan, and S. Lu. Rolling forcing: Autoregressive long video diffusion in real time. *arXiv preprint arXiv:2509.25161*, 2025.

[30] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[31] PySceneDetect Team. PySceneDetect Library, 2014. URL https://www.scenedetect.com.

[32] Qwen Team. Qwen3.5: Towards native multimodal agents, 2026. URL https://qwen.ai/blog?id=qwen3.5.

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

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

[35] Z. Teed and J. Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020.

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

[37] 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.

[38] 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 *ECCV*, 2024.

[39] W. Wang, Q. Lv, W. Yu, W. Hong, J. Qi, Y. Wang, J. Ji, Z. Yang, L. Zhao, S. XiXuan, et al. CogVLM: Visual expert for pretrained language models. In *NeurIPS*, 2024.

[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 2004.

[41] H. Wu, D. Li, B. Chen, and J. Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. *arXiv preprint arXiv:2407.15754*, 2024.

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

[43] 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.

[44] Q. Ye, H. Xu, G. Xu, J. Ye, M. Yan, Y. Zhou, J. Wang, A. Hu, P. Shi, Y. Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023.

[45] Z. You, Z. Li, J. Gu, Z. Yin, T. Xue, and C. Dong. Depicting beyond scores: Advancing image quality assessment through multi-modal language models. In *ECCV*, 2024.

[46] Z. You, J. Gu, X. Cai, Z. Li, K. Zhu, C. Dong, and T. Xue. Enhancing descriptive image quality assessment with a large-scale multi-modal dataset. *IEEE TIP*, 2025.

[47] Z. You, K. Wang, H. Zhang, X. Cai, J. Gu, T. Xue, C. Dong, and Z. Zhang. PhotoFramer: Multi-modal image composition instruction. In *CVPR*, 2026.

[48] Z. Yu, M. Megaro-Boldini, R. W. Sumner, and A. Djelouah. Unboxed: Geometrically and temporally consistent video outpainting. In *CVPR*, 2025.

[49] 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.

[50] L. Zhong, F. Li, Y. Huang, J. Liu, R. Pei, and F. Song. Outdreamer: Video outpainting with a diffusion transformer. *arXiv preprint arXiv:2506.22298*, 2025.

[51] D. Zhou, W. Wang, H. Yan, W. Lv, Y. Zhu, and J. Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022.

[52] K. Zhu, J. Gu, Z. You, Y. Qiao, and C. Dong. An intelligent agentic system for complex image restoration problems. In *ICLR*, 2025.

[53] B. Zi, S. Zhao, X. Qi, J. Wang, Y. Shi, Q. Chen, B. Liang, R. Xiao, K.-F. Wong, and L. Zhang. Cococo: Improving text-guided video inpainting for better consistency, controllability and compatibility. In *AAAI*, 2025.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We have discussed the limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.

- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

   Justification: We don't offer any theoretical results in this paper.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We have provided the experimental details in the experiment section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed

instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [No]

   Justification: Codes and datasets will be made publicly available upon acceptance.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: We have provide the setting details in the experiment section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: We report the experimental results following the convention in video outpainting, the same as previous works.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The information on the computer resources is shown in the experiment section.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We strictly adhere to the NeurIPS Code of Ethics.

Guidelines:

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

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: Video out-painting has no obvious negative societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: Our method is formed based on academic datasets.

    Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: We cite all original papers and make sure that our usage is legal.

Guidelines:

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

13. **New assets**

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

Answer: [No]

Justification: Our method is formed based on existing academic datasets.

Guidelines:

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

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

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

Answer: [N/A]

Justification: We do not conduct any crowd-sourcing experiments and research.

Guidelines:

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

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

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

Answer: [N/A]

Justification: We do not conduct any experiments and research about human subjects.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: LLM is used only for writing, editing, or formatting purposes.

Guidelines:

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