LEARNING TO RETRIEVE: INTERNALIZING MEMORY RETRIEVAL FOR VIDEO WORLD MODELS

Anonymous authors
Paper under double-blind review

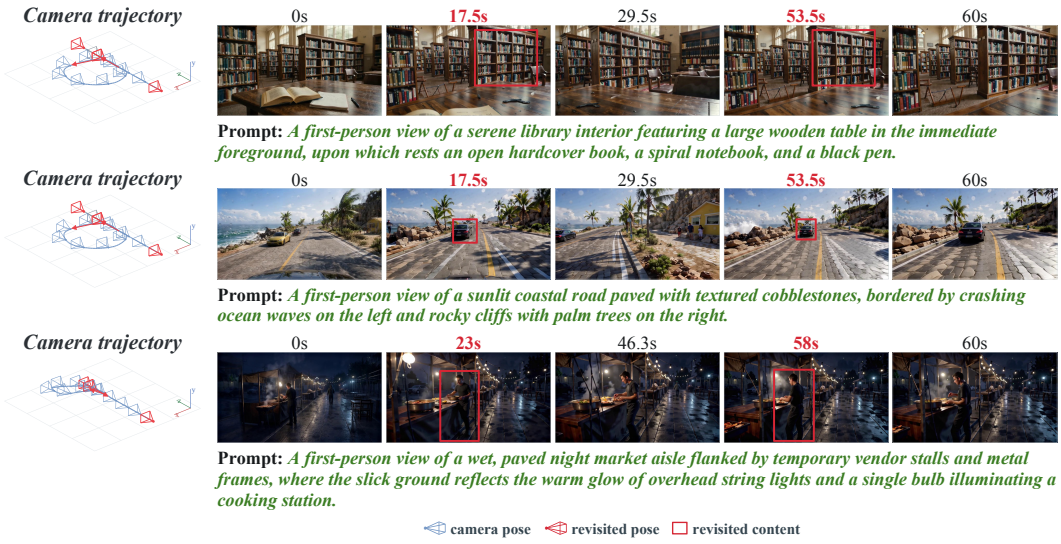


Figure 1: **Teaser demonstration.** We introduce Learning-to-Retrieve (L2R), which can generate consistent videos without external memory modules or 3D reconstruction pipelines. Red boxes mark content observed from matched camera poses separated by a long temporal interval, illustrating that L2R preserves scene identity when the camera revisits a previously observed region.

ABSTRACT

Video world models aim to generate explorable, 3D-consistent scene videos conditioned on camera trajectories. Existing approaches often rely on external memory systems that explicitly retrieve previously observed content to mitigate scene drift during long-horizon generation. However, these auxiliary memory pathways operate outside the model's internal generative dynamics, preventing the model from intrinsically learning when and what historical information should be retrieved. We propose to internalize memory retrieval into the generation process, allowing retrieval to emerge as an intrinsic behavior of the video world model rather than relying on an external memory system. Based on this principle, we introduce **Learning-to-Retrieve (L2R)**, which repurposes the model's persistent internal state as a memory for historical context. A camera-conditioned retrieval gate selectively accesses relevant historical information from this state, determining *what to retrieve*, while a retrieval trigger determines *when to retrieve*. We further supervise the trigger with a 3D re-visibility signal, activating retrieval when previously observed content re-enters the current view while otherwise preserving the existing context. Together, these components enable the model to intrinsically acquire memory retrieval behavior and incorporate relevant historical observations into generation without a separate retrieval pathway. Across multiple base models and camera-revisit benchmarks, L2R improves long-term scene consistency while eliminating the need for an external memory bank or 3D conditions.

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.

1 INTRODUCTION

Video world models aim to generate video sequences that explore a 3D scene according to a user-specified camera trajectory, conditioned on one or a small set of reference frames (Yu et al., 2025a; Xiao et al.; Zhu et al., 2026). Beyond perceptual realism and accurate camera control, a key requirement is scene-level consistency: upon revisiting previously observed regions, the generated content should remain consistent with earlier generations.

Existing strategies for enforcing scene consistency can be broadly grouped into two paradigms. Yu et al. (2025c); Huang et al. (2025); Hu et al. (2026); Zheng et al. (2026); Yang et al. (2026) formulate this problem as conditional generation, leveraging reconstruction models (Wang et al., 2025; 2026a) to produce conditions intended to improve the fidelity and coherence of the generated video. However, the effectiveness of this approach is closely tied to reconstruction accuracy. Reconstruction errors can propagate to downstream generation, degrading long-term quality.

The second paradigm integrates memory retrieval into the video generation model to improve long-term scene consistency. In these methods, long-term memory serves as an auxiliary context module: when past frames are spatially relevant to the target view, they are retrieved to augment the default short-term sliding-window context. The principal differences between these methods lie in the retrieval strategy. Earlier approaches (Li et al., 2025; Yu et al., 2025a; Xiao et al.; Yu et al., 2026b; Wang et al., 2026b) select historical context via camera overlap or explicit 3D signals and supply the retrieved views to the generator. More recent approaches make retrieval learnable, but still rely on separate memory access pathways. For example, MemLearner (Yu et al., 2026a) queries historical context using learnable query tokens, while CaR (Peng et al., 2026) adds a dedicated retrieval-attention branch on compressed context. Although such designs improve long-range scene consistency, their reliance on external or auxiliary memory access pathways introduces a mismatch between retrieval and the model's internal generative dynamics.

In this paper, we advance the perspective that ***video world models should intrinsically acquire the memory retrieval policy***. Rather than treating memory retrieval as an external context construction procedure, we aim to make retrieval part of the generation process itself. This requires two capabilities to be jointly learned: maintaining historical information that remains accessible over long horizons, and selectively accessing that information according to the current observation. By integrating these two capabilities into the generative pathway, memory access can directly adapt to the model's evolving generation dynamics.

To realize this design, we formulate memory retrieval within an attention mechanism with a persistent contextual state and selective state access. In this formulation, historical observations are continuously accumulated in the model's internal state, while the state-access mechanism determines which historical information is retrieved for the current generation. Our construction builds on the state-update structure (Katharopoulos et al., 2020; Yang et al., 2024; 2025b; Pan et al., 2025), but repurposes it for memory retrieval. A retrieval trigger determines *when to retrieve*, while a camera-conditioned gate determines *what historical information to retrieve*, jointly coupling memory access with the current view. This shift repurposes the persistent attention state as an internal memory from which historical context can be selectively retrieved during long-horizon generation.

Building on this observation, we introduce **L2R**, a mechanism designed to approximate memory retrieval in video world models. L2R employs a camera-conditioned gate to selectively access historical information from the accumulated state according to the current viewpoint. In addition, L2R incorporates a retrieval trigger that determines whether a given token should initiate retrieval: when the current view revisits previously observed content, retrieval is activated; otherwise, the model relies on the existing context. We supervise the retrieval trigger using the 3D signal based on *visibility re-entry*: tokens whose content exits the field of view or becomes occluded and subsequently reappears are annotated as retrieval-relevant. This supervision encourages the model to identify situations in which historical context is necessary. Collectively, L2R enables attention to acquire retrieval behavior intrinsically: the trigger learns *when to retrieve*, and the camera-conditioned gate learns *what to retrieve*, thereby coupling the current view with access to historical context without introducing a separate retrieval pathway or requiring explicit 3D correspondence at inference time.

We evaluate L2R on scene exploration with re-appear and loop-closure trajectories under multiple state-update rules. Given a reference image and a camera trajectory, video world models using L2R maintain scene consistency in the generated video without degrading visual quality or camera

controllability. Both quantitative and qualitative results demonstrate that the method outperforms existing approaches in terms of consistency and exhibits robust generalization capabilities to unseen camera trajectories. Furthermore, ablation studies validate the effectiveness of the proposed design.

2 Related Work

Memory in video generation. In video generation, particularly in autoregressive (AR) settings, the incorporation of an explicit memory is critical for maintaining and exploiting long-term consistency. Existing systems represent and utilize historical information through a variety of mechanisms, including sliding-window strategies (Chen et al., 2026), attention sinks (Yang et al., 2025a), positional extrapolation (Zhao et al., 2025; Yesiltepe et al., 2026), state-space architecture (Yu et al., 2025d), token compression of past context (Zhang & Agrawala, 2025), and hybrid combinations of these techniques (Yi et al., 2025). More recently, ARL^2 (Li et al., 2026) introduced a memory mechanism for video generation, using the recurrent state as the memory bank. However, it lacks the ability to perform conditioned retrieval, thereby compromising the model's memory capabilities.

Video world models are designed to generate scene-level videos that are both temporally coherent and dynamically consistent, typically using video diffusion models conditioned on camera trajectories or action controls. In these scenarios involving camera pose or 3D data, memory can serve not only as an external component but can also be retrieved. Many existing methods (Yu et al., 2025c;b; Hu et al., 2026; Yang et al., 2026; Zheng et al., 2026; Wei et al., 2026; Xu et al., 2026b) incorporate substantial 3D information, such as depth maps, point clouds, or 3D Gaussian splatting (3DGS), as an additional conditioning memory for the diffusion process. A complementary research direction (Li et al., 2025; Yu et al., 2025a; Xiao et al.; Wu et al., 2026; Xu et al., 2026a; Zhu et al., 2026) investigates the memory retrieval and compression mechanisms that operate outside the diffusion backbone. These methods store and retrieve historical frames based on camera-view overlap (Yu et al., 2025a) or index past observations via geometric associations to observed three-dimensional surface segments (Li et al., 2025). In addition, recent methods (Peng et al., 2026; Wang et al., 2026b; Yu et al., 2026a) introduce external learnable modules to retrieve memory for video world models, but keep retrieval separate from the generation model. Wonder (Xu et al., 2026a) uses sparse attention, allowing inference to attend to a small subset of context tokens. However, its top-k selection offers no principled guarantee that the chosen tokens are the most geometrically relevant.

3 Method

In contrast to previous methods, we propose **L2R**, a mechanism to accurately retrieve the historical context within the model, thus promoting improved consistency in the generated world videos. Sec. 3.1 presents the necessary preliminaries. Sec. 3.2 specifies the implementation of our L2R and explains how it retrieves the historical context. Sec. 3.3 summarizes the training pipeline.

3.1 Preliminaries

Memory retrieval in video world model. Given a reference video $\{V_i\}_{i=1}^{M}$ and a novel camera trajectory P_f, a video world model ϱ generates the video V_f observed along the target trajectory.

When the target viewpoint shares a high degree of consistency with a previous reference viewpoint (e.g. significant FoV overlap or similar geometric textures), existing approaches (Yu et al., 2025a) often introduce an external memory retrieval module to identify the trajectory-relevant information from the reference video and use the retrieved content as a context for generation.

$$\mathcal{C}_f = \begin{cases} \mathrm{Retrieve}\Big(\{V_i\}_{i=1}^{f-1}, P_f\Big), & \text{if target view revisits content,} \\ \mathcal{C}_f^{\mathrm{base}}, & \text{otherwise,} \end{cases} \qquad V_f = \varrho(\mathcal{C}_f, P_f), \tag{1}$$

where $\mathcal{C}_f^{\mathrm{base}}$ denotes the default context used for generation.

Here, retrieval consists of two implicit decisions: whether historical information can be needed for the current view, and which historical information should be accessed when retrieval is activated. In existing methods, both decisions are made through an auxiliary memory access pathway outside the core generation process.

3.2 LEARNING TO RETRIEVE

Internalize memory retrieval. Inside the video diffusion transformer, video is compressed into a 1d token sequence $\{x_t\}_{t=1}^{N}$, where t denotes the index of the token and N is the sequence length. In this setting, Eq. 1 will be transformed into the following form:

$$c_t = \begin{cases} \text{Retrieve}(\{x_i\}_{i=1}^{t-1}, p_t), & \text{if } r_t = 1, \\ \{x_i\}_{i=t-w}^{t-1} & \text{if } r_t = 0, \end{cases} \qquad x_t = \varrho(c_t, p_t), \tag{2}$$

where $p_t \in \mathbb{R}^{12}$ is the camera Plücker ray of x_t. w is the size of the sliding window. r_t indicates whether memory retrieval is activated for x_t. It emulates the video world model by initiating retrieval when the geometric overlap between the target and context views is sufficiently high.

To internalize this retrieval, we store the historical context within the model state, allowing subsequent tokens to directly access the information from preceding tokens. A common practice is to encode such information in an internal recurrent state (Yang et al., 2024; 2025b; Pan et al., 2025; Yu et al., 2025d). Denoting the accumulated state before processing x_t as s_{t-1}, which encodes information accumulated from preceding tokens, the historical context c_t can therefore be read directly from this state.

To realize the retrieval defined in Eq. 2, we read the historical context directly from the accumulated state. Specifically, r_t determines whether retrieval is activated, while the pose-conditioned gate γ_t determines which information is retrieved. The resulting context c_t is then used to update the state:

$$c_t = \begin{cases} \gamma_t \; s_{t-1}, & \text{If } r_t = 1, \\ s_{t-1}, & \text{If } r_t = 0, \end{cases} \qquad s_t = \varphi(c_t, p_t), \tag{3}$$

where φ denotes the state transition rule.

Designing γ_t: camera-guided retrieval. As shown in Eq. 3, memory retrieval relies on the camera pose p_t as a condition. To this end, we introduce camera-guided retrieval for L2R:

$$\log \gamma_t \leftarrow \log \gamma_t \odot \big(1 + \tanh(W_g \phi_t)\big), \tag{4}$$

where feature $\phi_t \in \mathbb{R}^{D_p}$ is the projection of p_t.

To promote numerical stability in the gated retrieval procedure, following previous work (Yang et al., 2024; 2025b; Pan et al., 2025), we perform multiplicative gating computations in log space.

Designing r_t: retrieval trigger and re-visibility supervision. As shown in Eq. 3, the behavior of L2R-GLA is consistent with the memory module of a video world model: the memory retrieval operator is triggered only when the viewpoint associated with the current token exhibits a high degree of consistency with viewpoints already represented in the context. The variable r_t is introduced to parameterize and learn this retrieval activation.

We learn r_t under re-visibility supervision. As shown in Figure 2, for each frame in a training clip, the supervisory target for r_t is computed offline via a three-stage procedure: (1) estimating per-frame depth using a feed-forward geometric model (Wang et al., 2025; 2026a); (2) unprojecting each token in the frame to a 3D point using the estimated depth; and (3) re-projecting the 3D points of earlier frames into the current frame using the known camera trajectory. If a 3D surface point observed in the previous frame exits the frustum (or is occluded) and reappears in the current frame, the label $y_t = 1$ is assigned to the corresponding token. We optimize r_t using the following objective:

$$\mathcal{L}_{\text{r}} = \frac{1}{|\mathcal{V}|} \sum_{t \in \mathcal{V}} \text{BCE}_{w_+}(\hat{r}_t, \; y_t) \, , \tag{5}$$

where $\hat{r}_t$ is the sigmoid probability before binarization, $\mathcal{V}$ contains valid target-frame tokens, and $w_+ = 3$ is the positive-class weight. For numerical stability, we compute the weighted BCE directly from gate logits and mask invalid tokens and context frames. Since binarization is non-differentiable, we stop gradients through r_t and force the model to retrieve memory on this discrete trigger.

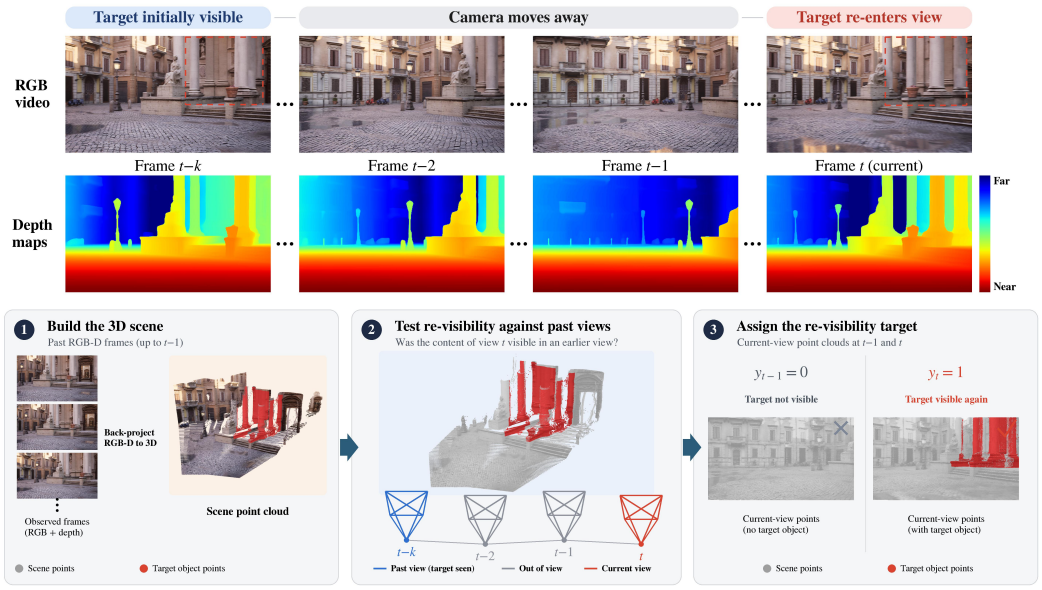


Figure 2: **Three stages to compute re-visibility supervision.** (1) estimate per-frame depth; (2) unproject each token to 3D points; and (3) re-project previous frames' 3D points into the current frame. If a previously observed 3D point leaves the frustum (or is occluded) and later reappears, assign $y_t = 1$ to the corresponding token.

Extension. Eq. 3 defines a mechanism-independent retrieval recipe that can be instantiated with different gated state transition mechanisms (Yang et al., 2024; 2025b). In this paper, we validate this by instantiating the same recipe with both Gated Linear Attention (GLA) (Yang et al., 2024) and Gated DeltaNet (GDN) (Yang et al., 2025b). For GDN, the instantiation requires an additional modification to its delta correction, and we retain its subtractive memory gating. Details of this adaptation are provided in Appendix B.

Instantiating L2R with these two state transition rules yields **L2R-GLA** and **L2R-GDN**, respectively. We use these names throughout the remainder of this paper.

3.3 Training

During model training, we support both training from scratch and supervised fine-tuning (SFT), depending on the experimental configuration required for evaluation. For most evaluation protocols, training is initialized from parameters loaded from a pre-trained video generation model.

Integration into a pretrained model. In every group of n blocks in the pre-trained camera-controllable video diffusion model, we replace the first $n-1$ self-attention layers with our memory layer (L2R-GLA). The remaining blocks retain softmax attention, so the attention captures the local spatial structure while the memory state provides temporal persistence. For each replacement, we initialize the memory layer's query, key, value, and output projections from the attention projections, and zero-initialize all camera-dependent paths so the layer initially matches the pre-trained token representation.

Objective. We optimize the rectified-flow objective together with re-visibility supervision:

$$\mathcal{L} = \mathcal{L}_{\text{denoise}} + \lambda_{\text{r}}\, \mathcal{L}_{\text{r}}, \tag{6}$$

where $\mathcal{L}_{\text{denoise}}$ is the timestep-weighted mean-squared error between the predicted flow velocity and $\epsilon - z_0$, where z_0 denotes the clean video latents and ϵ is Gaussian noise. This loss is computed only on target-frame latents; reference and context frames are excluded. The visibility loss $\mathcal{L}_{\text{r}}$ in Eq. 5 is normalized over valid target tokens within each memory layer and then averaged across memory layers. We use $\lambda_{\text{r}} = 0.05$ by default.

4 EXPERIMENTS

4.1 SETUP

We evaluate the performance of L2R-GLA and L2R-GDN along two dimensions: ***memory capacity*** and ***video quality***. The evaluation of the former characterizes the module's ability to maintain long-term consistency, whereas the evaluation of the latter verifies that this memory mechanism does not degrade visual fidelity or compromise camera controllability.

Memory capacity. Following Echo-Memory (King et al., 2026), we conduct a comprehensive evaluation of memory capabilities. We adopt Wan2.1-T2V-1.3B (Wan et al., 2025) as the base model and follow the camera-conditioned method introduced in Echo-Memory. To ensure experimental comparability, we use the static scene-exploration dataset (Yu et al., 2025a) and *all methods use exactly the same training and inference settings*. The implementation details are shown in Appendix A. Performance is evaluated using a three-branch protocol. (i) *Replay*: the model generates videos conditioned on given camera trajectories, and memory capacity is quantified using frame-aligned Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS). (ii) *In-domain loop closure*: outgoing and returning frames at matched camera poses are compared using the same image-quality metrics to measure revisit consistency. (iii) *Open domain return (O-V)*: scene-level consistency and object-identity preservation are assessed on first-frame probes released using the benchmark's vision language judge. This assesses the model's generalization capability to unseen trajectories and scenes.

Video quality. To isolate whether our memory modules improve long-horizon coherence *without* trading off perceptual fidelity, we curate a training corpus of 5-second clips *without camera loops*, sampled from RealEstate10K, DL3DV, and MiraData9K. For evaluation, we build a "forth-and-back" set on RealEstate10K and DL3DV, in which the second half explicitly retraces the camera path back to the start, yielding a controlled test of *out-of-domain* viewpoint regression under increasing drift. Visual quality is reported with VBench (Huang et al., 2024). Camera controllability is measured by pose agreement between the ground-truth trajectory and the trajectory re-estimated from generated frames using Pi3X (Wang et al., 2026a). We report rotation and translation errors (R_{err}, T_{err}) following VMem (Li et al., 2025). We instantiate the pipeline on Wan2.2-TI2V-5B and SANA-WM-2.6B (Zhu et al., 2026). Remaining implementation details are deferred to Appendix A.

4.2 RESULTS ON MEMORY CAPACITY

Table 1 compares memory capacity across replay, loop closure, and open-domain return. L2R-GLA performs best, ranking first on all replay metrics and on SSIM and O-V for loop closure and open-domain return. On replay, it raises PSNR from 13.71 (best baseline, VMem (Li et al., 2025)) to 14.74 (+1.03 dB), with higher SSIM and lower LPIPS. Versus Context as Memory (Yu et al., 2025a) at K=20, it improves PSNR by 2.20 dB (12.54→14.74), SSIM by 0.064 (0.449→0.513), and reduces LPIPS by 0.066 (0.496→0.430). The GLA baseline degrades in SSIM and LPIPS, likely because unconstrained retrieval causes memorization drift. Overall, integrating retrieval into the video generation model substantially improves consistency when previously observed content is revisited. L2R-GDN also shows strong performance, achieving 13.86 dB PSNR on replay and 69.37 O-V, which suggests that the proposed formulation remains effective when combined with GDN.

A similar trend is observed under loop closure. L2R-GLA achieves the best SSIM of 0.375 and the second-best PSNR of 13.14, closely matching the strongest baseline, VMem, at 13.17. It also substantially improves over Context as Memory at K=20, increasing PSNR from 11.07 to 13.14 and SSIM from 0.359 to 0.375. Although Context as Memory attains a lower LPIPS, L2R-GLA provides stronger pixel-level fidelity together with higher structural similarity. This indicates that the proposed internal retrieval mechanism is particularly effective at preserving the appearance and structure of previously observed content during viewpoint revisits.

For open-domain return, L2R-GLA achieves an O-V score of 72.57, outperforming VMem (Li et al., 2025) by 7.84 points and Context as Memory at K=20 by 13.94 points. Since this protocol evaluates held-out first-frame probes with distinctive objects, the improvement suggests that the learned internal retrieval policy can generalize beyond the training trajectories and recover previously observed scene content under unseen return scenarios. This further supports the ability of L2R to preserve scene identity without relying on an external memory bank.

Table 1: **Quantitative comparison on memory capacity.** Best in **bold**, second best underlined. * denotes methods are retrained because their released weights were not trained on dataset (Yu et al., 2025a). The results for the other methods are taken from Echo-Memory (King et al., 2026).

Method	Replay			Loop closure			O-V ↑
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	
I2V baseline (Wan2.1)	10.03	0.398	0.534	10.32	0.291	0.643	12.25
Spatial Memory	13.60	0.411	0.554	10.04	0.260	0.617	6.00
VideoSSM	12.69	0.344	0.581	12.23	0.298	0.584	34.75
ARL^{2} *	12.73	0.326	0.594	12.13	0.309	0.633	43.06
Context as Memory, K=5	11.92	0.408	0.501	10.72	0.307	0.596	50.75
Context as Memory, K=20	12.54	0.449	0.496	11.07	0.359	**0.543**	58.63
WorldMem*	12.34	0.471	0.536	11.59	0.348	0.594	59.74
VMem*	13.71	0.502	0.507	**13.17**	0.352	0.581	64.73
L2R-GLA (ours)	**14.74**	**0.513**	**0.430**	13.14	**0.375**	0.603	**72.57**
L2R-GDN (ours)	13.86	0.409	0.558	12.81	0.374	0.614	69.37

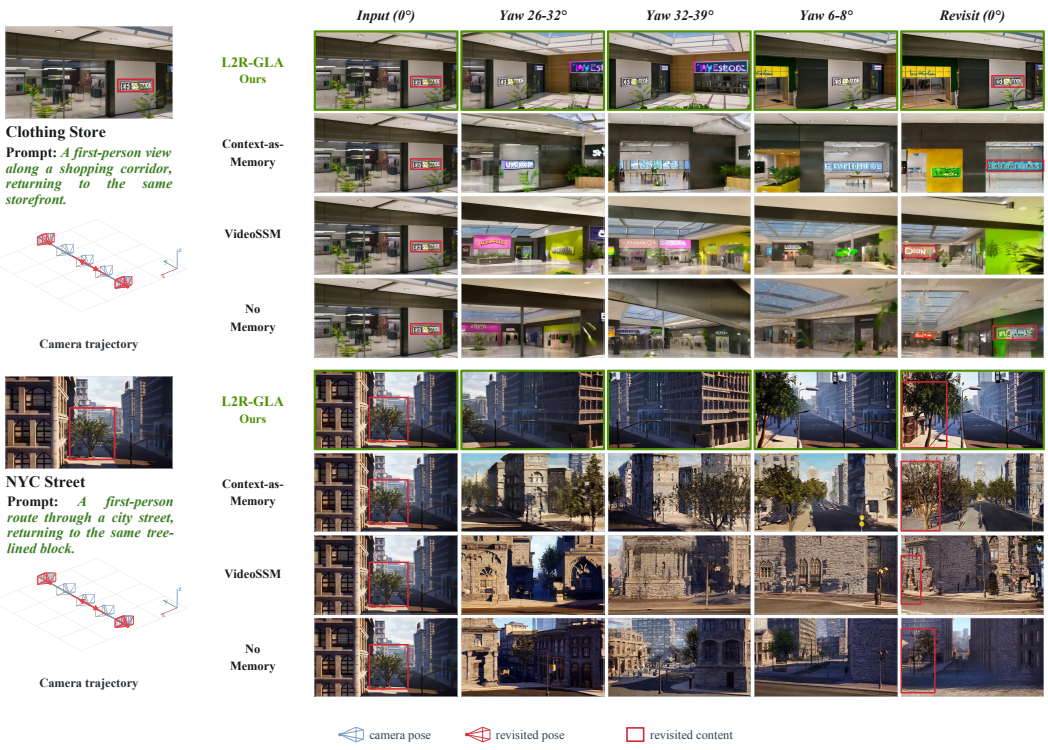


Figure 3: **Qualitative comparison on memory capacity.** Red boxes highlight the revisited regions. L2R-GLA has the best scene consistency.

The qualitative comparisons in Figure 3 further illustrate the memory behavior of L2R-GLA. After the camera moves away from a region and later returns to a matched viewpoint, L2R-GLA better preserves the appearance of the revisited content while maintaining the surrounding scene layout. The red boxes highlight these regions, showing that internal retrieval helps recover previously observed details with improved scene consistency.

4.3 Results on Video Quality

Table 2 demonstrates that L2R improves revisit consistency while maintaining strong video quality and camera controllability across different underlying video generation models. With Wan2.2 as the base model, both L2R variants achieve the highest or second-highest scores across all six visual-quality dimensions, with L2R-GLA ranking first in all of them. Meanwhile, L2R substantially

Table 2: **Quantitative results of video quality, camera controllability, and re-visit consistency** on RealEstate10K and DL3DV samples. Best in **bold**, second best underlined. * marks methods we retrained on our dataset because checkpoints weren't released; others use official checkpoints.

Method	Visual quality (↑)						Camera control (↓)		Consistency (↑)	
	Subject Consist	Bg Consist	Motion Smooth	Temporal Flicker	Aesthetic Quality	Imaging Quality	R_{err}	T_{err}	PSNR	SSIM
ViewCrafter	78.51	88.30	95.12	92.43	46.28	63.42	8.86	0.482	13.81	0.5217
SEVA	90.53	93.79	96.68	93.41	46.01	64.08	**5.26**	0.249	16.62	0.5427
VMem	87.71	92.50	94.16	90.41	42.05	61.50	10.31	0.518	16.85	0.5505
SANA-WM	91.73	93.67	96.81	93.99	46.58	66.47	5.40	0.383	15.52	0.5076
AnchorWeave-Wan2.2*	88.70	92.35	95.94	92.71	47.50	65.24	6.81	0.299	16.55	0.5472
CaR-Wan2.2	91.47	93.28	97.56	94.37	49.20	66.34	12.98	0.354	16.84	0.5456
L2R-GLA (Wan2.2)	**92.67**	**94.41**	**97.68**	**95.53**	**50.14**	**67.18**	6.29	0.300	16.79	0.5480
L2R-GDN (Wan2.2)	92.46	94.38	97.62	95.24	49.60	65.97	6.28	0.282	17.03	0.5546
L2R-GLA (SANA-WM)	91.36	94.17	97.02	93.84	46.81	66.07	5.61	0.259	16.97	0.5556
L2R-GDN (SANA-WM)	90.85	93.86	96.68	93.43	46.16	65.13	5.40	**0.246**	**17.29**	**0.5597**

improves camera controllability over prior memory-based methods. Compared with CaR, L2R-GLA reduces the rotation and translation errors from 12.98 to 6.29 and from 0.354 to 0.300, respectively, and L2R-GDN achieves 6.28 and 0.282. For revisit consistency, L2R-GDN attains the best PSNR of 17.03, improving over CaR by 0.19 dB, while L2R-GLA remains competitive at 16.79. These results indicate that the proposed internal retrieval mechanism improves scene consistency without requiring a trade-off with video quality or camera control.

Using SANA-WM as the base model, the same trend is observed. L2R-GDN achieves substantial gains in revisit consistency, improving PSNR from 15.52 to 17.29 and SSIM from 0.5076 to 0.5597, while also reducing the translation error from 0.383 to 0.246. Although its visual-quality scores are slightly lower than the SANA-WM baseline, the gap remains small across the six dimensions. L2R-GLA similarly improves revisit consistency to 16.97 PSNR and 0.5556 SSIM while maintaining competitive visual quality and camera control. Overall, these results show that L2R transfers across different underlying state-transition mechanisms and consistently strengthens long-term scene consistency without substantially degrading generation quality or camera controllability.

4.4 Ablation

To fully understand how each design element in L2R-GLA affects its memory capability, we conducted ablation studies on Echo-Memory.

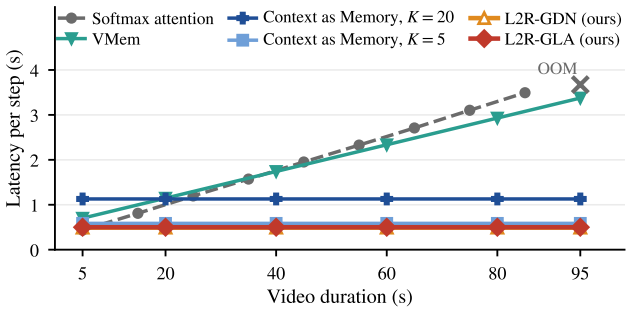
Table 3: **Ablation on the memory capacity.** GLA (baseline) is our starting point. The last three rows progressively add camera-guided retrieval γ_t, retrieval trigger r_t and re-visibility supervision (Eq. 5). Best in **bold**, second best underlined.

Variant	Replay			Loop closure			O-V ↑
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	
No Memory	10.03	0.398	0.534	10.32	0.291	0.643	12.25
VideoSSM	12.69	0.344	0.581	12.23	0.298	**0.584**	34.75
Context as Memory, K=5	11.92	0.408	0.501	10.72	0.307	0.596	50.75
GLA	10.55	0.199	0.763	12.74	0.273	0.584	66.13
+ Camera-guided retrieval (γ_t)	13.88	0.421	0.560	12.76	**0.399**	0.687	68.74
+ Retrieval trigger (r_t)	14.02	0.402	0.483	13.11	0.383	0.601	71.42
+ Re-visibility supervision (Eq. 5)	**14.74**	**0.513**	**0.430**	**13.14**	0.375	0.603	**72.57**

Effects of the retrieval gate γ_t and the retrieval trigger r_t. Table 3 separates the two mechanisms and highlights their distinct contributions. Starting from the GLA baseline, introducing camera-guided retrieval with γ_t yields the largest improvement in replay quality, with PSNR increasing from 10.55 to 13.88, SSIM from 0.199 to 0.421, and LPIPS decreasing from 0.763 to 0.560. This demonstrates the importance of using the camera pose to selectively access relevant historical information. Adding the retrieval trigger r_t further improves the results, increasing Replay PSNR from

Method	Layer ratio	Replay PSNR ↑	Loop closure PSNR ↑
I2V baseline	-	10.03	10.32
L2R-GDN	1:1	13.62	12.98
	3:1	13.88	12.76
	7:1	8.95	8.75
L2R-GLA	1:1	14.26	13.55
	3:1	14.74	13.14
	7:1	10.52	10.37

(a) Layer-ratio ablation



(b) Latency per denoising step

Figure 4: **Layer-ratio ablation and inference efficiency.** Left: L2R-GDN and L2R-GLA with layer ratios of 1:1, 3:1, and 7:1 on Echo-Memory benchmark, evaluated by replay and loop-closure PSNR. Right: latency comparison on one H100, 5 s chunks.

13.88 to 14.02 and reducing LPIPS from 0.560 to 0.483, while also improving Loop-closure PSNR from 12.76 to 13.11 and LPIPS from 0.687 to 0.601. The overall O-V score likewise increases from 68.74 to 71.42. These results indicate that camera-conditioned retrieval provides the main capacity gain, while the retrieval trigger further improves retrieval selectivity by activating memory access when historical information is needed.

Effects of the re-visibility supervision (Eq. 5). Re-visibility supervision further strengthens scene consistency by sharpening the trigger on revisits (Table 3). On replay, it improves PSNR by 0.72 dB, increases SSIM by 0.111, and reduces LPIPS from 0.483 to 0.430. Importantly, the auxiliary target is not merely an across-the-board image-metric regularizer. Its primary effect is to make triggering more selective and useful for revisited content, which directly translates into stronger memory.

Layer ratios. We further vary the ratio of L2R-GLA / L2R-GDN layers to standard softmax attention (Table 4 (a)). Both attention variants outperform the I2V baseline in replay and loop-closure PSNR, but the ratio matters: 3:1 maximizes replay PSNR, 1:1 maximizes loop-closure PSNR, and 7:1 degrades both. We therefore adopt 3:1 L2R-GLA in the main comparison to prioritize replay strength while keeping loop closure competitive. We attribute this trend to the fixed-capacity bottleneck of the linear-attention state: as trajectories grow longer and more diverse, the recurrent summary must overwrite earlier details, so stacking too many L2R-GLA layers amplifies information loss and can hurt both replay and loop closure. Retaining a fraction of softmax-attention layers acts as a high-bandwidth fallback that re-injects global context and stabilizes long-horizon generation, preventing this form of memory collapse. We discuss this further in Appendix D.

4.5 Efficiency

Figure 4(b) shows the per-step denoising cost on one H100 as rollout length increases, with resolution, camera conditioning, and chunk size matched. Full-history softmax attention rises from 0.43 s at 5 s to 3.67 s at 90 s and runs out of 80 GiB at 95 s. Both L2R layers are duration-invariant: 0.50 s for L2R-GLA and 0.49 s for L2R-GDN. External retrieval also decouples cost from duration but adds overhead: Context as Memory takes 0.59 s at $K{=}5$ and 1.13 s at $K{=}20$. Geometric retrieval pays to maintain memory: VMem rebuilds its surfel index each step by running point-map and global alignment over all stored frames, increasing from 0.70 s to 3.38 s between 5 s and 95 s chunks, near the softmax-attention curve.

5 Conclusion

This study establishes a useful analogy between external context memory retrieval mechanisms employed in scene-consistent video world models and the state-update dynamics. Leveraging this relationship, we introduce L2R, which designs camera-guided retrieval, retrieval triggering, and re-visibility supervision through a 3D geometry-based target. The resulting module can be integrated into pretrained camera-controllable video world models. Empirical results indicate that L2R substantially improves long-range scene consistency in video world models, eliminating the need for external memory retrievers or explicit 3D condition injection.

Ethics Statement

This paper presents research aimed at advancing the field of video world models. Our work may have multifaceted impacts on society. Given the rapidly growing capabilities of video generation technology, we urge against its misuse, such as the creation of misleading videos or video generation for malicious purposes.

Reproducibility Statement

We state that L2R is highly reproducible. Implementation and datasets details on our main experiences are provided in Appendix A. It is anticipated that these descriptions can sufficiently demonstrate the reproducibility of L2R. We plan to open-source the code and weight files after the paper passes peer review.

References

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.

JiaKui Hu, Jialun Liu, Liying Yang, Xinliang Zhang, Kaiwen Li, Shuang Zeng, Yuanwei Li, Haibin Huang, Chi Zhang, and Yanye Lu. Geometry-as-context: Modulating explicit 3d in scene-consistent video generation to geometry context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4258–4268, 2026.

Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson Lau, Wangmeng Zuo, et al. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *ACM Transactions on Graphics (TOG)*, 44 (6):1–15, 2025.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21807–21818. IEEE, 2024.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning (ICML)*, 2020.

Wayne King, Zeyue Xue, Yuxuan Bian, Jie Huang, Haoran Li, Yaowei Li, Yaofeng Su, Yuming Li, Haoyu Wang, Shiyi Zhang, Songchun Zhang, Yuwei Niu, Sihan Xu, Junhao Zhuang, Haoyang Huang, and Nan Duan. Echo-memory: A controlled study of memory in action world models. *arXiv preprint arXiv:2606.09803*, 2026.

Kunyang Li, Mubarak Shah, and Yuzhang Shang. Attend locally, remember linearly: Linear attention as cross-frame memory for autoregressive video diffusion. *arXiv preprint arXiv:2605.16579*, 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.

Yuqi Pan, Yongqi An, Zheng Li, Yuhong Chou, Ruijie Zhu, Xiaohui Wang, Mingxuan Wang, Jinqiao Wang, and Guoqi Li. Scaling linear attention with sparse state expansion. *arXiv preprint arXiv:2507.16577*, 2025.

Zhan Peng, Jie Ma, Huiqiang Sun, Chong Gao, Zhijie Xue, Zhiyu Pan, Zhiguo Cao, Jun Liang, and Jing Li. Compression and retrieval: Implicit memory retrieval for video world models. *arXiv preprint arXiv:2606.23105*, 2026.

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.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. pi3: Permutation-equivariant visual geometry learning. In *International Conference on Learning Representations*, volume 2026, pp. 10481–10497, 2026a.

Zun Wang, Han Lin, Jaehong Yoon, Jaemin Cho, Yue Zhang, and Mohit Bansal. Anchorweave: World-consistent video generation with retrieved local spatial memories. In *European Conference on Computer Vision*, pp. 203–224. Springer, 2026b.

Zhengxuan Wei, Xu Guo, Xinghui Li, Xunzhi Xiang, Min Wei, Yiran Zhu, Qiulin Wang, Xintao Wang, Pengfei Wan, Xiangwang Hou, et al. Geometry-aware implicit memory for video world models. *arXiv preprint arXiv:2606.02436*, 2026.

Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video world models with long-term spatial memory. *Advances in Neural Information Processing Systems*, 38:49371–49393, 2026.

Zeqi Xiao, LAN Yushi, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. Worldmem: Long-term consistent world simulation with memory. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Jiacong Xu, Hanwen Jiang, Zhixin Shu, Kalyan Sunkavalli, Vishal M Patel, and Yiqun Mei. Wonder: Video world model done better. *arXiv preprint arXiv:2607.26037*, 2026a.

Tian-Xing Xu, Zi-Xuan Wang, Guangyuan Wang, Li Hu, Zhongyi Zhang, Peng Zhang, Bang Zhang, and Song-Hai Zhang. Ucm: Unified modeling of camera control and memory with time-aware positional encoding warping for world models. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–12, 2026b.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, et al. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025a.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. In *International Conference on Machine Learning (ICML)*, 2024.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. In *International Conference on Learning Representations (ICLR)*, 2025b.

Yuxue Yang, Lue Fan, Ziqi Shi, Junran Peng, Feng Wang, and Zhaoxiang Zhang. Neoverse: Enhancing 4d world model with in-the-wild monocular videos. *arXiv preprint arXiv:2601.00393*, 2026.

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*, pp. 40256–40265, 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. *arXiv preprint arXiv:2512.05081*, 2025.

Jiwen Yu, Jianhong Qin, Haoran Chen, Xintao Zhang, Ruixiang Yang, Pengfei Wan, Di Zhang, and Xihui Liu. Context as memory: Scene-consistent interactive long video generation with memory retrieval. *arXiv preprint arXiv:2506.03141*, 2025a.

Jiwen Yu, Jianxiong Gao, Jianhong Bai, Yiran Qin, Kaiyi Huang, Quande Liu, Xintao Wang, Pengfei Wan, Kun Gai, and Xihui Liu. Memlearner: Learning to query context memory for video world models. *arXiv preprint arXiv:2606.31734*, 2026a.

Mark Yu, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 100–111, 2025b.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025c.

Wei Yu, Runjia Qian, Yumeng Li, Liquan Wang, Songheng Yin, Dennis Anthony, Yang Ye, Yidi Li, Weiwei Wan, Animesh Garg, et al. Mosaicmem: Hybrid spatial memory for controllable video world models. *arXiv preprint arXiv:2603.17117*, 2026b.

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*, 2025d.

Lvmin Zhang and Maneesh Agrawala. Packing input frame context in next-frame prediction models for video generation. *arXiv e-prints*, pp. arXiv–2504, 2025.

Min Zhao, Guande He, Yixiao Chen, Hongzhou Zhu, Chongxuan Li, and Jun Zhu. Riflex: A free lunch for length extrapolation in video diffusion transformers. *arXiv preprint arXiv:2502.15894*, 2025.

Guangcong Zheng, Teng Li, Xianpan Zhou, and Xi Li. Realcam-vid: High-resolution video dataset with dynamic scenes and metric-scale camera movements. *arXiv preprint arXiv:2504.08212*, 2025.

Sixiao Zheng, Minghao Yin, Wenbo Hu, Xiaoyu Li, Ying Shan, and Yanwei Fu. Versecrafter: Dynamic realistic video world model with 4d geometric control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40277–40290, 2026.

Haoyi Zhu, Haozhe Liu, Yuyang Zhao, Tian Ye, Junsong Chen, Jincheng Yu, Tong He, Song Han, and Enze Xie. Sana-wm: Efficient minute-scale world modeling with hybrid linear diffusion transformer. *arXiv preprint arXiv:2605.15178*, 2026.

A Implementation details

A.1 More efficient analysis

Parameters. The L2R layer reuses the query, key, value, and output projections of the self-attention it replaces, so the new parameters are projection layers to obtain the retrieval operator γ_t and the retrieval trigger r_t. When using Wan2.1 as backbone, one softmax self-attention holds $9.45\,\mathrm{M}$ parameters and L2R-GLA holds $11.20\,\mathrm{M}$, of which $9.45\,\mathrm{M}$ are inherited. Replacing the first three blocks of four blocks therefore adds $5.25\,\mathrm{M}$ parameters.

FLOPs. For one $5\,\mathrm{s}$ chunk of Wan2.1 (N=18,480, D=1536), a softmax self-attention layer costs $2.45\,\mathrm{TFLOPs}$ per forward pass, whereas L2R-GLA costs $0.49\,\mathrm{TFLOPs}$, about $5\times$ fewer. L2R-GDN has the same linear-in-N complexity, with its additional delta correction introducing only a constant-factor overhead over L2R-GLA. Replacing the first three blocks of four blocks therefore reduces the total cost of the 30-block DiT from 111 to $95\,\mathrm{TFLOPs}$ per forward pass (14%). Unlike softmax attention, whose cost grows quadratically with sequence length, both L2R-GLA and L2R-GDN scale linearly with the context length, making the computational advantage more pronounced for longer contexts.

A.2 Re-visit benchmark and evaluation on RealEstate10K and DL3DV

We construct **Revisit-Bench**, a controlled benchmark for evaluating memory and loop-closure consistency in video world models. The benchmark is constructed from real camera trajectories and video frames from RealEstate10K and DL3DV-10K. All benchmark generation is deterministic with a fixed random seed of 42.

A.2.1 Overview

A video world model should preserve sufficiently long-range scene information to maintain consistency when the camera moves away from an observed region and subsequently revisits it. In particular, when the camera returns to its initial viewpoint, the generated frame should be consistent with the initial observation. Revisit-Bench quantifies this capability through a controlled trajectory construction protocol.

Each benchmark sequence contains 81 frames, consisting of one reference frame and 80 generated target frames. The camera trajectory is derived from real relative camera motion and transformed into a temporal palindrome, such that the camera exactly returns to its initial viewpoint at a predefined revisit time. Under the static-scene assumption, every target viewpoint has a corresponding ground-truth frame from the original video, allowing direct pixel-level evaluation.

We construct two trajectory splits using the same set of 300 scenes, with different revisit patterns:

- `palindrome_far`: the folding point is 40, and the camera revisits the initial viewpoint at $t = 80$. The camera traverses the entire trajectory before returning, resulting in the largest memory span.
- `palindrome_near`: the folding point is 24, and the camera revisits the initial viewpoint at $t = 48$. After revisiting, the camera continues moving in the original forward direction for 32 additional steps, preventing degenerate solutions in which the model simply remains at the revisited viewpoint.

Each split contains 300 scenes, including 150 scenes from RealEstate10K and 150 from DL3DV-10K. Therefore, each evaluated method produces 600 sequences in total across the two splits.

A.2.2 Data Sources

The benchmark is constructed from RealCam-Vid (Zheng et al., 2025). We use video clips from RealEstate10K and DL3DV-10K.

Each entry in `RealCam-Vid_train.npz` contains the following fields:

- `dataset_source`: either `RealEstate10K` or `DL3DV-10K`;

- `video_path`: the video path relative to `DATA_ROOT`;
- `camera_extrinsics`: a sequence of world-to-camera (w2c) extrinsic matrices of shape $(F, 4, 4)$;
- `camera_intrinsics`: normalized camera intrinsics (f_x, f_y, c_x, c_y), where the focal lengths and principal-point coordinates are normalized by the source image width and height;
- `align_factor`: a per-window scale factor used to normalize the translation magnitude, consistent with the training data loader;
- `short_caption` and `long_caption`: textual descriptions of the scene.

The benchmark is sampled from these data while using different trajectory constructions. Thus, the benchmark is designed to evaluate revisit consistency.

A.2.3 FILTERING

We randomly permute the source records using a fixed seed of 42 and scan them sequentially. For each source dataset, we select 150 scenes subject to the following criteria:

1. **Video-level deduplication.** At most one window is selected from each source video.
2. **Valid camera poses.** All extrinsic matrices and relative poses must contain finite values, and the required matrix inversions must be numerically stable.
3. **Motion filtering.** Motion statistics are computed from the relative camera poses after applying the same alignment scale used during training. The median per-frame translation step must lie in $[0.02, 0.8]$, removing nearly static clips and extreme trajectories, and the maximum single-frame rotation must not exceed 10°.

A.2.4 CAMERA POSE PROCESSING

We process camera poses using the same coordinate transformation as the training data loader. Given the sampled source extrinsics $\mathbf{W}_t \in \mathbb{R}^{4\times4}$, which follow the OpenCV world-to-camera convention, we first convert them to camera-to-world coordinates and apply the per-window alignment factor:

$$\mathbf{C}_t = \mathbf{W}_t^{-1}, \qquad \mathbf{C}_t[:3, 3] \leftarrow \mathbf{C}_t[:3, 3] \cdot s, \tag{7}$$

where s denotes `align_factor`. We then convert the poses back to world-to-camera coordinates and express them relative to the first frame:

$$\mathbf{W}_t^{\mathrm{rel}} = \mathbf{W}_t \left(\mathbf{W}_0\right)^{-1}, \qquad \mathbf{C}_t^{\mathrm{rel}} = \left(\mathbf{W}_t^{\mathrm{rel}}\right)^{-1}. \tag{8}$$

Consequently, $\mathbf{C}_0^{\mathrm{rel}} = \mathbf{I}$.

The resulting poses use the OpenCV camera-to-world convention, where the camera coordinate axes are x-right, y-down, and z-forward. The median translation step is approximately 0.48 for the training clips and 0.17 for the benchmark sequences.

For intrinsics, we recover source-pixel camera intrinsics from the normalized intrinsics provided by RealCam-Vid. Given normalized parameters (f_x, f_y, c_x, c_y) and source resolution (w, h), the intrinsic matrix is

$$\mathbf{K} = \begin{bmatrix} f_x w & 0 & c_x w \\ 0 & f_y h & c_y h \\ 0 & 0 & 1 \end{bmatrix}. \tag{9}$$

We use $(w, h) = (1280, 740)$ for RealEstate10K and $(w, h) = (960, 580)$ for DL3DV-10K. The same intrinsic matrix is used for all 81 frames within each sequence and is stored as an $(81, 3, 3)$ array.

A.2.5 PALINDROME TRAJECTORY CONSTRUCTION

Let $t \in \{0, \ldots, 80\}$ denote the frame index, let f denote the folding point, and let $\mathbf{C}_t^{\text{rel}}$ denote the real relative camera poses obtained above. We construct the camera trajectory as

$$\mathbf{P}(t) = \begin{cases} \mathbf{C}^{\text{rel}}_{\min(t,\, 2f-t)}, & t \le 2f, \\ \mathbf{C}^{\text{rel}}_{t-2f}, & t > 2f \quad \text{(near split only)}. \end{cases} \tag{10}$$

The first part of the trajectory therefore mirrors the original real camera motion directly rather than interpolating poses or manually reversing motion directions. Every pose in the palindrome is consequently an actual viewpoint present in the source trajectory.

At $t = 2f$, the camera pose is exactly the identity:

$$\mathbf{P}(2f) = \mathbf{I}. \tag{11}$$

This property is explicitly verified during benchmark construction with an absolute tolerance of 10^{-9}.

For `palindrome_far`, $f = 40$, yielding the trajectory

$$0 \to 40 \to 80,$$

where the camera returns to the initial viewpoint at $t = 80$. There is no post-revisit segment.

For `palindrome_near`, $f = 24$, yielding

$$0 \to 24 \to 48,$$

followed by

$$t = 49, \ldots, 80 \quad \mapsto \quad \mathbf{C}_1^{\text{rel}}, \ldots, \mathbf{C}_{32}^{\text{rel}}.$$

Thus, after returning to the initial viewpoint at $t = 48$, the camera continues along the original forward direction for 32 additional steps. This post-revisit segment prevents a trivial solution in which the generated trajectory remains stationary after loop closure.

A.2.6 GROUND-TRUTH FRAMES AND STATIC-SCENE ASSUMPTION

We assume a static scene, following the standard setting for loop-closure evaluation. Under this assumption, the ground-truth image corresponding to each generated viewpoint is the source frame associated with that viewpoint. We therefore define

$$\text{gt_frame_indices}[t] = \begin{cases} \min(t,\, 2f - t), & t \le 2f, \\ -(t - 2f), & t > 2f \quad \text{(near split only)}. \end{cases} \tag{12}$$

The negative indices in the near split identify the post-revisit overshoot segment and indicate that the absolute value should be used to retrieve the corresponding source frame. In particular, frames $t = 0$ and $t = 2f$ correspond to exactly the same source viewpoint. Hence, the ground-truth frame at the revisit point is identical to the input reference image.

A.2.7 TEXT PROMPTS

The text prompt for each sequence is taken from the `short_caption` field of RealCam-Vid. If the short caption is unavailable, we fall back to `long_caption`; if both captions are missing, we use the fixed prompt "A slow camera motion through a real scene." The resulting prompts contain approximately 10–25 words, with an average length of 18.2 words.

A.2.8 STATISTICS

Each split contains 300 scenes, consisting of 150 RealEstate10K scenes and 150 DL3DV-10K scenes. Since both splits use the same set of source scenes, each evaluated method produces 600 generated sequences in total: 300 for the far split and 300 for the near split.

For the original trajectories before palindrome construction, the median per-frame translation step, measured in the aligned coordinate scale, has a mean of 0.408 and a median of 0.171, with values ranging from 0.040 to 0.783. The maximum per-frame translation step has a mean of 0.39, while the maximum single-frame rotation has a mean of 2.19°. The latter is bounded by the 10° filtering threshold. For comparison, the median translation step of the training clips is approximately 0.48.

A.2.9 Evaluation Protocol

We evaluate generated videos using six groups of metrics.

Revisit consistency. The primary metric measures the agreement between the generated revisit frame $t_r = 2f$ and the input reference image. We report PSNR and SSIM, with LPIPS optionally reported as an additional perceptual metric. For uint8 images, PSNR is computed as

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{255^2}{\mathrm{MSE}} \right). \tag{13}$$

SSIM is computed on grayscale images after averaging the three RGB channels, using $\mathrm{data_range} = 255$.

Camera accuracy. We additionally follow the VMem (Li et al., 2025) evaluation protocol for camera trajectory accuracy. Camera poses are estimated from the generated video using Pi3X and compared with the ground-truth camera poses after applying the same alignment procedure. Both trajectories are first expressed relative to their first frame:

$$\mathbf{P}'(t) = \mathbf{P}(0)^{-1} \mathbf{P}(t). \tag{14}$$

This removes the arbitrary global origin and orientation associated with the estimated trajectory.

Because the estimated camera trajectory contains an unknown global similarity scale, translation is normalized by the distance of the farthest frame:

$$\mathbf{t} \leftarrow \frac{\mathbf{t}}{\max_i \|\mathbf{t}_i\|_2}. \tag{15}$$

No Umeyama alignment is applied. We then report the mean translation error

$$d_t = \|\mathbf{t}_{\mathrm{gt}} - \mathbf{t}_{\mathrm{gen}}\|_2 \tag{16}$$

over $t = 1, \ldots, T-1$, together with the rotational geodesic error

$$d_R = \arccos \left(\frac{\mathrm{tr}\left(\mathbf{R}_{\mathrm{gen}} \mathbf{R}_{\mathrm{gt}}^{\mathsf{T}}\right) - 1}{2} \right), \tag{17}$$

reported in degrees.

VBench. We further report six VBench Huang et al. (2024) dimensions: Subject Consistency (DINO), Background Consistency (CLIP), Motion Smoothness (AMT-S), Temporal Flickering (frame-difference MAE), Aesthetic Quality (CLIP with a linear head), and Imaging Quality (MUSIQ). The first 81 frames of each generated video are used, and the prompt recorded in the corresponding manifest entry is provided as the text input.

A.3 Training Settings

Memory capacity. Following Echo-Memory (King et al., 2026), we train L2R-GLA and L2R-GDN on 8 A100 GPUs. The hyper-parameters are as follows.

Video quality. Although we use two base models, we consult the same training hyperparameters for them to train our own model.

B Extension to other state update rules

L2R defines the retrieval condition r_t and the state-access gate γ_t, while the underlying state-space mechanism determines how the retrieved state is updated. In implementation, the vector-valued retrieval gate γ_t is converted to a diagonal matrix.

$$S_t = \begin{cases} \gamma_t \, S_{t-1} + v_t k_t^{\top}, & \text{If } r_t = 1, \\ S_{t-1} + v_t k_t^{\top}, & \text{If } r_t = 0. \end{cases} \tag{18}$$

Setting	Value
Backbone	Wan2.1
Resolution	352×640
Chunk length T	81 frames
Optimizer	AdamW
Learning rate	5×10^{-5}
Per-device batch / grad accumulation	1 / 1
GPUs	8 (A100-80G)
Total steps	5 k
Timestep shift	15
Spike rejection threshold	15.0
Target-frame-only supervision	enabled
Flow noise shift	1.0
Relative-RT action encoding	enabled

Setting	Value
Backbone	Wan2.2 / SANA-WM
Resolution	704×1280
Chunk length T	81 frames
Optimizer	AdamW
Learning rate	8×10^{-5}
Per-device batch / grad accumulation	2 / 1
GPUs	64 (H20-96G)
Total steps	5 k
Timestep shift	9.98
Spike rejection threshold	1.0
Target-frame-only supervision	enabled
Flow noise shift	1.0
Relative-RT action encoding	enabled

For GDN, the same retrieval formulation is combined with its delta correction:

$$S_t = \begin{cases} \gamma_t \left(S_{t-1} - \beta_t S_{t-1} k_t k_t^\top \right) + \beta_t v_t k_t^\top, & \text{If } r_t = 1, \\ S_{t-1} + \beta_t v_t k_t^\top, & \text{If } r_t = 0. \end{cases} \tag{19}$$

Here, γ_t controls which historical information is accessed, while β_t controls the subtractive delta update and corresponding write strength in GDN.

C More qualitative results

Figure 5 shows four further Echo-Memory loop-closure cases (the camera turns away for two chunks and turns back). Each row shows the input frame, three frames along the trajectory and the frame at the revisited pose; red boxes mark the revisited content, and the left panel gives the scene, the prompt and the camera trajectory.

D Failure mode

Figure 6 shows loop-closure rollouts on the same scenes for the two retrieval baselines and for L2R-GLA with a 7:1 memory-layer ratio. With only four memory layers, the recurrent state has to summarize the whole trajectory at few depths of the network, and the softmax layers that remain no longer provide a high-bandwidth fallback for the content the state has overwritten: the return frames lose the observed structure and the loop is not closed, in line with the drop of the 7:1 rows in Figure 4(a), whereas the 1:1 and 3:1 ratios close the loop (Section 4.4).

E Future work

Future research could focus on the issue of memory retrieval in dynamic scenes. Optimizing retrieval triggers for dynamic scenes using more accurate geometric or external supervision represents

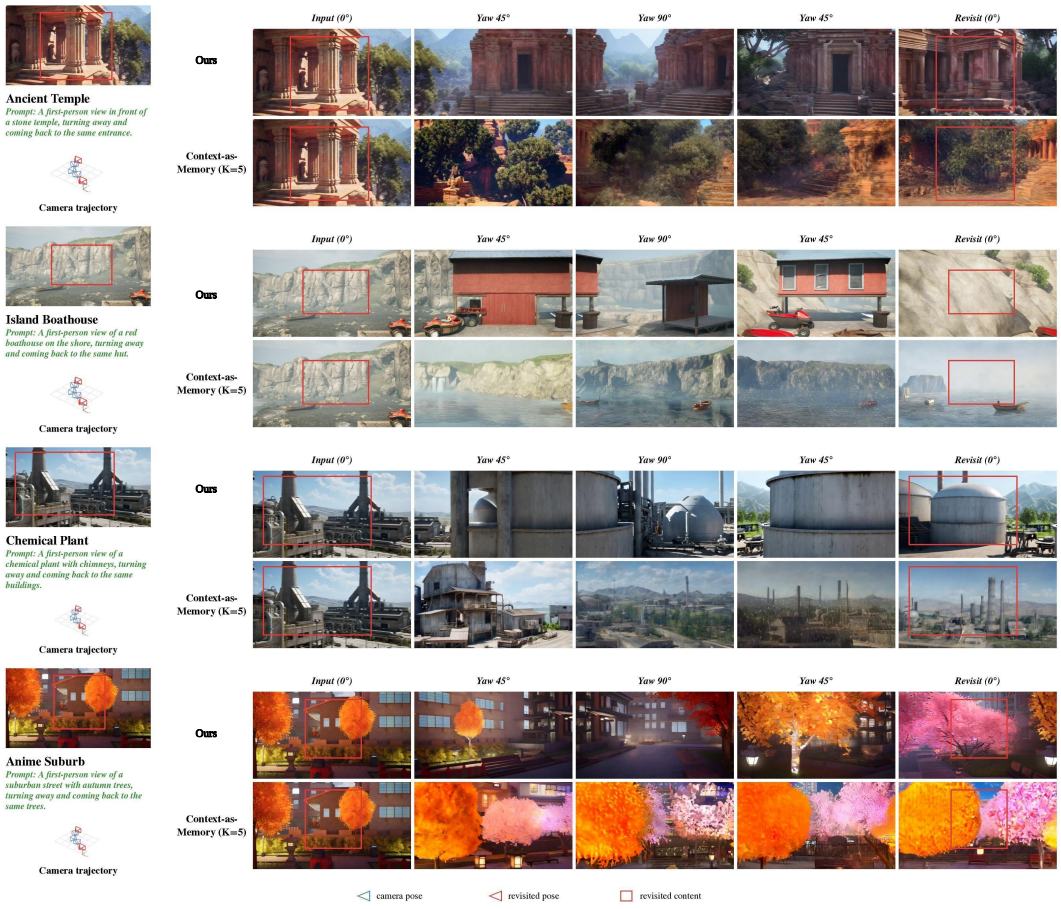


Figure 5: Loop closure on four further scenes. Top row of each case: Ours; bottom row: Context as Memory with $K{=}5$ retrained on the same data.

a viable path forward. We encourage the community to build upon this work to explore more comprehensive internal memory systems suitable for video world models.

F AI ASSISTANCE

We used AI to polish the grammar of this paper. AI was not used to assist with the core ideas or the experimental design.

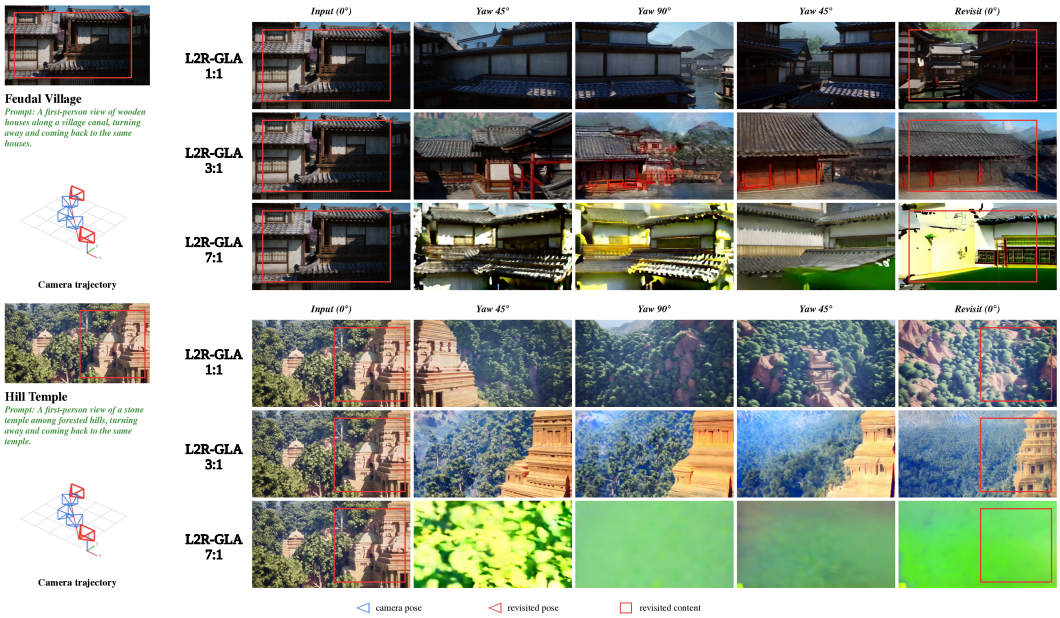


Figure 6: Failure mode at a high memory-layer ratio. Loop closure on two scenes with memory-layer ratios of 3:1, 1:1 and 7:1. With too few memory layers the rollout loses the observed structure on the return.