Memory-V2V: Memory-Augmented Video-to-Video Diffusion for Consistent Multi-Turn Editing

Dohun Lee[1,2,*], Chun-Hao Paul Huang[1], Xuelin Chen[1], Jong Chul Ye[2], Duygu Ceylan[1], and Hyeonho Jeong[1]

[1] Adobe Research, London, UK
{chunhaoh, xuelinc, ceylan, hyeonhoj}@adobe.com
[2] Kim Jaechul Graduate School of AI, KAIST, Daejeon, South Korea
{leedh7, jong.ye}@kaist.ac.kr
https://dohunlee1.github.io/MemoryV2V/

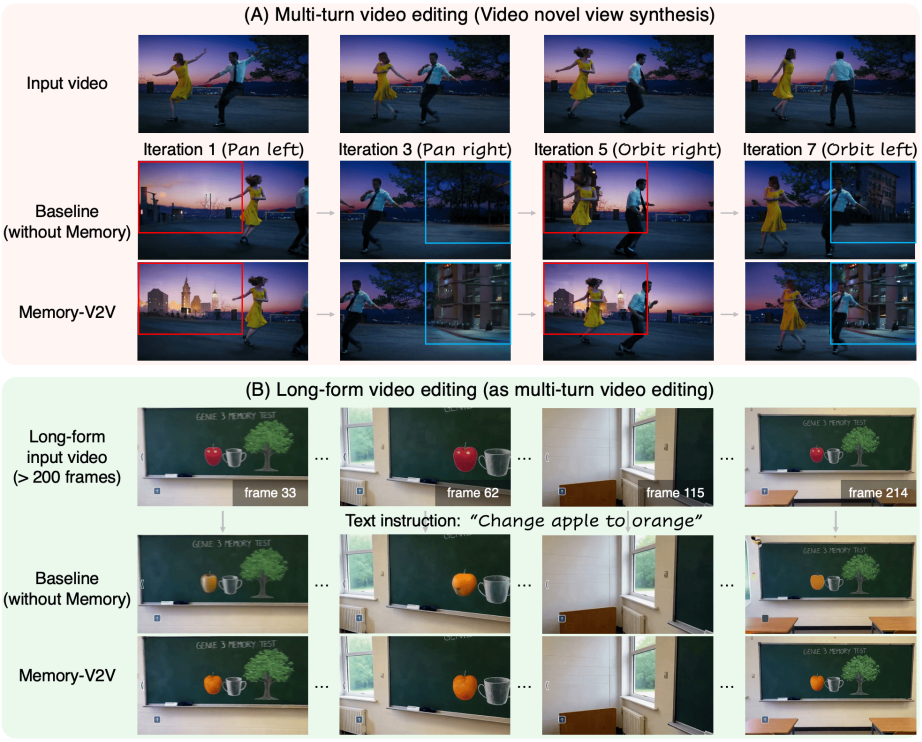


Fig. 1: Memory-V2V enables iterative video editing with long-term memory, producing results that remain consistent with edits from previous turns. (A) and **(B)** show results for video novel view synthesis and text-guided long video editing, respectively. Colored boxes in **(A)** highlight novel-view regions that are expected to remain consistent across editing iterations. Each iteration performs independent denoising. Additional video results are provided on the supplementary project page.

* Work done during an internship at Adobe Research.

Abstract. Video-to-video diffusion models achieve impressive single-turn editing performance, but practical editing workflows are inherently iterative. When edits are applied sequentially, existing models treat each turn independently, often causing previously generated regions to drift or be overwritten. We identify this failure mode as the problem of cross-turn consistency in multi-turn video editing. We introduce Memory-V2V, a memory-augmented framework that treats prior edits as structured constraints for subsequent generations. Memory-V2V maintains an external memory of previous outputs, retrieves task-relevant edits, and integrates them through relevance-aware tokenization and adaptive compression. These technical ingredients enable scalable conditioning without linear growth in computation. We demonstrate Memory-V2V on iterative video novel view synthesis and text-guided long video editing. Memory-V2V substantially enhances cross-turn consistency while maintaining visual quality, outperforming strong baselines with modest overhead.

Keywords: Video editing · Multi-turn editing · Video diffusion models

1 Introduction

Videos are rapidly becoming a dominant medium of modern communication and expression, spanning domains from entertainment to robotics simulation. With the advent of large-scale generative models, users increasingly expect video editing tools that enable attribute-level control — allowing modification of subjects [1, 23, 24], motion [1, 20, 23], or viewpoint [2, 37, 50]. However, in practice, video editing workflows are inherently iterative: users progressively refine outputs over multiple interactions, a setting we term *multi-turn video editing*.

While recent video editing frameworks demonstrate impressive results under single-pass interaction, they often fail to maintain cross-turn consistency across sequential edits. For example, as shown in Fig.1(A), a state-of-the-art video novel view synthesis model, ReCamMaster [2], successfully re-renders the input video from multiple target viewpoints, yet the novel-view regions across iterations remain inconsistent. Similarly, in Fig.1(B), when segments of a long video are iteratively edited using LucyEdit [1], each segment adheres to the local editing prompt, but the global appearance gradually drifts. For instance, an apple edited into an orange appears as visually different oranges across segments.

In this work, we formulate consistent multi-turn video editing as a distinct problem setting and introduce Memory-V2V, a memory-augmented framework tailored to this scenario. Our central design principle is that multi-turn editing requires the memory to function as a constraint mechanism. This is fundamentally different from existing memory-augmented long video generators that assume temporal continuity. Hence, instead of treating previous outputs as additional temporal context, we regard them as a structured external memory that selectively enforces consistency across editing iterations.

A naive solution for enforcing multi-turn consistency is to condition on all previous edits. However, this leads to prohibitive computational growth and introduces redundant or irrelevant context. Instead, we introduce Memory-V2V

guided by two design principles: selective retrieval and relevance-aware capacity allocation. First, rather than conditioning on all previous edits or temporal adjacency retrieval used in autoregressive generation, we retrieve only the most relevant prior videos for the current turn. Crucially, retrieval is task-aware and non-temporal. For video novel view synthesis, we introduce a geometric Field-of-View (FOV) retrieval mechanism that ranks prior edits based on camera overlap. For text-guided long video editing, we retrieve segments according to semantic similarity of their corresponding source videos. Second, we allocate computational capacity proportionally to memory relevance. Retrieved videos are dynamically tokenized with varying spatial resolutions, preserving fine-grained detail for highly relevant edits while aggressively compressing less relevant ones. Furthermore, we introduce adaptive token merging based on attention responsiveness, reducing redundant computation while maintaining essential visual cues. Together, these components enable scalable conditioning on heterogeneous memory sources without linear growth in cost.

We evaluate Memory-V2V on two representative video-to-video editing tasks: video novel view synthesis and text-guided long video editing. Memory-V2V substantially improves cross-iteration consistency in both tasks while also outperforming state-of-the-art baselines in single-turn editing performance. Our contributions can be summarized as follows:

- We formulate *multi-turn video editing* as a distinct problem setting, highlighting the challenge of preserving consistency across independently denoised edits.
- We propose Memory-V2V, a memory augmented framework for iterative editing, where prior edits are treated as structured constraints rather than temporal continuations.
- We introduce task-aware retrieval mechanisms, including FOV-based geometric retrieval and semantic segment retrieval, that select relevant prior edits without incurring linear context growth. We develop dynamic tokenization and adaptive token merging enabling scalable multi-turn conditioning with reduced computational overhead.
- We demonstrate substantial improvements in cross-turn consistency on iterative novel view synthesis and long video editing. To enable training for long video editing, we introduce a dataset augmentation pipeline that temporally extends only target videos from short source–target pairs, eliminating the need for long paired video-to-video training data.

2 Related Work

Video novel view synthesis. Given a monocular video of a dynamic scene, video novel view synthesis aims to generate plausible videos captured from unseen camera trajectories, while preserving the underlying 3D structure and temporal dynamics [40]. Due to the scarcity of large-scale paired multi-view video datasets, prior works have focused on test-time optimization or scene-specific overfitting of existing monocular video generators [19, 29, 53]. Recently,

approaches such as ReCamMaster [2] leverage large-scale synthetic 4D datasets rendered from simulation engines [10,14] to finetune text-to-video diffusion models into multi-view video-to-video models. As shown in Fig. 1(A), when such models are applied iteratively, the generated outputs fail to maintain consistency for regions that were not visible in the input video. In contrast, as demonstrated in Fig. 1(A), we incorporate an explicit visual memory into pre-trained models such as ReCamMaster [2] to ensure strong cross-iteration consistency.

Another line of work [19, 37, 41, 50] employs point-cloud renderings, which, though incomplete, approximate the desired camera trajectories as geometric proxies. Video diffusion models then use these renderings as spatial guidance to synthesize complete and geometrically consistent videos. While some works iteratively refine point clouds to improve static scene novel view synthesis [37,52], they fail to generalize to dynamic scenes with non-rigid motion.

Text-guided video editing. Early text-driven video editing approaches can be broadly divided into two categories [1]. The first relies on inference-time solutions, often using deterministic inversion with test-time optimization to align edits with text prompts [4, 13, 20, 43]. The second conditions video diffusion models on explicit visual cues such as depth maps or optical flow fields [7,8,15,21]. While effective in constrained settings, they often introduce temporal artifacts and struggle to generalize to flexible, open-domain editing scenarios.

Recently, instruction-based foundational video-to-video editing models [1,23, 24, 34, 35] have emerged, jointly conditioning on source videos and text instructions. These models deliver precise, high-fidelity edits while preserving subject identity and motion. However, these models are constrained by a limited temporal context window. A naive workaround is to split a long video into shorter segments and edit each independently, but as shown in Fig. 1(b), this approach leads to significant appearance inconsistencies across segments. In contrast, as illustrated in Fig. 1(b), our memory mechanism enables coherent long-form video editing with consistent appearance and motion across segments. By casting long video editing as a multi-turn editing problem and equipping video-to-video diffusion models [1] with explicit visual memory, we achieve precise and temporally consistent results, *even when segments are denoised independently.*

Long video generation with memory or context. Although modern video models generate visually impressive results, they inherently lack 3D and long-term consistency due to their 2D frame-based representations and limited temporal context [39]. To address this, recent works first reformulate full-sequence video diffusion models into auto-regressive generators [5, 16, 38, 48], then incorporate memory modules that capture information from past generations. One line of work maintains an external cache of previous generations and conditions the current generation by retrieving a subset of them, either as raw RGB frames [28, 45, 49] or as hidden states [3, 51]. Alternatively, other methods aim to compress the long-term context into compact forms such as latent states [9, 33, 58], semantic video tokens [22, 32, 55], or vision–language model

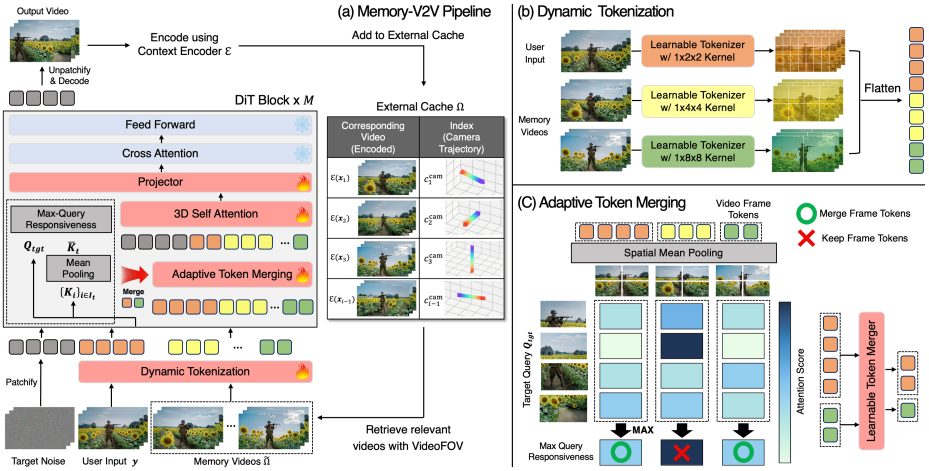


Fig. 2: Memory-V2V framework. (a) From an external cache of previously edited videos, only the top-k most relevant videos are retrieved and used as memory inputs to ensure cross-iteration consistency. (b) Dynamic tokenizers allocate more tokens to highly relevant videos—preserving fine details while maintaining an efficient overall token budget. (c) Adaptive token merging further reduces redundant computations by compressing less informative frames based on their attention-based responsiveness to the target query.

features [6]. Unlike prior literature, we address memory across independently generated editing iterations. In this setup, the model must selectively maintain consistency across multiple edited videos rather than a continuous frame stream.

3 Memory-V2V

3.1 Overview

Existing video-to-video diffusion models are *single-turn video editing* models, which are trained to generate an output video conditioned on a single source video and auxiliary editing signals, approximating the conditional distribution $p(\boldsymbol{x} \mid \boldsymbol{y}, \boldsymbol{c})$, where $\boldsymbol{x}$ is the output video, $\boldsymbol{y}$ is the user-provided source video, and $\boldsymbol{c}$ denotes additional conditions like text or camera pose. In contrast, our *multi-turn video editing* task requires incorporating prior editing history by allowing $\boldsymbol{y}$ to include both the current input video and previously edited videos, enabling the model to maintain cross-edit consistency across iterations.

Memory-V2V achieves this through a hybrid retrieval and compression strategy, built upon an effective memory representation. In the following sections, we first detail our framework in the context of video novel view synthesis. We consider ReCamMaster [2], a pretrained video DiT model capable of single-turn video novel view synthesis, as our base model. We then introduce the core components of Memory-V2V, including an efficient video retrieval mechanism that

selectively fetches relevant past videos from an external cache, dynamic tokenization strategies to represent the retrieved videos, and a compressor that effectively processes the retrieved information (Sec. 3.3). Retrieval is non-parametric, while the tokenization and merging modules are learned end-to-end. Finally, we extend Memory-V2V to text-guided, long-form video editing by formulating it as a multi-turn video editing task (Sec. 3.4).

3.2 Video Retrieval-based Dynamic Tokenization

After each editing iteration j, we store the latent video $\mathcal{E}(\boldsymbol{x}_j) \in \mathbb{R}^{F\times H\times W\times C}$ in an external cache Ω, indexed by its corresponding camera trajectory $\boldsymbol{c}_j^{\text{cam}}$. The latent video's temporal and spatial dimensions are denoted by F, H, W, and C is the latent channel size, and the context encoder $\mathcal{E}$ corresponds to the VAE encoder. At the i-th editing iteration ($i>j>0$), our goal is to produce a new video $\boldsymbol{x}_i$ that remains consistent with both the user-input video $\boldsymbol{y}$ and the previously generated videos in $\Omega = \{(\mathcal{E}(\boldsymbol{x}_j), \boldsymbol{c}_j^{\text{cam}}) \,|\, 0 < j < i\}$, while accurately following the new target camera trajectory $\boldsymbol{c}_i^{\text{cam}}$. Since the total number of cached frames in Ω increases rapidly over multiple editing turns, it is infeasible to condition the pretrained video-to-video DiT model on the entire editing history. We argue that only a subset of the previously edited videos is necessary to maintain spatial and temporal consistency. To this end, we sort all cached latent videos in Ω and dynamically retrieve a set of the top-k most relevant ones, denoted as $\tilde{\Omega}$.

Video FOV retrieval. In case of video novel-view synthesis, we determine the relevance via our proposed VideoFOV retrieval algorithm, which quantifies the geometric overlap between the field-of-view (FOV) of the target camera trajectory $\boldsymbol{c}_i^{\text{cam}}$ and those of cached videos. Specifically, given a target camera trajectory $\boldsymbol{c}_i^{\text{cam}} = \{\boldsymbol{c}_{i,t}^{\text{cam}} \,|\, 1 \le t \le F\}$, where F is the number of frames, we place a unit sphere at the first camera position $\boldsymbol{c}_{i,1}^{\text{cam}}$ and uniformly sample M viewing directions on its surface [3]. For each frame along the trajectory, a sampled point is marked as visible if it lies within the projected image bounds and has positive depth after being transformed to the camera's local coordinate system. The per-frame FOV $\mathcal{F}_{\text{frame}}(\boldsymbol{c}_{i,t}^{\text{cam}})$ is defined as the set of visible sampled points, and the video-level FOV $\mathcal{F}_{\text{video}}(\boldsymbol{c}_i^{\text{cam}})$ is obtained as the union of all frame-level FOVs:

$$\mathcal{F}_{\text{video}}(\boldsymbol{c}_i^{\text{cam}}) = \bigcup_{t=1}^{F} \mathcal{F}_{\text{frame}}(\boldsymbol{c}_{i,t}^{\text{cam}})\,. \tag{1}$$

Given a target trajectory $\boldsymbol{c}_i^{\text{cam}}$ and a cached trajectory $\boldsymbol{c}_j^{\text{cam}}$, we define two complementary FOV similarity metrics:

$$\begin{aligned} s_{\text{overlap}}(\boldsymbol{c}_i^{\text{cam}}, \boldsymbol{c}_j^{\text{cam}}) &= \frac{|\mathcal{F}_{\text{video}}(\boldsymbol{c}_i^{\text{cam}}) \cap \mathcal{F}_{\text{video}}(\boldsymbol{c}_j^{\text{cam}})|}{|\mathcal{F}_{\text{video}}(\boldsymbol{c}_i^{\text{cam}}) \cup \mathcal{F}_{\text{video}}(\boldsymbol{c}_j^{\text{cam}})|}, \\ s_{\text{contain}}(\boldsymbol{c}_i^{\text{cam}}, \boldsymbol{c}_j^{\text{cam}}) &= \frac{|\mathcal{F}_{\text{video}}(\boldsymbol{c}_i^{\text{cam}}) \cap \mathcal{F}_{\text{video}}(\boldsymbol{c}_j^{\text{cam}})|}{|\mathcal{F}_{\text{video}}(\boldsymbol{c}_i^{\text{cam}})|}. \end{aligned} \tag{2}$$

[3] In all experiments, we set $M = 64,800$.

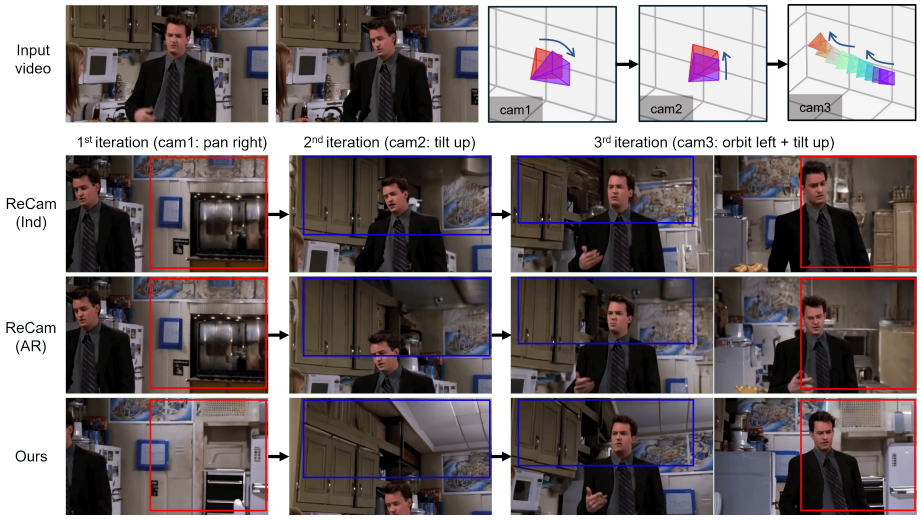


Fig. 3: Qualitative results for multi-turn video novel view synthesis. Compared to baselines, Memory-V2V (Ours) generates videos from new camera trajectories while maintaining consistency across all novel regions generated (e.g., red & blue box).

The final relevance score is a weighted combination:

$$s(\boldsymbol{c}_i^{\text{cam}}, \boldsymbol{c}_j^{\text{cam}}) = \lambda\, s_{\text{overlap}}(\boldsymbol{c}_i^{\text{cam}}, \boldsymbol{c}_j^{\text{cam}}) + (1-\lambda)\, s_{\text{contain}}(\boldsymbol{c}_i^{\text{cam}}, \boldsymbol{c}_j^{\text{cam}}), \tag{3}$$

where we set λ=0.5.

All cached videos in Ω are then ranked by $s(\boldsymbol{c}_i^{\text{cam}}, \boldsymbol{c}_j^{\text{cam}})$, and the top-$k$ highest-scoring videos are selected as the retrieved video set $\tilde{\Omega}$. The full algorithm and implementation details are provided in the supplementary material.

Dynamic tokenization. Unlike long video generation [28,45], where the memory context typically consists of a small number of frames (<10), the multi-turn video editing treats entire videos as retrieval units. As iterations accumulate, the total number of past conditioning frames can easily reach hundreds [4], making direct encoding computationally prohibitive. Hence, we propose to tokenize the conditional videos dynamically based on their relevance to the current edit.

Our goal is to apply multiple learnable tokenizers to each retrieved video. In practice, we train tokenizers with spatio-temporal compression factors ($f' \times h' \times w'$) of 1×2×2, 1×4×4, and 1×8×8, where the 1×2×2 tokenizer processes the user-input video, the 1×4×4 tokenizer processes the top-3 most relevant retrieved videos, and the 1×8×8 tokenizer processes the remaining ones. This adaptive tokenization preserves fine-grained details for the most relevant videos while keeping the total token count manageable.

[4] For video novel view synthesis, a single video contains 81 frames.

3.3 Adaptive Token Merging

While retrieval-based dynamic tokenization effectively allocates the token budget, the quadratic complexity of the DiT's self-attention still leads to high computational cost as the token sequence length increases. Recent studies [44,54,57] reveal that DiT attention maps are inherently sparse, with only a small subset of entries contributing meaningfully to the output. Hence, we propose to adaptively merge unresponsive tokens before the attention operation. This strategy avoids redundant computation while preserving the essential context information.

Frame-level token responsiveness. Previous studies [3,18,47,56] show that the spatially averaged attention features (e.g., query or key features) serve as a strong proxy for measuring how much each frame contributes to the model's prediction. Formally, given query, key, and value matrices $Q, K, V \in \mathbb{R}^{N \times D}$ within a DiT block, we represent all tokens belonging to frame t with a single aggregated vector $\bar{K}_t$ obtained by spatially averaging the key features:

$$\bar{K}_t = \frac{1}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} K_i, \tag{4}$$

where $\mathcal{I}_t$ denotes the set of token indices for frame t. Next, we estimate the responsiveness of each frame by computing its maximum attention response to the target queries Q_{tgt}:

$$R_t = \max_{q \in Q_{\text{tgt}}} (\text{softmax}\left(\frac{q \cdot \bar{K}_t^\top}{\sqrt{D}} \right)). \tag{5}$$

The responsiveness score R_t measures how strongly frame t influences the current generation. A higher R_t indicates that at least one target query token strongly attends to that frame whereas low responsive frames can be safely compressed.

Adaptive token merging. We introduce a learnable convolutional operator $\mathcal{C}_\theta$ that adaptively merges tokens belonging to frames with low responsiveness. Given a set of tokens $\{X_i\}_{i \in \mathcal{I}_t}$ from frame t with low responsiveness score R_t, the operator fuses them into a compact representation:

$$\tilde{X}_t = \mathcal{C}_\theta(\{X_i\}_{i \in \mathcal{I}_t}), \quad \tilde{X}_t \in \mathbb{R}^{\frac{N_t}{r} \times D}, \tag{6}$$

where r is the spatial-temporal reduction factor. We scale r proportionally to the number of conditional videos, as a larger memory context introduces higher redundancy and thus benefits from stronger compression. Empirically, we find that our adaptive merging strategy better preserves generation quality compared to completely discarding low-importance tokens (see Fig. 7).

Block selection for token merging. To determine where to apply token merging within the DiT architecture, we analyze the stability of responsiveness

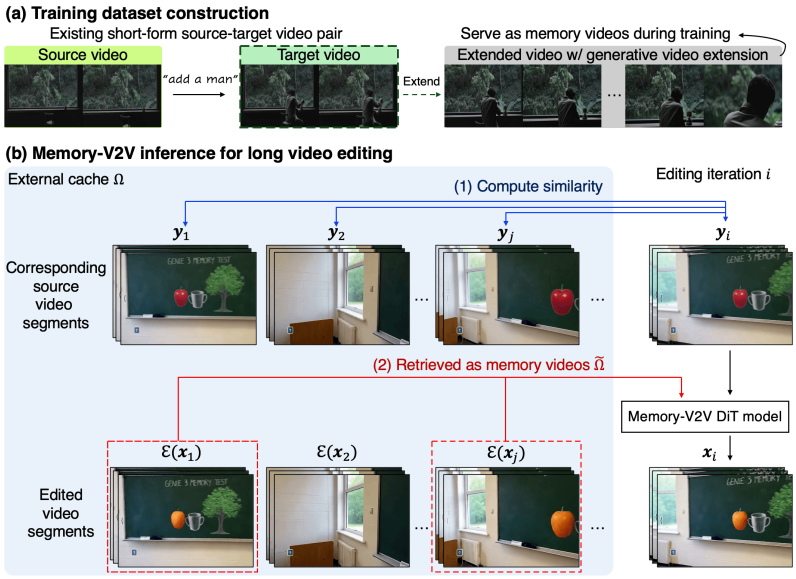


Fig. 4: Long-video editing as multi-turn video editing with Memory-V2V. (a) We extend target videos from an existing video editing dataset for Memory-V2V training. (b) During inference, the external cache Ω stores the editing history as source–target video pairs. At the i-th editing turn, relevant memory videos are retrieved based on the similarity between source video segments.

scores R_t across transformer blocks. In a 30-block DiT, we group layers into early (1–10), middle (11–20), and late (21–30) stages and measure how well the first block in each stage (1, 11, 21) correlates with subsequent blocks. As shown in Tab. 1, the first block (Block 1) shows weak correlation with the rest (Blocks 2–30), meaning early merging risks discarding frames that later become important. Responsiveness stabilizes in mid and late transformer stages, where low-importance frames remain consistently uninformative. We therefore apply token merging at Blocks 10 and 20, where representations are sufficiently mature for reliable compression.

3.4 Extension to Text-guided Long Video Editing

We show that Memory-V2V can be applied to other video editing tasks such as text-guided long video editing. We assume access to a pretrained single-turn text-based video editing model, LucyEdit [1], as our base model in this experiment. Given a long user-input video $\boldsymbol{y}$ typically exceeding 200 frames, which is significantly longer than the temporal context window of the base model ($\sim$ 81 frames), we reformulate the task as a memory-aware iterative editing.

We first divide the long source video $\boldsymbol{y}$ into shorter segments $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_T$ that fit within the base model's temporal window and iteratively edit each segment. When editing the i-th source segment $\boldsymbol{y}_i$, the top-k most relevant edited seg-

Table 1: Pearson/Spearman correlations and Bottom-k overlap (k=50%) evaluating consistency of frame responsiveness across transformer blocks. Correlations measure linear and rank-order alignment. Bottom-k overlap reflects whether low-responsive frames remain consistently uninformative across layers.

	Pearson r ↑	Spearman ρ ↑	Bottom-k overlap (%) ↑
DiT Block 1 vs 2–30	0.608 ± 0.137	0.506 ± 0.267	0.730 ± 0.160
DiT Block 11 vs 12–30	0.723 ± 0.115	0.657 ± 0.139	0.758 ± 0.120
DiT Block 21 vs 22–30	0.753 ± 0.144	0.683 ± 0.126	0.793 ± 0.114

Fig. 5: Text-guided long video editing results by iterative multi-turn editing. In contrast to (iterative) LucyEdit [1] or other frame level propagation baselines, Memory-V2V (Ours) consistently inserts the same 'hat' (left) and changes apple into the same 'orange' (right) across all segments.

ments are retrieved from an external cache $\Omega = \{\mathcal{E}(\boldsymbol{x}_j) \mid 0 < j < i\}$, which stores all past edited segments as video latents. Unlike camera-conditioned video novel view synthesis where each generated video $\boldsymbol{x}_j$ can naturally be indexed by its camera pose $\boldsymbol{c}_j^{\text{cam}}$ for retrieval, text-conditioned editing poses a unique challenge. Text instructions are often ambiguous, making text-based similarity unreliable. Instead, we retrieve memory segments based on visual similarity between source video segments using DINO features [31]. Detailed retrieval algorithms are presented in the supplementary. The retrieved videos are then dynamically tok-

Table 2: Quantitative comparison on multi-turn video novel view synthesis.

	Multi-view Consistency ↓				Camera Accuracy ↓		Visual Quality ↑			
	1st Iter. vs 2nd Iter.	1st Iter. vs 3rd Iter.	2nd Iter. vs 3rd Iter.	Avg. Score	RotErr	TransErr	Subject Consistency	Imaging Quality	Temporal Flickering	Motion Smoothness
TrajCrafter [50]	0.1254	0.2110	0.2090	0.1818	3.66	57.44	0.9452	**0.7385**	0.9661	0.9880
ReCam (Ind) [2]	0.1665	<u>0.1982</u>	0.2031	0.1892	1.97	24.23	<u>0.9483</u>	0.7186	**0.9765**	<u>0.9931</u>
ReCam (AR) [2]	<u>0.1181</u>	0.1985	**0.1290**	<u>0.1485</u>	-	-	-	-	-	-
Memory-V2V (Ours)	**0.1168**	**0.1525**	<u>0.1379</u>	**0.1357**	**1.65**	**13.47**	**0.9494**	<u>0.7242</u>	<u>0.9728</u>	**0.9933**

Table 3: Quantitative comparison results for text-guided long video editing. The higher the better for all metrics.

	Subject Consistency	Background Consistency	Aesthetic Quality	Imaging Quality	Temporal Flickering	Motion Smoothness	DINO-F	CLIP-F
LucyEdit (Ind)	0.8683	0.9026	0.4601	0.6429	0.9844	0.9915	0.6856	0.8225
LucyEdit (FIFO)	0.8737	0.9042	0.4527	0.5598	0.9844	0.9919	0.6784	0.8198
TokenFlow [13]	0.9191	0.9172	0.4206	0.6649	0.9845	0.9896	0.6480	0.7563
RAVE [25]	<u>0.9220</u>	**0.9322**	0.4703	0.6012	0.9813	0.9872	0.7308	0.7713
CCEdit [12]	0.8698	0.9101	0.4113	0.6592	0.9788	0.9762	0.6088	0.7228
FlowEdit [27]	0.9027	0.9123	0.4865	0.6433	0.9852	0.9931	**0.8612**	**0.8943**
DynVFX [46]	0.9119	0.9210	**0.5179**	<u>0.6666</u>	<u>0.9855</u>	<u>0.9937</u>	<u>0.8387</u>	0.8682
Memory-V2V (Ours)	**0.9326**	<u>0.9233</u>	<u>0.4950</u>	**0.6759**	**0.9862**	**0.9939**	0.8019	<u>0.8741</u>

enized where adaptive token merging is also applied when generating the edited segment $\boldsymbol{x}_i$. Once all segments are iteratively edited, they are stitched together to form a single output video $\boldsymbol{x}$ (see Fig. 4(b) for overview).

By casting long video editing as a multi-turn memory-conditioned process, we eliminate the need for long source–target video pairs, which are difficult to obtain in practice. Instead of requiring long paired data, Memory-V2V can be trained using short source–target pairs by extending only the target videos. Concretely, starting from a public short video editing dataset, we generatively extend each target video using an off-the-shelf video extension model [55]. The overall pipeline is visualized in Fig. 4(a).

4 Experiments

4.1 Implementation Details

Video novel view synthesis. Starting from the single-turn video novel view synthesis model ReCamMaster [2], we finetune it for iterative novel view synthesis using a synthetic multi-camera video dataset containing 10 synchronized videos per scene, each captured from distinct camera trajectories. We finetune the self-attention layers, MLP projector, and camera encoder from the base model, together with the newly introduced dynamic tokenizers and adaptive token compressors. During training, 1–6 videos are randomly sampled at each iteration to serve as memory videos $\tilde{\Omega}$. To ensure all tokenizers with varying kernel sizes are trained without reliance on the other, we randomly use one of the tokenizer for the 50% of the training time, while for the other 50% we use a mix of them. Moreover, adaptive token merging is enabled with 50% probability, using a compression factor r randomly sampled from $[0.3, 0.7]$.

Table 4: Quantitative ablation on each component of Memory-V2V.

	Time (s) ↓	MEt3R ↓	Subject Consistency ↑	Aesthetic Quality ↑	Imaging Quality ↑	Motion Smoothness ↑
Dynamic Tokenization Only	980.80	0.2234	**0.9368**	0.5632	**0.7307**	0.9917
+ Video Retrieval	965.95	**0.2169**	0.9338	0.5632	0.7288	0.9918
+ Adaptive Token Merging	661.31	0.2344	0.9361	0.5626	0.7287	**0.9921**
+ All (Full Model)	**648.5**	0.2208	0.9351	**0.5636**	0.7300	0.9919

Text-guided long video editing. We extend the single-turn, instruction-driven video editing model LucyEdit [1] to a long video editor. We finetune it to take both the source video and previously edited (history) videos as input. Training is conducted on 56K samples [24] filtered from the publicly available Señorita-2M dataset [59]. Each sample in Señorita contains a triplet of text instruction, source video, and target video. However, since both the source and target clips in Señorita are short-form videos, we employ a generative video extension model [55] to temporally extend the target videos, using these extended segments as the memory videos during training (see Fig. 4(a)).

Both video novel synthesis and long video editing models are trained with the rectified flow matching loss [11] on 32 A100 GPUs with a total batch size of 32. Due to strong pretrained initialization, stable convergence is achieved within 1–2K finetuning steps.

4.2 Memory-V2V for Video Novel View Synthesis

In this experiment, we evaluate on 40 publicly available videos [2, 30, 42, 50], each serving as a user input. We compare Memory-V2V against ReCamMaster (ReCam) [2] and TrajectoryCrafter (TrajCrafter) [50]. For ReCamMaster, two inference modes are evaluated: (1) ReCam (Ind), which reuses the same user-input video as the source for every iteration, and (2) ReCam (AR), which uses the previous output as the next input in a sequential manner. ReCam (Ind) avoids error accumulation but lacks cross-turn memory, while ReCam (AR) enforces sequential conditioning yet accumulates drift across iterations. Qualitative comparisons are shown in Fig. 3, and quantitative results in Tab. 2. For each input video, we iteratively generate three outputs with highly overlapping camera trajectories (e.g., pan-left and orbit-right) and measure pairwise consistency across the 1st–2nd, 1st–3rd, and 2nd–3rd iterations. We adopt MEt3R to measure cross-view geometric consistency, VBench [17] to evaluate visual quality, and report rotation & translation errors. We observe that as the number of iterations increases, the cross-consistency significantly drops for the base models (TrajCrafter and ReCam (Ind)). While ReCam (AR) preserves consistency between subsequent iterations (e.g., 2nd and 3rd), it fails to keep other iterations (e.g., 1st and 3rd) consistent. In contrast, Memory-V2V results in consistent generations across all iterations while successfully preserving the visual quality and improving camera adherence of the base model.

4.3 Memory-V2V for Long Video Editing

We evaluate the long video editing performance on 50 videos from the Señorita test set [59]. We first compare against two variants of LucyEdit [1]: LucyEdit (Ind), which edits each video segment independently, and LucyEdit (FIFO), which follows FIFO-Diffusion's diagonal denoising strategy to process consecutive frames with increasing noise levels, simulating autoregressive generation [26].In addition, we consider five more baselines: TokenFlow [13], which propagates diffusion features across frames to encourage temporal consistency; RAVE [25], which employs randomized noise shuffling to produce consistent videos while enabling fast video editing; CCEdit [12], which uses keyframe appearance and structure guidance for consistent editing; FlowEdit [27] performs inversion-free, optimization-free prompt based editing via flow matching; and DynVFX [46] enables zero-shot video augmentation by manipulating attention features in a pretrained T2V model. As shown in Fig. 5, while TokenFlow and RAVE can maintain coarse edited-object consistency across frames, they often fail to preserve fine-grained subject identity and unintentionally modify background regions during editing. CCEdit frequently exhibits limited editing controllability, leading to incomplete or inconsistent edits. LucyEdit preserves subject identity and background structure via token reuse through self-attention, but still struggles to maintain edited-object consistency over long sequences. FlowEdit often fails to follow the instruction, leaving the target object unedited or inconsistently edited across frames. DynVFX can edit the object, but may generate the new object at unintended locations rather than replacing the original one as shown in Fig. 5. In contrast, Memory-V2V simultaneously preserves subject identity, background structure, and edited-object consistency by leveraging explicit memory conditioning, enabling stable and coherent edits across extended video sequences (>200 frames). We report visual quality and consistency metrics in Tab. 3, using VBench and cross-frame DINO/CLIP similarity metrics [31,36], where Memory-V2V consistently outperforms all baselines. Cross-frame DINO/CLIP similarity should be considered alongside edit success: overly high values may indicate insufficient editing, whereas overly low values can suggest unintended changes or temporal instability.

4.4 Ablation Study

Ablation on each component. We first ablate the effects of main components of Memory-V2V, as summarized in Tab. 4 and Fig. 6. Dynamic tokenization combined with video retrieval improves cross-iteration consistency, while adaptive token merging notably reduces computational cost. As shown in Fig. 6, retrieval guided by VideoFOV effectively preserves long-term consistency even after multiple iterations (e.g., between the 1st and 5th generations), whereas token merging maintains efficiency without visible degradation.

Ablation on token responsiveness and token reduction. We analyze the effect of conditioning tokens ranked by their responsiveness (Sec. 3.3) under two

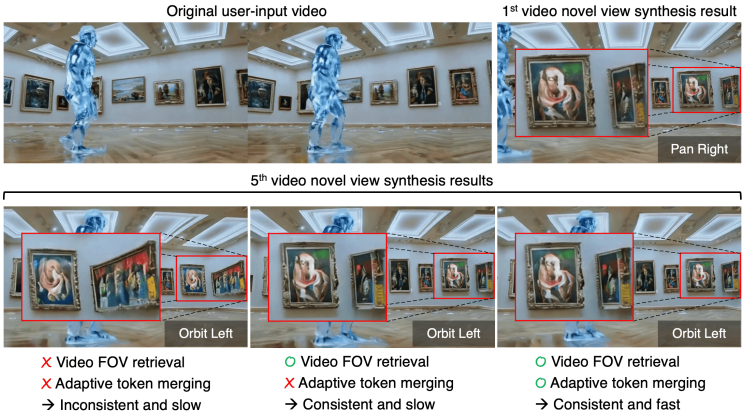


Fig. 6: Qualitative ablation on video retrieval and adaptive token merging.

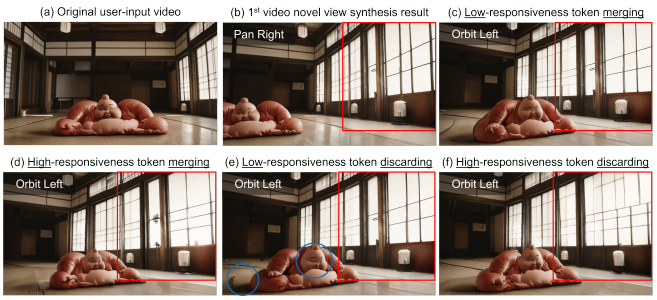


Fig. 7: Qualitative ablation on token responsiveness and reduction strategies.

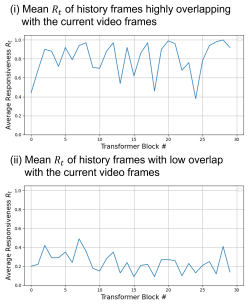


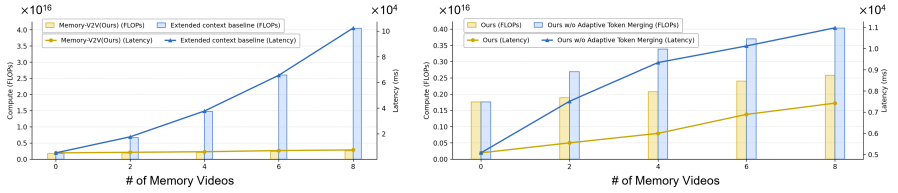
Fig. 8: Mean responsiveness R_t analysis.

reduction strategies: merging and discarding. [5] Multi-turn video editing needs to achieve cross-iteration consistency across edits while preserving the visual quality of the generated videos.

Highly responsive conditioning tokens are essential for cross-consistency across edits. When these tokens are either merged (Fig. 7(d)) or discarded (f), the overlapping novel-view regions become inconsistent. The red-boxed region, which should match (b), fails to remain consistent in both cases. This observation is also supported by the frame-responsiveness analysis in Fig. 8(i), where high-responsive tokens are strongly correlated with the target camera trajectory.

Discarding low-responsive tokens may degrade fine visual details and motion. Fig. 7 shows that discarding these tokens introduces unnatural artifacts in the blue-highlighted regions. Furthermore, Fig. 8(ii) shows that these tokens are less correlated with the video frames generated in the current turn, but they may still contribute to structural consistency. Therefore, instead of discarding them, merging low-responsive tokens reduces computation without affecting

[5] Tokens are partitioned by responsiveness score R_t: the top 50% are high-responsive, and the bottom 50% are low-responsive. Additional experimental details are provided in the supplementary material.



(a) FLOPs and Latency: Ours vs. Naïve scaling context window (b) FLOPs and Latency: Ours vs. Ours w/o Adaptive Token Merging

Fig. 9: Computational cost (FLOPs, Latency) analysis. (a) Comparison between Memory-V2V and naive context-window scaling for memory videos. (b) Comparison of Memory-V2V with and without adaptive token merging. Notably, the computational gains increase as the number of memory videos grows.

cross-consistency and visual fidelity. Adaptive merging preserves cross-turn consistency while reducing computation, whereas discarding low-responsive tokens introduces visual artifacts. Additional examples are provided on the supplementary project page.

Computational cost analysis. We analyze the computational gains of Memory-V2V in Fig. 9. As shown in Fig. 9(a), compared to naive context-window scaling, our method reduces FLOPs and latency by over 90%. Furthermore, Fig. 9(b) shows that adaptive token merging further reduces FLOPs and latency by 30%.

5 Conclusion

We formulated multi-turn video editing as a distinct consistency challenge and introduced Memory-V2V, a memory-augmented framework that treats prior edits as structured constraints. By combining task-aware retrieval and relevance-aware compression, Memory-V2V enables scalable and consistent iterative editing across diverse video tasks.

Limitations. Memory-V2V inherits the limitations (e.g., large view changes) of the underlying single-turn editing model on which it is built. Additionally, since the training data consists only of continuous single-shot videos, Memory-V2V may struggle with editing multi-shot long videos containing abrupt scene transitions. Multi-shot video editing is an exciting research direction we would like to explore in the future.

Acknowledgements

We thank Minguk Kang and Sihyun Yu for their insightful discussions at the early stage of the project. This work was supported by the National Research Foundation of Korea under Grant RS-2024-00336454. This work was supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korean government(MSIT) (No. RS-2024-00457882, AI Research Hub Project). This research was supported by the Advanced GPU Utilization Support Program funded by the Government of the Republic of Korea (Ministry of Science and ICT) (02-26-01-0404).

References

1. AI, D.: Open-weight text-guided video editing (2025), https://platform.decart.ai/
2. Bai, J., Xia, M., Fu, X., Wang, X., Mu, L., Cao, J., Liu, Z., Hu, H., Bai, X., Wan, P., et al.: Recammaster: Camera-controlled generative rendering from a single video. arXiv preprint arXiv:2503.11647 (2025)
3. Cai, S., Yang, C., Zhang, L., Guo, Y., Xiao, J., Yang, Z., Xu, Y., Yang, Z., Yuille, A., Guibas, L., et al.: Mixture of contexts for long video generation. arXiv preprint arXiv:2508.21058 (2025)
4. Ceylan, D., Huang, C.H.P., Mitra, N.J.: Pix2video: Video editing using image diffusion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23206–23217 (2023)
5. Chen, B., Martí Monsó, D., Du, Y., Simchowitz, M., Tedrake, R., Sitzmann, V.: Diffusion forcing: Next-token prediction meets full-sequence diffusion. Advances in Neural Information Processing Systems **37**, 24081–24125 (2024)
6. Chen, N., Huang, M., Meng, Y., Mao, Z.: Longanimation: Long animation generation with dynamic global-local memory. arXiv preprint arXiv:2507.01945 (2025)
7. Chen, W., Ji, Y., Wu, J., Wu, H., Xie, P., Li, J., Xia, X., Xiao, X., Lin, L.: Control-a-video: Controllable text-to-video generation with diffusion models. arXiv e-prints pp. arXiv–2305 (2023)
8. Cong, Y., Xu, M., Simon, C., Chen, S., Ren, J., Xie, Y., Perez-Rua, J.M., Rosenhahn, B., Xiang, T., He, S.: Flatten: optical flow-guided attention for consistent text-to-video editing. arXiv preprint arXiv:2310.05922 (2023)
9. Dalal, K., Koceja, D., Xu, J., Zhao, Y., Han, S., Cheung, K.C., Kautz, J., Choi, Y., Sun, Y., Wang, X.: One-minute video generation with test-time training. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 17702–17711 (2025)
10. Engine, U.: Unreal engine. Retrieved from Unreal Engine: https://www. unrealengine. com/en-US/what-is-unreal-engine-4 (2018)
11. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning (2024)
12. Feng, R., Weng, W., Wang, Y., Yuan, Y., Bao, J., Luo, C., Chen, Z., Guo, B.: Ccedit: Creative and controllable video editing via diffusion models. arXiv preprint arXiv:2309.16496 (2023)
13. Geyer, M., Bar-Tal, O., Bagon, S., Dekel, T.: Tokenflow: Consistent diffusion features for consistent video editing. arXiv preprint arXiv:2307.10373 (2023)
14. Greff, K., Belletti, F., Beyer, L., Doersch, C., Du, Y., Duckworth, D., Fleet, D.J., Gnanapragasam, D., Golemo, F., Herrmann, C., et al.: Kubric: A scalable dataset generator. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3749–3761 (2022)
15. Gu, Y., Zhou, Y., Wu, B., Yu, L., Liu, J.W., Zhao, R., Wu, J.Z., Zhang, D.J., Shou, M.Z., Tang, K.: Videoswap: Customized video subject swapping with interactive semantic point correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7621–7630 (2024)
16. Huang, X., Li, Z., He, G., Zhou, M., Shechtman, E.: Self forcing: Bridging the train-test gap in autoregressive video diffusion. arXiv preprint arXiv:2506.08009 (2025)

17. Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., et al.: Vbench: Comprehensive benchmark suite for video generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21807–21818 (2024)
18. Jeong, H., Chang, J., Park, G.Y., Ye, J.C.: Dreammotion: Space-time self-similar score distillation for zero-shot video editing. In: European Conference on Computer Vision. pp. 358–376. Springer (2024)
19. Jeong, H., Lee, S., Ye, J.C.: Reangle-a-video: 4d video generation as video-to-video translation. arXiv preprint arXiv:2503.09151 (2025)
20. Jeong, H., Park, G.Y., Ye, J.C.: Vmc: Video motion customization using temporal attention adaption for text-to-video diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9212–9221 (2024)
21. Jeong, H., Ye, J.C.: Ground-a-video: Zero-shot grounded video editing using text-to-image diffusion models. arXiv preprint arXiv:2310.01107 (2023)
22. Jiang, J., Li, W., Ren, J., Qiu, Y., Guo, Y., Xu, X., Wu, H., Zuo, W.: Lovic: Efficient long video generation with context compression. arXiv preprint arXiv:2507.12952 (2025)
23. Jiang, Z., Han, Z., Mao, C., Zhang, J., Pan, Y., Liu, Y.: Vace: All-in-one video creation and editing. arXiv preprint arXiv:2503.07598 (2025)
24. Ju, X., Wang, T., Zhou, Y., Zhang, H., Liu, Q., Zhao, N., Zhang, Z., Li, Y., Cai, Y., Liu, S., et al.: Editverse: Unifying image and video editing and generation with in-context learning. arXiv preprint arXiv:2509.20360 (2025)
25. Kara, O., Kurtkaya, B., Yesiltepe, H., Rehg, J.M., Yanardag, P.: Rave: Randomized noise shuffling for fast and consistent video editing with diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
26. Kim, J., Kang, J., Choi, J., Han, B.: Fifo-diffusion: Generating infinite videos from text without training. Advances in Neural Information Processing Systems **37**, 89834–89868 (2024)
27. Kulikov, V., Kleiner, M., Huberman-Spiegelglas, I., Michaeli, T.: Flowedit: Inversion-free text-based editing using pre-trained flow models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19721–19730 (2025)
28. Li, R., Torr, P., Vedaldi, A., Jakab, T.: Vmem: Consistent interactive video scicene generation with surfel-indexed view memory. arXiv preprint arXiv:2506.18903 (2025)
29. Meng, Y., Zhu, Z., Hui, L., Hou, J.: Nvs-solver: Video diffusion model as zero-shot novel view synthesizer. In: The Thirteenth International Conference on Learning Representations (2024)
30. Nan, K., Xie, R., Zhou, P., Fan, T., Yang, Z., Chen, Z., Li, X., Yang, J., Tai, Y.: Openvid-1m: A large-scale high-quality dataset for text-to-video generation. arXiv preprint arXiv:2407.02371 (2024)
31. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
32. Ouyang, W., Xiao, Z., Yang, D., Zhou, Y., Yang, S., Yang, L., Si, J., Pan, X.: Tokensgen: Harnessing condensed tokens for long video generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18197–18206 (2025)

33. Po, R., Nitzan, Y., Zhang, R., Chen, B., Dao, T., Shechtman, E., Wetzstein, G., Huang, X.: Long-context state-space video world models. arXiv preprint arXiv:2505.20171 (2025)
34. Polyak, A., Zohar, A., Brown, A., Tjandra, A., Sinha, A., Lee, A., Vyas, A., Shi, B., Ma, C.Y., Chuang, C.Y., et al.: Movie gen: A cast of media foundation models. arXiv preprint arXiv:2410.13720 (2024)
35. Qin, B., Li, J., Tang, S., Chua, T.S., Zhuang, Y.: Instructvid2vid: Controllable video editing with natural language instructions. In: 2024 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6. IEEE (2024)
36. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
37. Ren, X., Shen, T., Huang, J., Ling, H., Lu, Y., Nimier-David, M., Müller, T., Keller, A., Fidler, S., Gao, J.: Gen3c: 3d-informed world-consistent video generation with precise camera control. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6121–6132 (2025)
38. Ruhe, D., Heek, J., Salimans, T., Hoogeboom, E.: Rolling diffusion models, 2024. URL https://arxiv. org/abs/2402.09470 (2024)
39. Team, H., Wang, Z., Liu, Y., Wu, J., Gu, Z., Wang, H., Zuo, X., Huang, T., Li, W., Zhang, S., et al.: Hunyuanworld 1.0: Generating immersive, explorable, and interactive 3d worlds from words or pixels. arXiv preprint arXiv:2507.21809 (2025)
40. Van Hoorick, B., Wu, R., Ozguroglu, E., Sargent, K., Liu, R., Tokmakov, P., Dave, A., Zheng, C., Vondrick, C.: Generative camera dolly: Extreme monocular dynamic novel view synthesis. In: European Conference on Computer Vision. pp. 313–331. Springer (2024)
41. Wang, Z., Cho, J., Li, J., Lin, H., Yoon, J., Zhang, Y., Bansal, M.: Epic: Efficient video camera control learning with precise anchor-video guidance. arXiv preprint arXiv:2505.21876 (2025)
42. Wiedemer, T., Li, Y., Vicol, P., Gu, S.S., Matarese, N., Swersky, K., Kim, B., Jaini, P., Geirhos, R.: Video models are zero-shot learners and reasoners. arXiv preprint arXiv:2509.20328 (2025)
43. Wu, J.Z., Ge, Y., Wang, X., Lei, S.W., Gu, Y., Shi, Y., Hsu, W., Shan, Y., Qie, X., Shou, M.Z.: Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7623–7633 (2023)
44. Xi, H., Yang, S., Zhao, Y., Xu, C., Li, M., Li, X., Lin, Y., Cai, H., Zhang, J., Li, D., et al.: Sparse videogen: Accelerating video diffusion transformers with spatial-temporal sparsity. arXiv preprint arXiv:2502.01776 (2025)
45. Xiao, Z., Lan, Y., Zhou, Y., Ouyang, W., Yang, S., Zeng, Y., Pan, X.: Worldmem: Long-term consistent world simulation with memory. arXiv preprint arXiv:2504.12369 (2025)
46. Yatim, D., Fridman, R., Bar-Tal, O., Dekel, T.: Dynvfx: Augmenting real videos with dynamic content (2025), https://arxiv.org/abs/2502.03621
47. Yatim, D., Fridman, R., Bar-Tal, O., Kasten, Y., Dekel, T.: Space-time diffusion features for zero-shot text-driven motion transfer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8466–8476 (2024)
48. Yin, T., Zhang, Q., Zhang, R., Freeman, W.T., Durand, F., Shechtman, E., Huang, X.: From slow bidirectional to fast causal video generators. arXiv e-prints pp. arXiv–2412 (2024)

49. Yu, J., Bai, J., Qin, Y., Liu, Q., Wang, X., Wan, P., Zhang, D., Liu, X.: Context as memory: Scene-consistent interactive long video generation with memory retrieval. arXiv preprint arXiv:2506.03141 (2025)
50. YU, M., Hu, W., Xing, J., Shan, Y.: Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. arXiv preprint arXiv:2503.05638 (2025)
51. Yu, S., Hahn, M., Kondratyuk, D., Shin, J., Gupta, A., Lezama, J., Essa, I., Ross, D., Huang, J.: Malt diffusion: Memory-augmented latent transformers for any-length video generation. arXiv preprint arXiv:2502.12632 (2025)
52. Yu, W., Xing, J., Yuan, L., Hu, W., Li, X., Huang, Z., Gao, X., Wong, T.T., Shan, Y., Tian, Y.: Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. arXiv preprint arXiv:2409.02048 (2024)
53. Zhang, D.J., Paiss, R., Zada, S., Karnad, N., Jacobs, D.E., Pritch, Y., Mosseri, I., Shou, M.Z., Wadhwa, N., Ruiz, N.: Recapture: Generative video camera controls for user-provided videos using masked video fine-tuning. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 2050–2062 (2025)
54. Zhang, J., Xiang, C., Huang, H., Wei, J., Xi, H., Zhu, J., Chen, J.: Spargeattn: Accurate sparse attention accelerating any model inference. arXiv preprint arXiv:2502.18137 (2025)
55. Zhang, L., Agrawala, M.: Packing input frame context in next-frame prediction models for video generation. arXiv preprint arXiv:2504.12626 **2**(3), 5 (2025)
56. Zhang, P., Chen, Y., Huang, H., Lin, W., Liu, Z., Stoica, I., Xing, E., Zhang, H.: Vsa: Faster video diffusion with trainable sparse attention. arXiv preprint arXiv:2505.13389 (2025)
57. Zhang, P., Huang, H., Chen, Y., Lin, W., Liu, Z., Stoica, I., Xing, E.P., Zhang, H.: Faster video diffusion with trainable sparse attention. arXiv e-prints pp. arXiv–2505 (2025)
58. Zhang, T., Bi, S., Hong, Y., Zhang, K., Luan, F., Yang, S., Sunkavalli, K., Freeman, W.T., Tan, H.: Test-time training done right. arXiv preprint arXiv:2505.23884 (2025)
59. Zi, B., Ruan, P., Chen, M., Qi, X., Hao, S., Zhao, S., Huang, Y., Liang, B., Xiao, R., Wong, K.F.: Se\~ norita-2m: A high-quality instruction-based dataset for general video editing by video specialists. arXiv preprint arXiv:2502.06734 (2025)