LONGLIVE-RAG: A GENERAL RETRIEVAL-AUGMENTED FRAMEWORK FOR LONG VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive (AR) video diffusion enables variable-length synthesis, but long-horizon generation often suffers from accumulated errors and identity drift. For efficiency, existing methods commonly adopt sliding-window attention during generation. This creates an irreversible generation trajectory: once the active window accumulates appearance errors, subsequent generations can only condition on this degraded trajectory and drift further away. We address this limitation by formulating long video generation as a retrieval-augmented generation (RAG) problem. Rather than relying solely on the recent window, we treat previously generated latents as a dynamic, searchable history. We propose **LongLive-RAG**, a general retrieval framework for AR video generation. At each new block, LongLive-RAG uses a query embedding to retrieve relevant historical latents. This lightweight retrieval step adds only a small overhead relative to generation and lets the generator condition on non-local context instead of only the recent window. To make retrieval more discriminative, we introduce the Window Temporal Delta Loss that suppresses redundant local similarity and encourages embeddings to capture meaningful temporal changes. Together, these components help reduce error accumulation caused by sliding-window attention. Experiments across multiple AR backbones and generation lengths show improved long-video quality and the best average VBench-Long rank. To our knowledge, among open-ended AR long video generation methods, LongLive-RAG is the first to formulate self-generated latent history as content-addressable retrieval memory.

1 INTRODUCTION

Recent video diffusion models can synthesize short clips with strong visual quality (Yan et al., 2021; Yu et al., 2023; 2024; Kondratyuk et al., 2023; Deng et al., 2025; Jin et al., 2025; Team Wan et al., 2025; Polyak et al., 2024; Kong et al., 2024; Yang et al., 2024). Many applications, however, require longer videos in which subjects, backgrounds, and scene layouts remain consistent across tens or hundreds of seconds (Brooks et al., 2024; Feng et al., 2024; Hong et al., 2025; Sun et al., 2025; Yang et al., 2026b; Hu et al., 2023; Ren et al., 2025). AR video diffusion is a practical formulation for this setting because it generates latent blocks causally and can continue beyond a fixed clip length (Chen et al., 2024; Yin et al., 2025; Huang et al., 2025b; Zhu et al., 2026; Liu et al., 2025; Yang et al., 2026a; Kodaira et al., 2025; Henschel et al., 2025). In practice, long AR generation often suffers from error accumulation. To keep computation bounded, AR generators usually keep only the most recent blocks as context and discard earlier blocks. Once these recent blocks contain appearance drift, identity changes, or background artifacts, subsequent blocks may condition on these errors and further propagate them (Liu et al., 2024; Elmoghany et al., 2025; Brooks et al., 2024).

Existing methods for long video generation address this problem by changing how history is kept. Attention-sink methods retain fixed early tokens or frames as anchors (Xiao et al., 2024; Yang et al., 2026a; Li et al., 2026; Lu et al., 2025). Methods based on positional extrapolation extend the usable temporal range of the model (Su et al., 2024; Zhao et al., 2025; Yesiltepe et al., 2026). Methods based on compressed-history tokens summarize older states into substitute tokens or recurrent memory (Yi et al., 2026; Yu et al., 2025b; Kim et al., 2026; Zhang et al., 2025b). These methods can reduce error

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

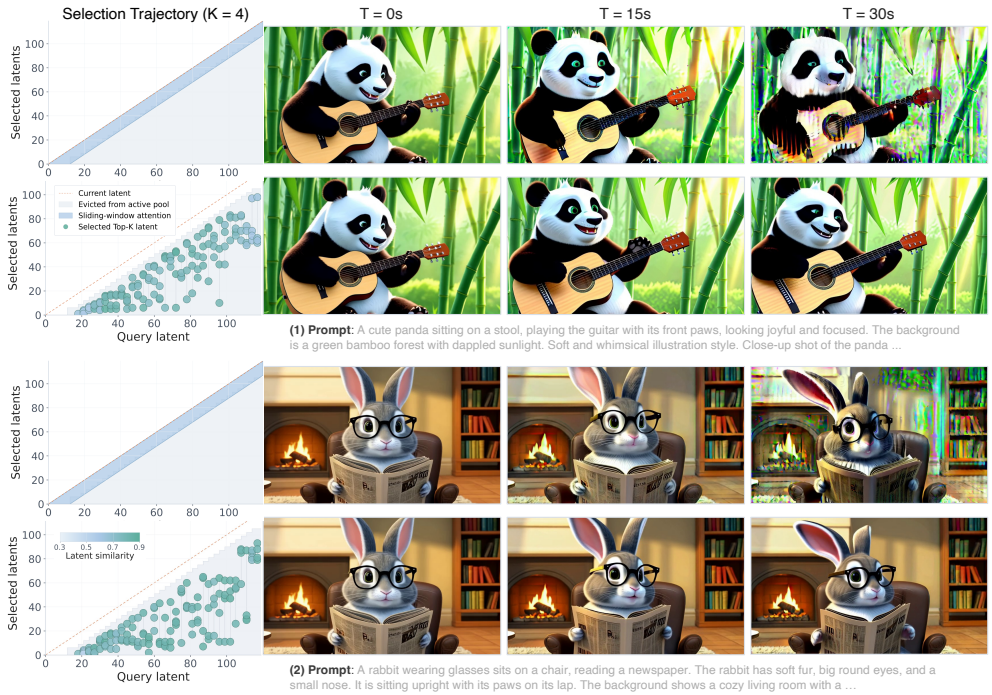


Figure 1: LongLive-RAG searches previously generated latents for context relevant to each new segment, helping preserve subjects and scenes over long rollouts.

and improve stability, but they still have limitations. Fixed anchors may not match the current content, positional extrapolation does not prevent error accumulation once the visible context has drifted, and compressed-history tokens may lose critical native visual details for subsequent generations.

Our observation is that long video generation needs a way to use generated states when they are useful, instead of relying only on the recent window or a fixed summary of the past, as illustrated in Figure 1. The generated history may contain earlier context that better preserves the subject appearance, background layout, or scene structure needed by the current block. Retrieval provides a direct way to use this history. Before generating a new block, the model can search previously generated latents and bring back relevant historical context. When the recent window has drifted, this retrieved context can provide a reference and reduce dependence on the corrupted local context.

Based on this observation, we propose **LongLive-RAG**, a general retrieval-augmented framework for AR video generation. LongLive-RAG stores previously generated latents and builds compact embeddings for search. For each new block, an encoder maps the latest completed latent to a query embedding. The method retrieves the top-K historical latents, combines them with the local latents, and uses them as the generator's attention context. The generator attends to the retrieved historical latents and local latents rather than compressed-history tokens. This gives the generator access to useful history outside the sliding window. It also keeps the base AR generator unchanged and adds only lightweight retrieval overhead relative to transformer attention.

The retrieval embedding must be suitable for search. Adjacent video latents are often very similar, so a reconstruction-only encoder can map many nearby blocks to almost identical embeddings. In that case, top-K retrieval may return blocks that add little beyond the local window. We introduce a Window Temporal Delta Loss to reduce excessive similarity among nearby latents, and add a smoothing term to keep embeddings stable over time. Together, these losses make retrieval more discriminative while preserving visual information.

Our main contribution is to formulate open-ended AR video generation as retrieval over self-generated latents, together with a lightweight retrieval framework and an embedding objective that selects useful non-local context. LongLive-RAG retrieves useful past latents with a lightweight encoder and uses them as extra attention context. We compare it with two alternatives for addressing sliding-window

limits: ∞-RoPE for positional extrapolation and Deep Forcing for compressed-history tokens. Across three AR backbones, Causal-Forcing, Self-Forcing, and LongLive, and three generation lengths, 30s, 60s, and 120s, LongLive-RAG achieves the best average VBench-Long rank and improves subject consistency, background consistency, motion smoothness, and imaging quality. Qualitative results and ablations further support these results.

2 Related Work

This section briefly reviews related work; Appendix C gives the full discussion.

AR long video generation. AR video diffusion emits frames or latents causally for streaming and variable-length synthesis (Chen et al., 2024; Yin et al., 2025; Huang et al., 2025b; Liu et al., 2025; Yang et al., 2026a; Kodaira et al., 2025; Henschel et al., 2025; Sand.ai et al., 2025; Chen et al., 2025; Gu et al., 2025; Zhai et al., 2025). Recent methods improve local denoising, self-generated context, or streaming attention (Yin et al., 2025; Huang et al., 2025b; Cui et al., 2025; Zhu et al., 2026; Chen et al., 2026; Liu et al., 2025; Yang et al., 2026a). They mainly improve the quality or causality of each local rollout step; LongLive-RAG is orthogonal to these advances and keeps the generator unchanged.

Context visibility and memory. Long video generation systems expose history through sliding windows, fixed anchors, positional extrapolation, or compressed-history tokens (Xiao et al., 2024; Yang et al., 2026a; Li et al., 2026; Lu et al., 2025; Su et al., 2024; Zhao et al., 2025; Yesiltepe et al., 2026; Yi et al., 2026; Chen et al., 2026; Yu et al., 2025b; Kim et al., 2026; Zhang et al., 2025b;a). These strategies differ in how much history is kept and how it is stored; LongLive-RAG instead searches the generated history and brings back context for attention.

Retrieval memory in video generation. Retrieval is natural in world models when geometry, camera pose, or scene coordinates are available (Hu et al., 2023; Feng et al., 2024; Li et al., 2025; Yu et al., 2025a; Xiao et al., 2025; Sun et al., 2025; Yang et al., 2026b; Wu et al., 2026). Open-ended text-to-video generation does not provide such explicit retrieval cues by default, so useful history must be found from the generated content itself; LongLive-RAG searches the generated history and uses the matched context during rollout.

Adaptive context retrieval. Concurrent work also studies retrieval and memory for long video generation. Mixture of Contexts (MoC) learns sparse attention routing over video chunks, while Memory Forcing uses geometry-indexed spatial memory for Minecraft scene revisitation (Cai et al., 2026; Huang et al., 2025a). MemFlow retrieves historical frames using the current text prompt (Ji et al., 2025). LongLive-RAG instead indexes self-generated latent blocks with a learned visual embedding and inserts the retrieved latent context into an otherwise unchanged AR generator. These methods share the goal of long-range consistency but differ in retrieval signal, memory representation, and whether retrieval is learned as part of attention or applied to generated latent history.

3 Method

LongLive-RAG augments AR video diffusion with retrieval over self-generated latents while keeping the base generator fixed. Section 3.1 defines the sliding-window and LongLive-RAG context assembly rules. Section 3.2 describes the paired embedding/context history banks and the retrieval rule. Section 3.3 explains retrieval embedding space training. Section 3.4 summarizes the inference.

3.1 AR Context Assembly

Sliding-window context. At block t, an AR video diffusion model denoises the current latent representation while attending to context from previously completed blocks (Yin et al., 2025; Huang et al., 2025b; Liu et al., 2025; Yang et al., 2026a). Standard inference typically keeps a bounded sliding-window context

$$\mathcal{A}_t^{\mathrm{sw}} = [\mathcal{C}_{\mathrm{sink}} \| \mathcal{C}_{\mathrm{loc}}], \tag{1}$$

where $\mathcal{C}_{\mathrm{sink}}$ stores optional fixed early context and $\mathcal{C}_{\mathrm{loc}}$ stores context from recent completed blocks (Xiao et al., 2024; Yang et al., 2026a; Yu et al., 2025b). Once earlier context is dropped, the generator

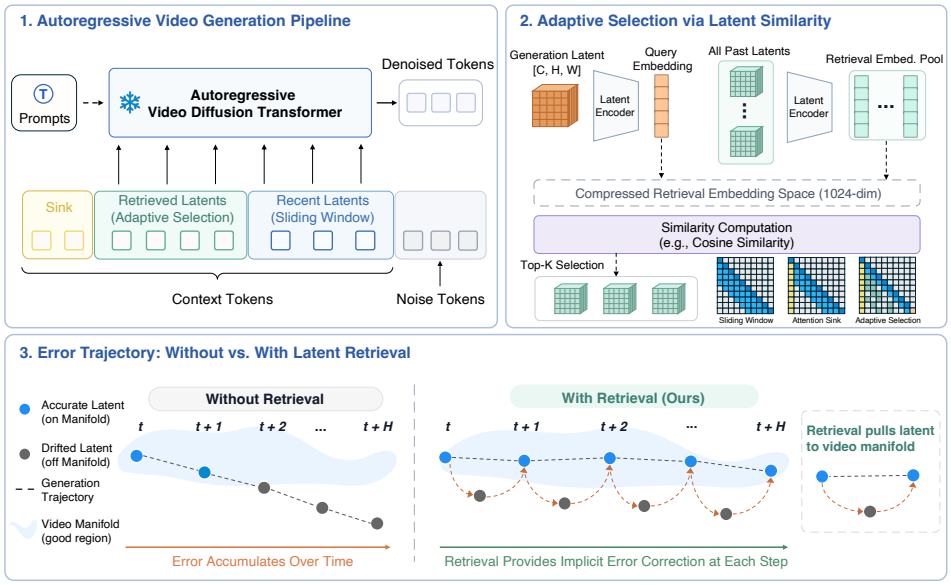


Figure 2: LongLive-RAG encodes completed latents, retrieves relevant historical entries, and combines them with recent context for the next autoregressive block.

cannot use it again; if the recent context has drifted, the next block is still generated from this drifted context.

LongLive-RAG context. LongLive-RAG does not update the base generator or add trainable layers inside the denoising backbone. Instead, it uses an offline-trained retrieval encoder and adds relevant historical context entries directly to the attention context:

$$\mathcal{A}_t^{\text{rag}} = [\mathcal{C}_{\text{sink}} \| \mathcal{M}_t \| \mathcal{C}_{\text{loc}}]. \tag{2}$$

Here $\mathcal{M}_t$ denotes the retrieved non-local context entries for block t. The current block is then denoised with the retrieved history and local context. Importantly, LongLive-RAG does not add attention layers or change the denoising rule.

3.2 Indexing the History

Compact embeddings for search. Let $\hat{x}_t \in \mathbb{R}^{C\times H\times W}$ be the clean latent at block t. This latent remains spatially dense because the generator needs it to preserve visual quality and fine details during denoising, as in WAN (Team Wan et al., 2025). Retrieval has a different requirement: it only needs a discriminative key for finding useful history, not the full representation used for synthesis. LongLive-RAG therefore performs search in a compressed retrieval embedding space instead of directly comparing all historical latents with spatial shape $[C, H, W]$. For a completed block i, an encoder produces a 1024-dimensional embedding v_i, and the base backbone provides the corresponding context entry $\mathcal{C}_i$. The historical search bank is $\mathcal{H}_v = \{(i, v_i)\}$, paired with a context bank $\mathcal{H}_{\text{ctx}} = \{(i, \mathcal{C}_i)\}$. The embeddings are used for retrieval, while the generator attends to the matched context.

We train a latent autoencoder with encoder E_ψ and decoder D_ψ using reconstruction loss,

$$\mathcal{L}_{\text{rec}} = \|D_\psi(E_\psi(\hat{x}_t)) - \hat{x}_t\|_2^2, \tag{3}$$

and use $v_t = E_\psi(\hat{x}_t)$ as the retrieval embedding. Here E_ψ is the encoder and $v_t \in \mathbb{R}^{1024}$ lies in the compact search space. The encoder is trained offline on clean latents with the base generator fixed; the decoder is used only to shape this space and is not inserted into the generator at inference. After block t is completed, LongLive-RAG computes v_t and associates it with the block context $\mathcal{C}_t$. Entries remain in the rolling local cache while they are recent; when they leave the local window, their paired $(v_i, \mathcal{C}_i)$ entries are added to $\mathcal{H}_v$ and $\mathcal{H}_{\text{ctx}}$.

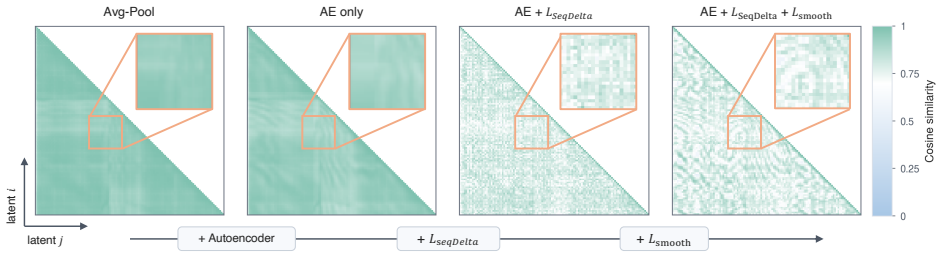


Figure 3: Embedding-space analysis. Darker green indicates a higher cosine similarity between the embeddings. Relying on reconstruction alone preserves content but leaves temporally nearby latents overly similar; the temporal delta separates redundant local states, and smoothing effectively stabilizes the embedding trajectory along the entire generation trajectory.

Similarity retrieval. For block t, LongLive-RAG performs retrieval once before denoising and reuses the selected context for all N denoising steps. The query is the embedding of the most recent completed latent, i.e., $v_q = v_{t-1}$ after the previous block has been finalized. LongLive-RAG ranks stored embeddings with cosine similarity,

$$a_i = \cos(v_q, v_i), \qquad i \in \mathcal{P}_t, \tag{4}$$

where $\mathcal{P}_t$ skips the most recent history entries to avoid retrieving near-duplicate recent context. The top-K indices $I_t = \text{Top-}K(\{a_i\}_{i\in\mathcal{P}_t})$ select matched entries from $\mathcal{H}_{\text{ctx}}$:

$$\mathcal{M}_t = [\mathcal{C}_i]_{i\in I_t}. \tag{5}$$

These retrieved context entries are combined with the local cache in the current attention context.

Context handling. Each entry in $\mathcal{H}_{\text{ctx}}$ follows the same interface used by the local window. LongLive-RAG does not introduce new attention layers; it only changes which historical context entries are exposed to attention at each block. Retrieved entries are therefore handled in the same way as local context entries inside the original generator.

Retrieval overhead. Table 1 reports the additional runtime introduced by LongLive-RAG retrieval over a 120s rollout. LongLive-RAG performs retrieval once per AR block and adds 4.08 ms per block, totaling 490 ms of retrieval overhead: 480 ms from latent encoding and 10 ms from top-K search. The overhead is dominated by encoding; similarity search is negligible at this scale. For long video generation, this cost is small compared with the diffusion rollout itself.

Table 1: 120s retrieval overhead.

Component	ms/block	Total (ms)
Latent encoding	3.96	480
Top-K search	0.08	10
Total	4.08	490

3.3 Learning the Embedding Space

Why reconstruction is not enough. The search space has a different requirement from the VAE latent space used by video generators such as WAN (Team Wan et al., 2025). For synthesis, adjacent latents should change smoothly and preserve dense visual detail. For retrieval, however, the embedding must also decide which history is worth bringing back into attention. If the embedding only optimizes reconstruction, it tends to preserve the local continuity of video too well: nearby latents become almost interchangeable search keys. Figure 3 illustrates this behavior. With AE-only training, the similarity map contains a broad high-similarity band around the diagonal, meaning that many neighboring blocks look equally valid to the retriever. In inference, such an index can waste the top-K budget on locally redundant states, adding little beyond the recent cache.

This motivates an anti-collapse term at the temporal scale where redundancy is most common. We do not want to repel all frames from each other, since the same subject or scene may legitimately reappear later. Instead, we only discourage excessive similarity within a short window, where frames are likely to be visually redundant and already covered by the local cache. The loss is therefore local

and margin-based: pairs below the margin are not pushed apart, and long-range repetitions are not treated as negatives. We first define a pairwise temporal delta penalty,

$$\mathcal{L}_{\Delta}(v_t, v_{\text{ref}}) = \lambda_{\Delta} \max(0,\, \cos(v_t, v_{\text{ref}}) - m), \tag{6}$$

which penalizes local pairs whose cosine similarity exceeds a margin m. The Window Temporal Delta Loss averages this penalty over a local temporal window. For a sequence of length T and window size w, let $\bar{w} = \min(w, T-1)$.

$$\mathcal{L}_{\text{SeqDelta}} = \frac{1}{\bar{w}} \sum_{\tau=1}^{\bar{w}} \frac{1}{T-\tau} \sum_{t=\tau+1}^{T} \mathcal{L}_{\Delta}(v_t, v_{t-\tau}). \tag{7}$$

Stable embedding trajectory. Temporal separation alone is still insufficient. If the embedding moves too sharply from one block to the next, the top-K set can change for reasons unrelated to meaningful content changes, which may make the retrieved context unstable. The right behavior is visible in Figure 3: after local redundancy is reduced, the embedding trajectory should remain organized rather than noisy. We therefore add a second-order smoothness penalty

$$\mathcal{L}_{\text{Smooth}} = \lambda_{\text{smooth}} \frac{1}{T-2} \sum_{t=3}^{T} \|v_t - 2v_{t-1} + v_{t-2}\|_2, \tag{8}$$

Combining these terms gives the final training objective:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{SeqDelta}} + \mathcal{L}_{\text{Smooth}}. \tag{9}$$

The three terms play distinct roles. Reconstruction keeps the embedding tied to visual content; temporal delta makes nearby redundant states less likely to dominate top-K retrieval; and smoothing keeps the embedding trajectory stable across the generation trajectory. This is the design principle behind Figure 3: LongLive-RAG does not seek a generic compressed representation, but a search geometry that matches the needs of non-local context selection.

3.4 Inference

Algorithm 1 LongLive-RAG inference

Require: $G_\theta, E_\psi; T, N, L, K, R$
State: $\mathcal{C}_{\text{sink}}, \mathcal{C}_{\text{loc}}, \mathcal{H}_{\text{ctx}}, \mathcal{H}_v$

$\mathcal{C}_{\text{sink}}, \mathcal{C}_{\text{loc}}, \mathcal{H}_{\text{ctx}}, \mathcal{H}_v \leftarrow \emptyset$
for block $t = 1, \ldots, T$ **do**
  // Step 1: query searches pool
  $I_t \leftarrow \emptyset$
  if $|\mathcal{H}_v| > R$ **then**
    $v_q \leftarrow v_{t-1}$ ▷ *latest embedding*
    $I_t \leftarrow$ top-K matches in $\mathcal{H}_v$ outside recent R
  end if
  // Step 2: denoise latent t
  $\mathcal{M}_t \leftarrow [\mathcal{H}_{\text{ctx}}[i]]_{i \in I_t}$
  $\mathcal{A}_t \leftarrow [\mathcal{C}_{\text{sink}} \| \mathcal{M}_t \| \mathcal{C}_{\text{loc}}]$
  for step $s = 1, \ldots, N$ **do**
    $x_t^{s+1} \leftarrow G_\theta(x_t^s;\, \mathcal{A}_t)$
  end for
  // Step 3: encoder updates pool
  $\hat{x}_t \leftarrow x_t^{N+1}$; $v_t \leftarrow E_\psi(\hat{x}_t)$
  Append context and v_t to $\mathcal{C}_{\text{loc}}$
  if $|\mathcal{C}_{\text{loc}}| > L$ **then**
    Add oldest entry to $(\mathcal{H}_{\text{ctx}}, \mathcal{H}_v)$
    Keep the latest L entries in $\mathcal{C}_{\text{loc}}$
  end if
end for

Algorithm 1 gives the inference procedure with a reduced notation set. Here G_θ is the frozen AR generator and E_ψ is the encoder. T is the number of latents, N is the number of denoising steps per block, L is the local-window size, K is the retrieval budget, and R is the recency guard that prevents the search from selecting near-duplicate recent context. The state consists of an optional sink cache $\mathcal{C}_{\text{sink}}$, a rolling local cache $\mathcal{C}_{\text{loc}}$ for the latest context, the pool $\mathcal{H}_v$, and a paired historical context pool $\mathcal{H}_{\text{ctx}}$. Entry i of $\mathcal{H}_v$ is the compact search key for entry i of $\mathcal{H}_{\text{ctx}}$. At block t, $v_q = v_{t-1}$ is the query from the latest completed latent, I_t is the selected top-K history indices, $\mathcal{M}_t$ is the matched context, and $\mathcal{A}_t = [\mathcal{C}_{\text{sink}} \| \mathcal{M}_t \| \mathcal{C}_{\text{loc}}]$ is the attention context. When no eligible historical entry exists, I_t is empty and the method reduces to the base sink-plus-local context. The latent x_t^s denotes block t at denoising step s, $\hat{x}_t = x_t^{N+1}$ is the clean completed latent, and $v_t = E_\psi(\hat{x}_t)$ is the embedding stored with the completed local entry before later offloading to the historical pool. The notation $|\cdot|$ denotes the number of entries in a bank, and $\|$ denotes concatenation along the context dimension. As illustrated in Figure 2(3), this gives later blocks access to content-relevant non-local context while retaining the original AR generator.

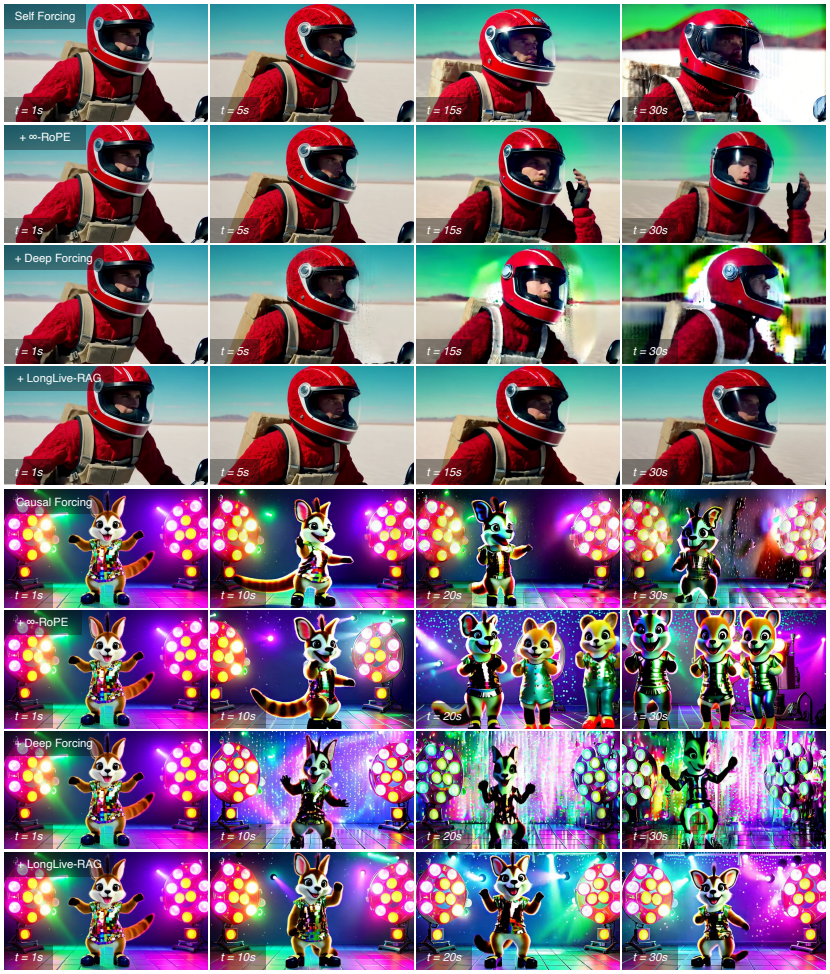


Figure 4: Qualitative comparison on 30s generations. LongLive-RAG better preserves subject and background appearance in these examples.

4 EXPERIMENTS

We evaluate whether LongLive-RAG improves long-video quality across three AR backbones and three generation lengths, 30s, 60s, and 120s, and whether learned retrieval contributes to the gains.

4.1 SETUP

Implementation details. We evaluate LongLive-RAG on three AR backbones: Causal-Forcing (Zhu et al., 2026), Self-Forcing (Huang et al., 2025b), and LongLive (Yang et al., 2026a). For each backbone, we compare the base model, ∞-RoPE (Yesiltepe et al., 2026), Deep Forcing (Yi et al., 2026), and LongLive-RAG. ∞-RoPE represents positional extrapolation, while Deep Forcing represents compressed-history tokens. All inference runs use the same base sampling settings, sink size 1, and total attention window size 12. LongLive-RAG uses $K = 6$, giving the context layout $[1\ \text{sink} \| 6\ \text{retrieved context} \| 5\ \text{local window}]$. Under this fixed context budget, the compared methods differ only in how the available slots are filled: the base model and ∞-RoPE use sink-plus-local context, while Deep Forcing uses compressed-history context following its inference rule. The reported inference and evaluation runs for the quantitative tables take about one week on 6 NVIDIA RTX A6000 GPUs, excluding retrieval-encoder data construction and autoencoder training. More implementation details are provided in Appendix E.

Table 2: VBench-Long results for 30s, 60s, and 120s generation. Each block fixes a base model and compares it with ∞-RoPE (Yesiltepe et al., 2026), Deep Forcing (Yi et al., 2026), and LongLive-RAG. Bold/underline mark best/second-best values; Avg. Rank is averaged over the six metrics. Avg. Rank is computed from unrounded metric values.

Method	Subject Consistency ↑	Background Consistency ↑	Motion Smoothness ↑	Dynamic Degree ↑	Aesthetic Quality ↑	Imaging Quality ↑	Avg. Rank ↓
30s generation							
Self-Forcing (Huang et al., 2025b)	96.21	95.39	98.39	**52.03**	56.69	63.31	3.17
+ ∞-RoPE (Yesiltepe et al., 2026)	97.32	96.38	98.59	46.82	56.78	63.93	2.00
+ Deep Forcing (Yi et al., 2026)	97.04	96.02	98.57	38.85	56.44	61.91	3.50
+ LongLive-RAG (Ours)	**97.57**	**96.56**	**98.76**	42.24	**57.17**	**65.43**	**1.33**
LongLive (Yang et al., 2026a)	97.35	96.15	98.70	44.74	**59.38**	68.15	2.67
+ ∞-RoPE (Yesiltepe et al., 2026)	97.27	96.19	98.68	**48.18**	58.69	67.99	3.17
+ Deep Forcing (Yi et al., 2026)	97.52	**96.43**	**98.82**	41.46	59.00	67.61	2.50
+ LongLive-RAG (Ours)	**97.53**	96.39	98.77	44.84	59.24	**68.42**	**1.67**
Causal-Forcing (Zhu et al., 2026)	94.60	94.68	96.56	73.96	54.58	65.53	3.00
+ ∞-RoPE (Yesiltepe et al., 2026)	93.93	94.11	96.21	**90.83**	55.42	68.26	2.33
+ Deep Forcing (Yi et al., 2026)	93.52	93.86	95.84	84.79	55.03	66.07	3.33
+ LongLive-RAG (Ours)	**95.43**	**94.79**	**97.16**	82.29	**57.31**	**70.07**	**1.33**
60s generation							
Self-Forcing (Huang et al., 2025b)	95.84	95.27	98.20	**51.72**	56.05	62.22	3.33
+ ∞-RoPE (Yesiltepe et al., 2026)	97.24	96.24	98.58	46.64	56.09	63.28	2.17
+ Deep Forcing (Yi et al., 2026)	96.08	95.38	98.24	41.44	56.68	60.81	3.17
+ LongLive-RAG (Ours)	**97.60**	**96.51**	**98.70**	44.69	**57.19**	**64.97**	**1.33**
LongLive (Yang et al., 2026a)	97.13	95.89	98.61	44.56	58.17	67.56	2.83
+ ∞-RoPE (Yesiltepe et al., 2026)	97.00	95.85	98.53	**53.36**	57.48	66.94	3.33
+ Deep Forcing (Yi et al., 2026)	97.17	96.04	**98.73**	45.13	57.48	67.27	2.50
+ LongLive-RAG (Ours)	**97.32**	**96.08**	98.62	49.90	**58.30**	**67.79**	**1.33**
Causal-Forcing (Zhu et al., 2026)	93.52	94.12	95.74	72.32	51.24	62.30	3.83
+ ∞-RoPE (Yesiltepe et al., 2026)	93.81	93.78	96.09	**92.47**	54.42	67.50	2.50
+ Deep Forcing (Yi et al., 2026)	94.27	94.18	**96.62**	78.59	52.12	64.25	2.33
+ LongLive-RAG (Ours)	**94.29**	**94.24**	96.48	88.20	**54.95**	**68.16**	**1.33**
120s generation							
Self-Forcing (Huang et al., 2025b)	96.12	95.32	98.27	43.39	55.64	61.57	3.33
+ ∞-RoPE (Yesiltepe et al., 2026)	97.15	96.09	98.55	**46.29**	55.11	61.81	2.33
+ Deep Forcing (Yi et al., 2026)	96.92	**96.83**	**98.97**	15.23	52.84	57.93	2.83
+ LongLive-RAG (Ours)	**97.64**	96.40	98.75	44.10	**56.30**	**64.16**	**1.50**
LongLive (Yang et al., 2026a)	96.93	95.64	98.58	47.12	**57.90**	66.95	2.50
+ ∞-RoPE (Yesiltepe et al., 2026)	96.81	95.65	98.48	**53.59**	56.73	66.19	3.17
+ Deep Forcing (Yi et al., 2026)	97.17	95.72	98.57	46.03	56.98	66.03	3.00
+ LongLive-RAG (Ours)	**97.22**	**95.88**	**98.62**	50.25	57.57	**66.95**	**1.33**
Causal-Forcing (Zhu et al., 2026)	92.98	**94.66**	95.41	63.79	47.31	58.23	3.33
+ ∞-RoPE (Yesiltepe et al., 2026)	93.45	93.56	95.95	**93.44**	53.47	66.81	2.17
+ Deep Forcing (Yi et al., 2026)	93.19	94.04	95.60	76.45	46.86	61.58	3.17
+ LongLive-RAG (Ours)	**94.38**	94.08	**96.56**	90.21	**54.82**	**68.23**	**1.33**

Evaluation protocol. Following Self-Forcing++ (Cui et al., 2025), we evaluate on a 128-prompt subset of MovieGenBench (Polyak et al., 2024). Following Self-Forcing (Huang et al., 2025b), all prompts are refined using Qwen2.5-7B-Instruct (Qwen Team, 2024). Table 2 reports VBench-Long metrics (Huang et al., 2024) for 30s, 60s, and 120s generations. Table 4 reports ablations on 30s Causal-Forcing and auxiliary VLM scores. The VLM prompt is provided in Appendix F.

4.2 Results

Qualitative results. Figure 4 compares LongLive-RAG with alternative history-handling methods, including ∞-RoPE and Deep Forcing, on 30s rollouts. LongLive-RAG better preserves subject and background appearance, while other methods can show appearance shifts, duplicated subjects, or color artifacts. These examples support the same trend measured by Table 2: retrieving selected non-local latents provides useful context beyond the fixed local window or compressed history.

(a) Embedding space.

Method	Subj. Cons. ↑	Bg. Cons. ↑	Motion Smooth. ↑	Imaging Quality ↑
Random retrieval	94.54	94.32	96.81	68.79
Avg-pool desc.	94.77	94.49	96.76	69.11
AE only	<u>94.82</u>	94.49	96.87	<u>69.48</u>
AE + SeqDelta	94.76	<u>94.54</u>	<u>97.04</u>	69.14
Ours	**95.43**	**94.79**	**97.16**	**70.07**

(b) Auxiliary VLM evaluation. (Max: 5)

Variant	Causal Forc.	Self Forc.	LongLive
Base	2.60	3.50	4.65
+ ∞-RoPE	4.10	4.15	4.35
+ Deep Forcing	3.55	4.35	4.70
+ LongLive-RAG	**4.70**	**4.45**	**4.75**

(c) Retrieval budget.

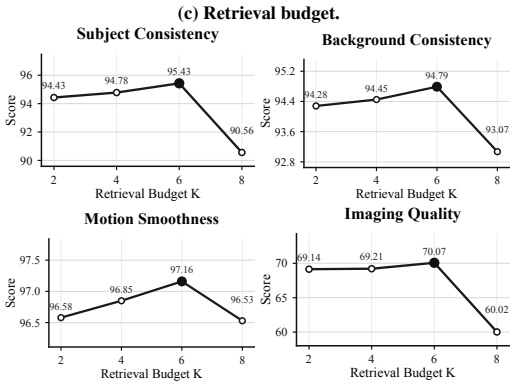


Table 4: Ablation study and auxiliary VLM evaluation. Left: embedding-space ablation and auxiliary VLM scores on 30s generations. Right: retrieval-budget trends under the same total attention budget.

Quantitative results. Table 2 shows that LongLive-RAG obtains the lowest Avg. Rank across all base-model and duration blocks. The gains are consistent across 30s, 60s, and 120s rollouts, suggesting that retrieval remains useful as generation length increases. Improvements are most visible on quality and consistency metrics, including subject consistency, background consistency, motion smoothness, and imaging quality. These results indicate that selected non-local latents help the generator preserve appearance and scene structure beyond the recent window.

Auxiliary VLM evaluation. Table 4 also reports VLM exposure scores on 30s generations as an auxiliary check. The scores show the same overall trend as the VBench-Long results.

4.3 Retrieved-context quality

Table 3: Quality of context latents.

Context	VAE cyc. loss ↓	CLIP-IQA ↑	MUSIQ ↑
Base	0.558	0.362	54.0
∞-RoPE	0.529	0.348	53.9
Deep Forcing	0.532	0.347	52.9
LongLive-RAG	**0.445**	**0.421**	**58.9**

To evaluate retrieved-context quality, at each generation step we decode the retrieved latents and compare them with the baseline. We measure VAE cycle loss as $\|\mathcal{E}_{\mathrm{VAE}}(\mathcal{D}_{\mathrm{VAE}}(z)) - z\|_2^2$, and report CLIP-IQA (Wang et al., 2023) and MUSIQ (Ke et al., 2021). This comparison and the ablations in Table 4 both use Causal-Forcing as the base generator. LongLive-RAG has the lowest cycle loss and highest image-quality scores.

4.4 Ablations

Embedding space. Table 4 compares retrieval embedding variants. Learned embeddings outperform random retrieval, average-pooled descriptors, and reconstruction-only embeddings. Adding $\mathcal{L}_{\mathrm{SeqDelta}}$ improves background consistency and motion smoothness over AE-only, while the full objective gives the strongest overall results across the retained metrics. These trends match the design in Section 3.3: retrieval benefits from a learned space that is both content-preserving and discriminative for non-local context.

Retrieval budget. With the embedding fixed, the retrieval-budget plots in Table 4 show that $K = 6$ gives the strongest consistency and imaging quality under the same total attention budget. This setting allocates enough slots to retrieved non-local context while preserving local context for smooth continuation. It therefore provides a practical balance between long-range reference and video continuity. Appendix G provides additional examples.

REFERENCES

bloc97. NTK-aware scaled RoPE allows LLaMA models to have extended (8k+) context size without any fine-tuning and minimal perplexity degradation. Reddit post, 2023. URL https://www.reddit.com/r/LocalLLaMA/comments/14lz7j5/.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 2206–2240, 2022.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. Technical report, OpenAI, 2024. URL https://openai.com/index/video-generation-models-as-world-simulators/.

Shengqu Cai, Ceyuan Yang, Lvmin Zhang, Yuwei Guo, Junfei Xiao, Ziyan Yang, Yinghao Xu, Zhenheng Yang, Alan Yuille, Leonidas Guibas, Maneesh Agrawala, Lu Jiang, and Gordon Wetzstein. Mixture of contexts for long video generation. In *International Conference on Learning Representations*, 2026.

Mathilde Caron, Hugo Touvron, Ishan Misra, Herve Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

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

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

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023.

Shuo Chen, Cong Wei, Sun Sun, Ping Nie, Kai Zhou, Ge Zhang, Ming-Hsuan Yang, and Wenhu Chen. Context forcing: Consistent autoregressive video generation with long context. *arXiv preprint arXiv:2602.06028*, 2026.

Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Self-forcing++: Towards minute-scale high-quality video generation. *arXiv preprint arXiv:2510.02283*, 2025.

Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Lol: Longer than longer, scaling video generation to hour. *arXiv preprint arXiv:2601.16914*, 2026.

Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=JE9tCwe3lp.

Mohamed Elmoghany, Ryan Rossi, Seunghyun Yoon, Subhojyoti Mukherjee, Eslam Mohamed Bakr, Puneet Mathur, Gang Wu, Viet Dac Lai, Nedim Lipka, Ruiyi Zhang, et al. A survey on long-video storytelling generation: architectures, consistency, and cinematic quality. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp. 7082–7094, 2025.

Ruili Feng, Han Zhang, Zhantao Yang, Jie Xiao, Zhilei Shu, Zhiheng Liu, Andy Zheng, Yukun Huang, Yu Liu, and Hongyang Zhang. The matrix: Infinite-horizon world generation with real-time moving control. *arXiv preprint arXiv:2412.03568*, 2024.

Ravi Ghadia, Avinash Kumar, Gaurav Jain, Prashant Nair, and Poulami Das. Dialogue without limits: Constant-sized kv caches for extended responses in llms. *arXiv preprint arXiv:2503.00979*, 2025.

Google DeepMind. Gemini 3.1 pro model card, 2026. URL https://deepmind.google/models/model-cards/gemini-3-1-pro/.

Yuchao Gu, Weijia Mao, and Mike Zheng Shou. Long-context autoregressive video modeling with next-frame prediction. *arXiv preprint arXiv:2503.19325*, 2025.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 3929–3938, 2020.

Roberto Henschel, Levon Khachatryan, Hayk Poghosyan, Daniil Hayrapetyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2568–2577, 2025.

Yicong Hong, Yiqun Mei, Chongjian Ge, Yiran Xu, Yang Zhou, Sai Bi, Yannick Hold-Geoffroy, Mike Roberts, Matthew Fisher, Eli Shechtman, et al. Relic: Interactive video world model with long-horizon memory. *arXiv preprint arXiv:2512.04040*, 2025.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

Junchao Huang, Xinting Hu, Boyao Han, Shaoshuai Shi, Zhuotao Tian, Tianyu He, and Li Jiang. Memory forcing: Spatio-temporal memory for consistent scene generation on minecraft. *arXiv preprint arXiv:2510.03198*, 2025a.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=mSiN7i0BYH.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, 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.

Sihui Ji, Xi Chen, Shuai Yang, Xin Tao, Pengfei Wan, and Hengshuang Zhao. Memflow: Flowing adaptive memory for consistent and efficient long video narratives. *arXiv preprint arXiv:2512.14699*, 2025.

Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong MU, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=66NzcRQuOq.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5148–5157, 2021.

Youngrae Kim, Qixin Hu, C-C Jay Kuo, and Peter A Beerel. Memrope: Training-free infinite video generation via evolving memory tokens. *arXiv preprint arXiv:2603.12513*, 2026.

Akio Kodaira, Tingbo Hou, Ji Hou, Markos Georgopoulos, Felix Juefei-Xu, Masayoshi Tomizuka, and Yue Zhao. Streamdit: Real-time streaming text-to-video generation. *arXiv preprint arXiv:2507.03745*, 2025.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Haodong Li, Shaoteng Liu, Zhe Lin, and Manmohan Chandraker. Rolling sink: Bridging limited-horizon training and open-ended testing in autoregressive video diffusion. *arXiv preprint arXiv:2602.07775*, 2026.

Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 25690–25699, 2025.

Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. Snapkv: Llm knows what you are looking for before generation. *Advances in Neural Information Processing Systems*, 37:22947–22970, 2024.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time. *arXiv preprint arXiv:2509.25161*, 2025.

Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024.

Yunhong Lu, Yanhong Zeng, Haobo Li, Hao Ouyang, Qiuyu Wang, Ka Leong Cheng, Jiapeng Zhu, Hengyuan Cao, Zhipeng Zhang, Xing Zhu, et al. Reward forcing: Efficient streaming video generation with rewarded distribution matching distillation. *arXiv preprint arXiv:2512.04678*, 2025.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*, 2023.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Xuanchi Ren, Yifan Lu, Tianshi Cao, Ruiyuan Gao, Shengyu Huang, Amirmojtaba Sabour, Tianchang Shen, Tobias Pfaff, Jay Zhangjie Wu, Runjian Chen, et al. Cosmos-drive-dreams: Scalable synthetic driving data generation with world foundation models. *arXiv preprint arXiv:2506.09042*, 2025.

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

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Wenqiang Sun, Haiyu Zhang, Haoyuan Wang, Junta Wu, Zehan Wang, Zhenwei Wang, Yunhong Wang, Jun Zhang, Tengfei Wang, and Chunchao Guo. Worldplay: Towards long-term geometric consistency for real-time interactive world modeling. *arXiv preprint arXiv:2512.14614*, 2025.

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

Zhongwei Wan, Xinjian Wu, Yu Zhang, Yi Xin, Chaofan Tao, Zhihong Zhu, Xin Wang, Siqi Luo, Jing Xiong, Longyue Wang, and Mi Zhang. D2o: Dynamic discriminative operations for efficient long-context inference of large language models. *arXiv preprint arXiv:2406.13035*, 2024.

Jianyi Wang, Kelvin C. K. Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Zheng Wang, Boxiao Jin, Zhongzhi Yu, and Minjia Zhang. Model tells you where to merge: Adaptive kv cache merging for llms on long-context tasks. *arXiv preprint arXiv:2407.08454*, 2024.

Ruiqi Wu, Xuanhua He, Meng Cheng, Tianyu Yang, Yong Zhang, Zhuoliang Kang, Xunliang Cai, Xiaoming Wei, Chunle Guo, Chongyi Li, et al. Infinite-world: Scaling interactive world models to 1000-frame horizons via pose-free hierarchical memory. *arXiv preprint arXiv:2602.02393*, 2026.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NG7sS51zVF.

Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. Worldmem: Long-term consistent world simulation with memory. *arXiv preprint arXiv:2504.12369*, 2025.

Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Ying-Cong Chen, Yao Lu, Song Han, and Yukang Chen. Longlive: Real-time interactive long video generation. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=nCAODkpsPJ.

Ying Yang, Zhengyao Lv, Yujia Zeng, Tianlin Pan, Haofan Wang, Yueming Lyu, Binxin Yang, Hubery Yin, Chen Li, Jing Lyu, Ziwei Liu, and Chenyang Si. Stableworld: Towards stable and consistent long interactive video generation. *arXiv preprint arXiv:2601.15281*, 2026b.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

Hidir Yesiltepe, Tuna Meral, Adil Kaan Akan, Kaan Oktay, and Pinar Yanardag. Infinity-RoPE: Action-Controllable Infinite Video Generation Emerges From Autoregressive Self-Rollout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 40256–40265, June 2026.

Jung Yi, Wooseok Jang, Paul Hyunbin Cho, Jisu Nam, Heeji Yoon, and Seungryong Kim. Deep Forcing: Training-Free Long Video Generation with Deep Sink and Participative Compression. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=gtmyFnvXAW.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22963–22974, 2025.

Jiwen Yu, Jianhong Bai, Yiran Qin, Quande Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Context as memory: Scene-consistent interactive long video generation with memory retrieval. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–11, 2025a.

Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10459–10469, 2023.

Lijun Yu, Jose Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David A Ross, and Lu Jiang. Language model beats diffusion - tokenizer is key to visual generation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gzqrANCF4g.

Yifei Yu, Xiaoshan Wu, Xinting Hu, Tao Hu, Yangtian Sun, Xiaoyang Lyu, Bo Wang, Lin Ma, Yuewen Ma, Zhongrui Wang, et al. Videossm: Autoregressive long video generation with hybrid state-space memory. *arXiv preprint arXiv:2512.04519*, 2025b.

Shangjin Zhai, Zhichao Ye, Jialin Liu, Weijian Xie, Jiaqi Hu, Zhen Peng, Hua Xue, Danpeng Chen, Xiaomeng Wang, Lei Yang, et al. Stargen: A spatiotemporal autoregression framework with video diffusion model for scalable and controllable scene generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26822–26833, 2025.

Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models. *arXiv preprint arXiv:2504.12626*, 2025a.

Lvmin Zhang, Shengqu Cai, Muyang Li, Chong Zeng, Beijia Lu, Anyi Rao, Song Han, Gordon Wetzstein, and Maneesh Agrawala. Tinyhistory: Lightweight video history embeddings via two-stage context learning. *arXiv preprint arXiv:2512.23851*, 2025b.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Yuxin Zhang, Yuxuan Du, Gen Luo, Yunshan Zhong, Zhenyu Zhang, Shiwei Liu, and Rongrong Ji. Cam: Cache merging for memory-efficient llms inference. In *Forty-first international conference on machine learning*, 2024.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, et al. H2o: Heavy-hitter oracle for efficient generative inference of large language models. *Advances in Neural Information Processing Systems*, 36:34661–34710, 2023.

Min Zhao, Guande He, Yixiao Chen, Hongzhou Zhu, Chongxuan Li, and Jun Zhu. RIFLEx: A free lunch for length extrapolation in video diffusion transformers. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=v3B79m7t8Z.

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

AI USE STATEMENT

We used a generative AI assistant to draft and polish portions of the manuscript based on author-provided research content, and to assist with text organization, citation placement, and LaTeX formatting. We did not use AI to develop research ideas or methods, design or conduct experiments, generate data, or prove mathematical claims. The authors reviewed all AI-assisted text and take responsibility for the final manuscript.

A CONCLUSION AND LIMITATIONS

We presented LongLive-RAG, a simple way for an AR video generator to look back at useful parts of the video it has already generated. Instead of relying only on the recent window, LongLive-RAG searches the generated history and brings back relevant context for the next block. We train the retrieval embeddings with reconstruction, Window Temporal Delta Loss, and smoothing so that the search keys preserve visual content, avoid redundant nearby matches, and remain stable over time. Experiments across multiple AR backbones and video lengths show that this improves long-video quality with little retrieval overhead.

Limitations. LongLive-RAG builds on a frozen base checkpoint. It improves how the model selects and reuses generated history, but it does not change the generator itself. As a result, the final video quality is still bounded by the capability of the base AR model.

B BROADER IMPACTS

LongLive-RAG is a research method for improving long-horizon consistency in AR text-to-video generation. It may support positive applications that require stable long video synthesis, such as creative tools, simulation, and interactive content generation. At the same time, improvements in long-video quality can inherit the misuse risks of the underlying video generator, including deceptive or misleading generated media. LongLive-RAG does not introduce a new base generator or a new data source; it changes how a frozen AR generator selects and reuses its generated history. Responsible use should therefore follow the license, usage terms, and safety practices of the underlying video generation models.

C ADDITIONAL RELATED WORK DISCUSSION

Video generation and AR rollout. Modern video generators include token-based models, diffusion transformers, and large-scale text-to-video systems (Yan et al., 2021; Yu et al., 2023; 2024; Kondratyuk et al., 2023; Deng et al., 2025; Jin et al., 2025; Team Wan et al., 2025; Polyak et al., 2024; Kong et al., 2024; Yang et al., 2024; Sand.ai et al., 2025; Chen et al., 2025). Many of these models denoise a fixed clip jointly, which ties computation to clip length. AR video generation instead emits frames or latent blocks causally, enabling streaming, interactive prompting, and variable-length output (Chen et al., 2024; Yin et al., 2025; Huang et al., 2025b; Liu et al., 2025; Yang et al., 2026a; Kodaira et al., 2025; Henschel et al., 2025; Gu et al., 2025; Zhai et al., 2025). LongLive-RAG is designed for this AR setting, where the generated trajectory itself becomes the source of future context.

Causal adaptation and self-generated context. Several methods convert or adapt pretrained video diffusion models into causal generators through distillation, causal teacher–student training, or self-generated context exposure (Yin et al., 2024; 2025; Huang et al., 2025b; Cui et al., 2025; Zhu et al., 2026; Chen et al., 2026; Liu et al., 2025; Yang et al., 2026a; Cui et al., 2026). These techniques reduce exposure bias and improve each local denoising step under causal rollout. LongLive-RAG addresses a complementary axis: given a causal generator and its self-generated trajectory, how should non-local context be selected at inference time? It leaves the denoiser unchanged and modifies only the context assembly policy.

Context windows, anchors, and positional extrapolation. The simplest scalable policy is a sliding window, which bounds memory by keeping only recent context (Yin et al., 2025; Huang et al.,

2025b; Liu et al., 2025; Yang et al., 2026a). Attention sinks and anchor-based variants preserve early tokens or frames to improve stability, but the chosen anchors are not necessarily the content needed by a later query (Xiao et al., 2024; Yang et al., 2026a; Li et al., 2026; Lu et al., 2025). A related line extends positional encodings so that models can operate beyond their trained length, including RoPE-based positional extrapolation and video-specific variants (Su et al., 2024; Chen et al., 2023; Peng et al., 2023; bloc97, 2023; Zhao et al., 2025; Yesiltepe et al., 2026). These methods increase the feasible context horizon or stabilize attention, but their selection rule is still primarily position-based. LongLive-RAG instead uses content-addressed retrieval to decide which historical entries are exposed to attention.

Compressed-history tokens and recurrent history. Methods based on compressed-history tokens and cache management summarize, select, or evict long histories through substitute tokens, recurrent states, compact sequence representations, or KV-cache policies (Zhang et al., 2023; Li et al., 2024; Wan et al., 2024; Ghadia et al., 2025; Yi et al., 2026; Yu et al., 2025b; Kim et al., 2026; Zhang et al., 2025b;a; Wang et al., 2024; Zhang et al., 2024). Such designs are attractive because they decouple memory cost from the full history length. Their limitation for generation is that the model often attends to the summary itself rather than to the original context produced by the generator. If the summary drops a rare object, identity cue, or background detail, a later denoising step cannot recover it directly. LongLive-RAG also uses compact representations, but only as search keys; after retrieval, the generator receives matched context entries in the backbone's native cache format. This separates the compression needed for indexing from the information used for denoising.

Retrieval-augmented generation. Retrieval-augmented generation retrieves external evidence or memory and conditions a generator on the retrieved content (Lewis et al., 2020; Guu et al., 2020; Borgeaud et al., 2022). These methods are typically used with a fixed text corpus or database to improve factual grounding or long-context access. LongLive-RAG uses the same high-level retrieval principle, but the retrieval source is different: it searches self-generated video latents produced during the current AR rollout. The retrieved items are also inserted as native generator context rather than as external text evidence. This makes retrieval a context-selection mechanism for long video generation.

Retrieval memory in video generation. Retrieval memory has been explored in world models and embodied video settings where explicit structure is available. Memory can be organized by camera pose, 3D geometry, scene coordinates, or field-of-view overlap, making it possible to identify relevant observations in a physically grounded way (Hu et al., 2023; Feng et al., 2024; Li et al., 2025; Yu et al., 2025a; Xiao et al., 2025; Sun et al., 2025; Yang et al., 2026b; Wu et al., 2026). Open-ended text-to-video generation does not provide such explicit retrieval cues by default: the camera can move freely, objects can reappear after occlusion, and no pose graph indicates which historical state should be reused. LongLive-RAG therefore retrieves from the generator's own latent trajectory and learns the search geometry from generated content.

D Why Retrieval in Latent Space?

Most contemporary video diffusion systems denoise in a VAE latent space and decode only after the latent rollout is complete (Team Wan et al., 2025; Polyak et al., 2024; Kong et al., 2024; Yang et al., 2024). A raw-pixel retrieval pipeline would break this execution pattern. After every completed AR block, the system would need to immediately run VAE decoding before search, then either store decoded frames or extract pixel-space features for all historical candidates. This inserts an extra decode-and-transfer path into each generation step, increasing memory traffic and communication overhead relative to the standard latent-first pipeline.

Latent retrieval also gives a simpler learning problem. The generator's latents already carry visual and semantic information in a representation aligned with the denoising model and its attention context. Training a lightweight encoder on these latents can therefore focus on making the search space discriminative for recurrence. In contrast, a pixel-space retriever must learn from decoded frames, which are higher-dimensional and less directly tied to the generator's internal state. Using an off-the-shelf image feature space is not a clean substitute: such features can be semantically broad but insufficiently discriminative for long generated videos, causing the nearest neighbors to remain overly local in time, similar to the collapse observed with reconstruction-only latent compression.

Figure 5 illustrates this behavior with DINO features (Caron et al., 2021) computed from decoded frames: the similarity structure is dominated by local neighborhoods rather than clean long-range matches.

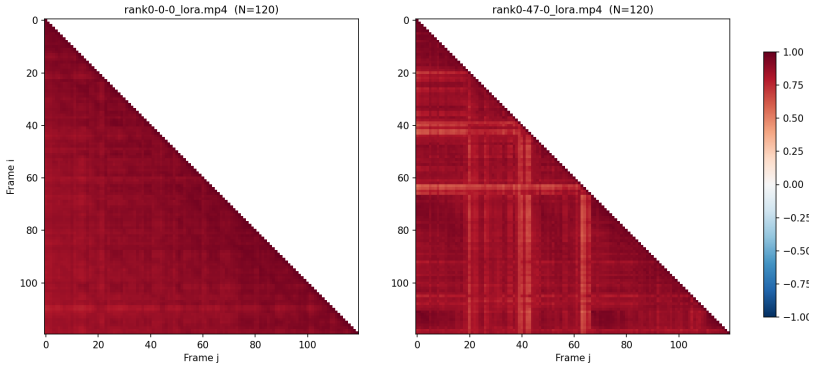


Figure 5: Raw-pixel retrieval analysis using DINO features (Caron et al., 2021) from decoded frames. The similarity map shows that off-the-shelf image features tend to retrieve temporally local neighbors, making them less suitable for long generated videos where useful context may be far outside the recent window.

Finally, the mapping between latent states and decoded pixels is not one-to-one for retrieval. The same or similar decoded appearance can correspond to different latent states that carry different denoising histories, context associations, or future trajectory implications. A pixel-space match can therefore be ambiguous when mapped back to the latent/context objects needed by the AR generator. LongLive-RAG avoids this mismatch by indexing the objects the generator actually produces and reuses: self-generated latents and their original context.

E Experiment Details

This appendix section records the experimental details used for LongLive-RAG.

Training data construction. We use the prompt pool from Self-Forcing (Huang et al., 2025b). We randomly sample 10% of the prompts with seed 0. For each selected prompt, we generate 90-latent LongLive rollouts with a frozen generator (Yang et al., 2026a). We use LongLive because it provides long AR rollout latents with history-dependent states in the target latent space. This choice is used only for collecting training latents for the retrieval encoder; it is not used to filter evaluation prompts or select reported samples. We save the clean denoised latent from each completed AR block and use these latents to train the autoencoder. Generating the resulting 20,000+ training latents takes about 7 days on 6 NVIDIA RTX A6000 GPUs. This data-construction stage is separate from the inference and evaluation runs reported in Section 4.1. Decoded videos are used only for sanity checking.

Why single base model? Our choice is to train one retrieval encoder and use it unchanged across all evaluated backbones. A natural alternative is to train a separate retrieval encoder for each target AR backbone, or to train on rollouts collected from all base models. We deliberately avoid this setting. LongLive-RAG is designed as a general RAG framework; per-backbone retrieval would make the method tied to each new AR generator. We therefore train one encoder in the shared WAN VAE latent space used by many recent AR video backbones, including Causal-Forcing, Self-Forcing, and LongLive (Zhu et al., 2026; Huang et al., 2025b; Yang et al., 2026a). Although these backbones may have different rollout distributions, their generated latents are decoded by the same WAN VAE and therefore share a common latent coordinate system. The same frozen encoder is used for all evaluated backbones in Table 2, without per-backbone retraining or data-size tuning.

Latent autoencoder training. The base video generators are frozen, and only the latent autoencoder is trained. The autoencoder is a convolutional encoder–decoder applied to each latent block independently. The encoder uses stride-2 convolutions, GroupNorm, SiLU activations, residual

blocks, global average pooling, and a linear projection to the retrieval embedding. The decoder maps the embedding back to a bottleneck feature map and reconstructs the latent with nearest-neighbor upsampling, convolutions, GroupNorm, SiLU activations, and residual blocks. The decoder has no output activation, and the output is cropped to the target latent size. For retrieval, we use the encoder output with L2 normalization.

We train with AdamW. The objective follows Section 3.3: mean-squared reconstruction loss, Window Temporal Delta Loss, and trajectory-smoothing loss. We split the latent dataset into training and validation sets, using 10% for validation. We track reconstruction, sequence-delta, smoothness, and total loss on both splits. The best checkpoint is selected by the lowest validation total loss. The retrieval autoencoder training job takes less than 100 NVIDIA RTX A6000 GPU hours after the training latents have been generated.

Table 5 summarizes the main hyperparameters. The values are read from the AE training config and the LongLive-RAG inference configs.

Table 5: Main hyperparameters used in LongLive-RAG.

parameter	**value**	**description**
Optimizer	AdamW	optimizer for training the latent autoencoder
Sequence length	8	continuous latent blocks per training chunk
Embedding dimension	1024	retrieval embedding dimension
Hidden dimensions	[64, 128, 256]	encoder and decoder channel dimensions
Learning rate	3×10^{-4}	learning rate for AdamW
Weight decay	1×10^{-4}	optimizer weight decay
Batch size	128	training batch size
Training epochs	400	total training epochs
λ_{rec}	1	reconstruction loss weight
λ_{Δ}	1.0	Window Temporal Delta Loss weight
Delta window w	3	local window for temporal delta pairs
Delta margin m	0.85	cosine-similarity margin
λ_{smooth}	1	trajectory-smoothing loss weight
Gradient clipping	1.0	maximum gradient norm
Mixed precision	enabled	mixed-precision training
Checkpoint selection	lowest val. total loss	validation total is the sum of tracked loss terms
Recency guard R	5	number of recent blocks excluded from retrieval

Evaluation protocol. We evaluate generated videos with the official VBench-Long scripts (Huang et al., 2024). The reported inference and evaluation runs for Table 2 and Table 4 take about one week on 6 NVIDIA RTX A6000 GPUs, excluding the retrieval-encoder data-construction and autoencoder-training stages described above. Runtime overhead is measured on one NVIDIA RTX A6000, with latent encoding and top-K search timed separately. We also conduct an auxiliary VLM evaluation on 30s generations.

F Auxiliary VLM Evaluation

VLM evaluation. We use Gemini 3.1-Pro (Google DeepMind, 2026) as the VLM judge to score each generated video. Following Deep Forcing and Self-Forcing++ (Yi et al., 2026; Cui et al., 2025), we randomly sample 20 cases from the 30s generations for this auxiliary evaluation. The VLM is given the generated video and returns an aggregate score following the prompt below. Based on the original VLM scoring setup, we revise the system prompt by adding system instructions, a scoring example, and an explicit output format. We average scores over the evaluated prompt set and report the mean in Table 4 as an auxiliary check.

VLM Evaluation Prompt

You are an expert video quality assessor. Your task is to evaluate the "Overall Exposure Quality" of the provided video file.
Since exposure can change over time, you must watch the entire video and provide a single aggregate score based on the severity and duration of any exposure issues.

Scoring Scale

0: Catastrophic. The video is dominated by pure white (blown out) or pure black (crushed) frames for a significant duration, rendering the content largely unreadable.
1: Severe. Large portions of the video suffer from severe over- or under-exposure, substantially impairing visibility and ruining the viewer's experience.
2: Noticeable. Persistent clipping in highlights or shadows throughout the video, or frequent noticeable shifts that cause a significant loss of detail.
3: Moderate. Exposure is generally acceptable, but contains brief periods of over/under-exposure, or constant but minor highlight/shadow clipping that preserves core visual information.
4: Minor. Exposure is mostly excellent. Small regions are occasionally too bright or dark, or there is a very brief, negligible fluctuation, but it does not disrupt visibility.
5: Perfect. Rock-solid, well-balanced lighting across the entire frame and duration. No distracting over-exposure, darkening, or unnatural shifts.

Evaluation Guidelines

If the video has mixed quality (e.g., mostly a "5" but has a 2-second period of "1"), penalize the overall score based on how distracting that bad period is (e.g., an overall score of 2 or 3).
Pay attention to both the spatial extent (how much of the screen is affected) and the temporal extent (how long the issue lasts).

Output Format

First, provide a brief analysis summarizing the exposure across the video's duration (mentioning any specific moments of over/under-exposure if they exist). Then, output the final integer score wrapped in `<score></score>` tags.

Example:
Analysis: The first 5 seconds are perfectly exposed, but when the camera pans to the window, the sky becomes completely blown out for the remainder of the video, losing all highlight detail.
Score: `<score>3</score>`

G Ablation Examples

We provide qualitative examples for the retrieval-budget and embedding-space ablations. Figure 6 shows examples with different retrieval budgets. Figure 7 shows examples using different embedding spaces.

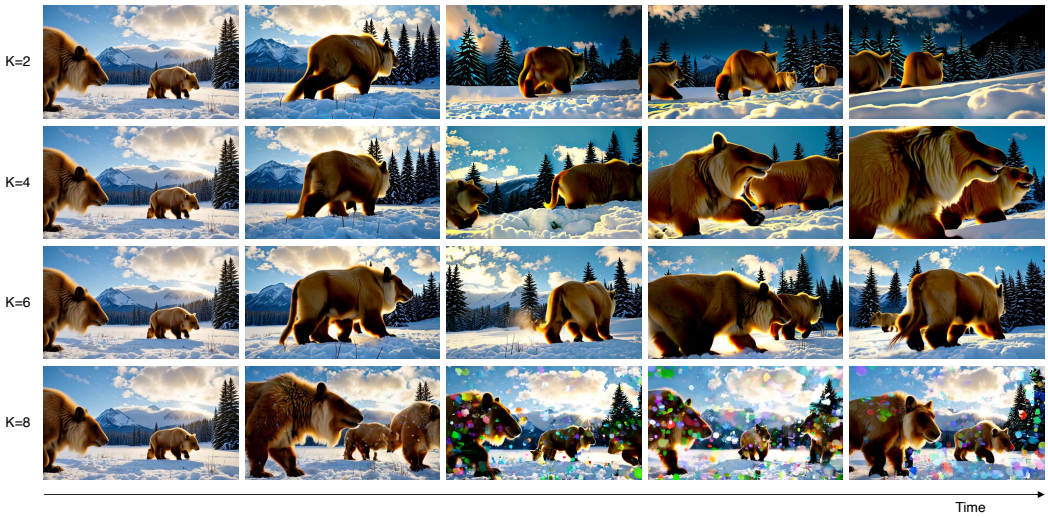


Figure 6: Retrieval-budget comparison. The balanced setting preserves both long-range references and video continuity.

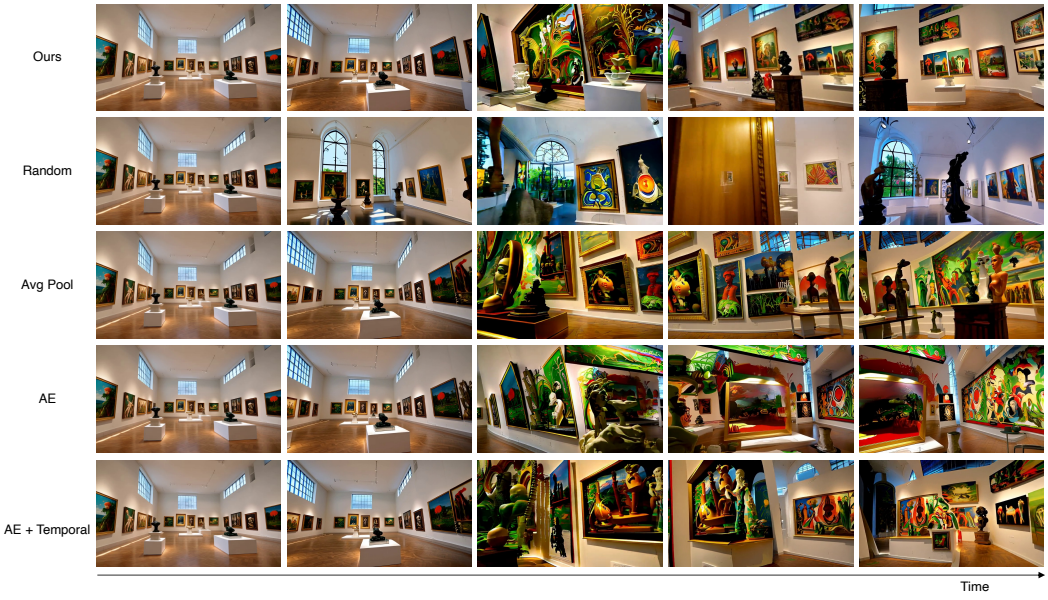


Figure 7: Embedding-space comparison. The learned LongLive-RAG embedding retrieves more useful historical context than random or hand-crafted alternatives.

H ADDITIONAL EVALUATIONS AND ANALYSES

We report two complementary evaluations based on MemFlow's multi-prompt scripts (Ji et al., 2025): direct return-to-scene performance on a 20-sequence subset and aggregate performance on the full 100-script test set.

Return-to-scene subset. We form 20 $A \to B \to A'$ sequences from MemFlow, where the final segment should recover the source subject or scene. For each segment s, we uniformly sample 16 frames F_s and average their CLIP image embeddings as c_s. Return Gap is $100[\max_m \cos(c_m, c_i) - \cos(c_j, c_i)]$, where i is the source, j the return, and m an intervening segment. LPIPS$(j \to i)$ is $|F_j|^{-1} \sum_{y \in F_j} \min_{x \in F_i} \mathrm{LPIPS}(y, x)$: each return frame is matched to its closest source frame (Zhang et al., 2018). Lower is better for both metrics. LongLive-RAG achieves the lowest Return Gap and LPIPS (Table 6).

Table 6: Return-to-scene on 20 $A \to B \to A'$ sequences.

Method	Return Gap ↓	LPIPS$(j \to i)$ ↓
Base	7.84	0.457
∞-RoPE	6.47	0.427
Deep Forcing	8.48	0.461
LongLive-RAG	**5.52**	**0.370**

Full 100-script test. We retain MemFlow's original protocol: 100 scripts, six prompts per video, and a prompt change every 10 seconds (Table 7).

Table 7: Results on the 100-script MemFlow multi-prompt test set. CLIP scores are reported for each of six prompt segments and averaged across segments. Best results are bold.

Method	Quality ↑	Consistency ↑	Aesthetic ↑	CLIP_1	CLIP_2	CLIP_3	CLIP_4	CLIP_5	CLIP_6	Mean ↑
Base	81.16	96.42	59.06	29.08	27.11	**27.62**	27.32	27.03	26.75	27.49
∞-RoPE	81.15	**96.73**	58.81	29.06	26.83	27.24	27.07	26.91	26.61	27.29
Deep Forcing	80.96	96.62	58.55	29.08	26.98	27.28	27.14	26.89	26.37	27.29
LongLive-RAG	**81.22**	96.57	**59.38**	**29.15**	**27.12**	27.49	**27.43**	**27.05**	**26.86**	**27.52**