WORLDKV: EFFICIENT WORLD MEMORY WITH WORLD RETRIEVAL AND COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive video diffusion models have enabled real-time, action-conditioned world generation. However, sustaining a persistent world, where revisiting a previously seen viewpoint yields consistent content, remains an open problem. Full KV cache attention may preserve this consistency but breaks real-time constraints: memory footprint and attention cost grow linearly with rollout length. Sliding window inference restores throughput but discards long-term memory. We observe that, even without memory-specific training, attention over historical KV caches follows viewpoint correspondence, concentrating on chunks that overlap with the current view. Motivated by this observation, we propose WorldKV, a training-free framework for persistent world modeling with two components: World Retrieval selectively retrieves scene-relevant chunks via camera/action correspondence, inserting them back into the native attention window without re-encoding. World Compression prunes redundant tokens within each chunk via key-key similarity to an anchor frame, halving per-chunk storage to fit $2\times$ more history under a fixed budget. On Matrix-Game-2.0 and LingBot-World-Fast, WorldKV matches or exceeds the fidelity achieved with Full KV memory, at roughly $2\times$ the throughput.

1 INTRODUCTION

Autoregressive video diffusion models with causal attention and KV caching have recently emerged as a promising architecture for real-time interactive world generation (Team et al., 2026b; He et al., 2025; Seo et al., 2026; Ye et al., 2025; Savva et al., 2026; Gao et al., 2026). These models generate camera- or action-conditioned visual streams at real-time frame rates, enabling applications in gaming (Sun et al., 2025), embodied AI agents (Team et al., 2026a), and robotic simulation (Gao et al., 2026).

Beyond producing plausible frames, the emerging goal is to sustain a persistent, explorable world in which a user can navigate freely, leave a room, and return to find it unchanged. Achieving this kind of persistence is tightly linked to *spatial and temporal memory*: the ability to retain and recall scene content across time and revisits. Yet despite rapid progress in model architectures, consistent memory remains an open challenge. A consistent world model should reconstruct the same structures and appearances when revisiting previously explored areas.

A recent observation is that the KV cache in these models is not merely a computational buffer but already functions as an emergent form of *world memory*. LingBot-World (Team et al., 2026b) demonstrated that, even without explicit memory training, attending to the full history of KV caches enables the model to maintain spatial and temporal consistency across revisits. However, LingBot-World (Team et al., 2026b) was trained on minute-level videos, so its long-term memory may reflect learned behavior rather than a property of the KV cache itself. In this paper, we show that the phenomenon is more fundamental: it appears even in models not trained on long sequences. On Matrix-Game-2.0 (He et al., 2025), which was trained on short sequences with a 6-frame sliding window, the model can nonetheless leverage past KV caches as long-term visual memory at inference time (Fig. 1). When we remove the sliding-window restriction and let the model attend to its entire KV cache history, the model successfully reproduces previously seen viewpoints, while the same model under sliding-window inference fails. The memory is already there; the question is how to access it without the full cost of attending to the entire KV cache.

Indeed, the cost of leveraging this emergent memory through full-history KV cache attention is substantial in practice: each frame produces 880 (He et al., 2025) to 1,560 (Team et al., 2026b) tokens,

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.

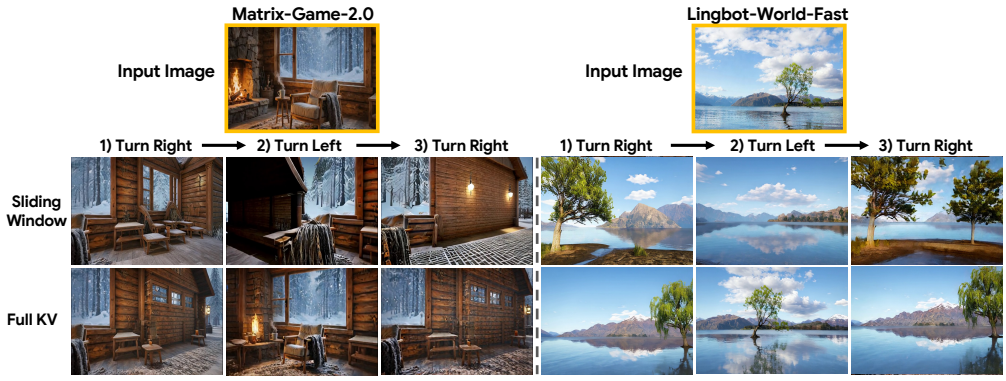


Figure 1: **Emergent memory from attending Full KV cache**. We empirically observed that, even though Matrix-Game-2.0 (He et al., 2025) was trained only on short clips, the model can use the KV cache as long-term visual context/memory.

accumulating hundreds of thousands of tokens over a one-minute rollout. The corresponding KV cache rapidly exceeds GPU VRAM capacity (Fig. 2 (a)). Even before out-of-memory failures, the rapidly growing attention cost degrades inference speed: on LingBot-World-Fast, FPS drops from 8.87 to 3.61 over a one-minute rollout (Fig. 2 (b)), breaking real-time constraints. Sliding-window inference is therefore a structural necessity for real-time generation, yet it comes with an inherent trade-off: the eviction that bounds attention cost is also what discards long-term memory. In addition, models operating under sliding-window inference tend to hallucinate new content or drift (He et al., 2025), as the KV caches from the original scene have long been evicted from the context.

Recent works address this through memory-augmented architectures: external memory banks retrieved via cross-attention (Xiao et al., 2025; Huang et al., 2025a), spatial compression of the entire history (Hong et al., 2025), or explicit 3D scene representations (Ren et al., 2025; Li et al., 2025b) that condition the video model on rendered views of the reconstructed geometry. While effective, these approaches require training dedicated memory modules or fine-tuning the backbone, and the 3D-representation methods additionally incur reconstruction latency at inference time.

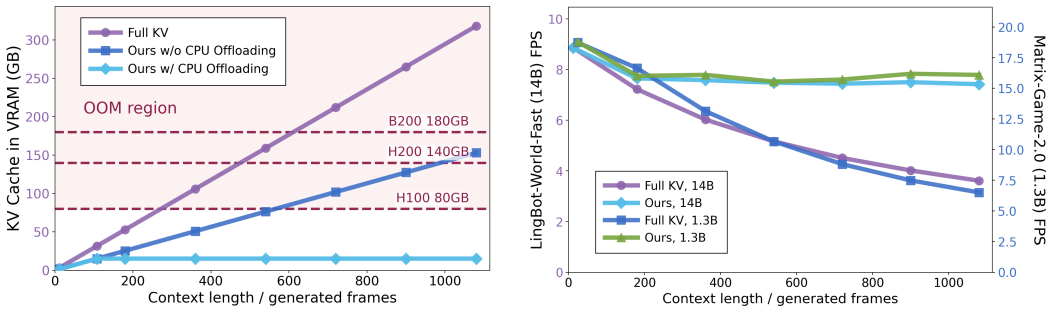
We take a different perspective. Rather than building external memory on top of the model, we analyze attention over its KV cache and find a view-correspondent pattern: historical chunks whose viewpoints overlap with the current view receive high attention (Fig. 3). This observation motivates selectively retrieving view-relevant KV chunks without attending to the entire history. We introduce **WorldKV**, a training-free framework that enables efficient long-term memory in autoregressive video world models through two complementary components: **World Retrieval** and **World Compression**.

World Retrieval preserves KV cache chunks by storing them in GPU/CPU memory and selectively retrieving scene-relevant caches back into the active attention window when the model revisits a scene. Retrieved KV caches are inserted back into the context natively, with no re-encoding or architectural changes required. The retrieval mechanism is modular, supporting camera/action-based and attention-based strategies as interchangeable components (Appendix C).

World Compression reduces redundancy in adjacent frames, which produce near-duplicate KV caches. By pruning redundant tokens based on Key-Key similarity, each chunk is roughly halved in size, allowing twice as much history under the same memory budget.

Our contributions are as follows:

- We show that autoregressive world models trained without memory-specific objectives already exhibit view-correspondent attention over their KV caches, and build on this emergent memory with **World Retrieval**, a training-free, retrieval-algorithm-agnostic framework that selectively retrieves view-relevant KV caches.
- We present **World Compression**, a Key-Key similarity-based pruning that compresses each chunk to approximately half of its original size, enabling $2\times$ more retrieval coverage attainable within a fixed attention budget and yielding consistently improving revisit fidelity.
- We quantitatively demonstrate, on two autoregressive video world models of different scales (Matrix-Game-2.0 (He et al., 2025), LingBot-World-Fast (Team et al., 2026b)), that training-



(a) KV cache size (LingBot-World-Fast 14B) (b) FPS over rollout length

Figure 2: **Cost of full-history KV cache attention.** (a) Full KV grows rapidly into the OOM region, while WorldKV grows gradually via compression and stays nearly flat with CPU offloading. (b) Full KV throughput degrades continuously on both backbones (14B on 4×B200, 1.3B on 4×H200), while WorldKV maintains stable throughput.

free KV cache management matches or exceeds Full KV cache attention on revisit fidelity while maintaining real-time inference.

2 Related Work

Autoregressive Video Diffusion. Recent work (Chen et al., 2024; Teng et al., 2025; Zhu et al., 2026; Chen et al., 2025) integrates diffusion modeling with autoregressive (AR) prediction for long-horizon and streaming video generation. CausVid (Yin et al., 2025) distills a bidirectional diffusion transformer into a causal AR generator. Self Forcing (Huang et al., 2025b) mitigates mismatch between training and inference by training on self-generated rollouts with KV caching. Rolling Forcing (Liu et al., 2025) jointly denoises multiple frames at progressively increasing noise levels. LongLive (Yang et al., 2025) introduces KV re-caching for smooth prompt transitions. Building on this line of work, real-time interactive video world models leveraging KV caching have emerged as a natural extension, exploiting cached past states for low-latency generation under streaming user input.

Interactive World Model. Building on autoregressive video diffusion, interactive world models predict action-conditioned future frames. Matrix-Game-2.0 (He et al., 2025) injects keyboard and mouse signals, while Hunyuan-GameCraft (Li et al., 2025a) unifies them into a camera action space. LingBot-World (Team et al., 2026b) scales interactive world generation toward diverse domains and long-horizon rollouts. A growing line of work has explored memory mechanisms for long-term consistency in world models. Context-as-Memory (Yu et al., 2025) stores past frames as memory and re-encodes FOV-overlapping frames as conditioning context, with retrieval-aware training. WorldPlay (Sun et al., 2025) rebuilds context from geometrically important past frames via KV cache recomputation, with memory-aware distillation. RELIC (Hong et al., 2025) introduces a learnable action-aware compression mechanism that stores historical latent memory in the KV cache. In contrast, our framework operates training-free, exploiting view relevance and redundancy in the existing KV cache. Comparisons with Context-as-Memory and WorldPlay is provided in Appendix E.

KV Cache Management. In autoregressive generation, the KV cache grows linearly with sequence length, creating a bottleneck for long-context inference. In LLMs, fixed-budget cache management has been studied through positional heuristics (Xiao et al., 2023), accumulated attention scores (Zhang et al., 2023), observation-window importance estimates (Ghadia et al., 2025), and query-aware page retrieval (Tang et al., 2024). While these methods reduce the cost of language-model decoding, they are not designed for dense spatiotemporal generation. Recent work has begun to explore training-free KV-cache management for long-horizon autoregressive video diffusion (Yi et al., 2025). We extend this direction to interactive world models, where long-horizon consistency further requires retrieving scene-relevant memory across revisited viewpoints while compressing redundant visual KV caches.

3 Preliminaries

Interactive World Models. An interactive world model aims to predict future visual observations from actions. Given the current visual state $s_t \in \mathcal{S}$ and an action $a_t \in \mathcal{A}$, the model defines a

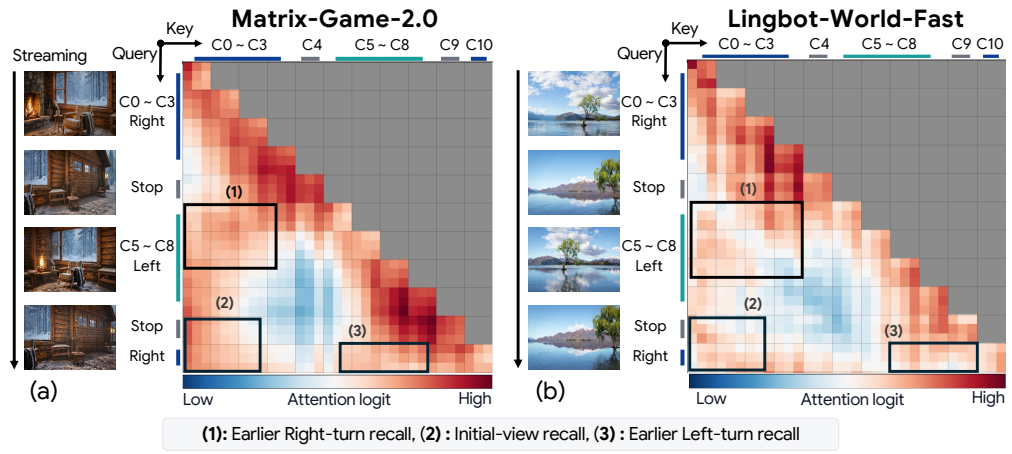


Figure 3: **Attention maps for the action sequence.** "Right (chunks C0–C3) → Stop (C4) → Left (C5–C8) → Stop (C9) → Right (C10)". The model assigns high attention to KV caches whose viewpoint overlaps with the current action. This pattern motivates using camera/action as the retrieval criterion for selecting relevant past KV caches.

conditional distribution over the next state s_{t+1}:

$$s_{t+1} \sim p(s_{t+1} \mid s_t, a_t), \tag{1}$$

where $p : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition distribution. In this work, "world model" refers specifically to this action-conditioned visual generation setting, where actions are discrete keyboard inputs or continuous camera trajectories. Recent world models (He et al., 2025; Team et al., 2026b; Savva et al., 2026; Team et al., 2026a) implement this transition using autoregressive video diffusion built on causal DiT architectures.

Autoregressive Video Diffusion with KV Cache. Autoregressive video diffusion models (Huang et al., 2025b; Teng et al., 2025; Chen et al., 2025) synthesize long videos by sequentially generating frames or chunks (e.g., 3 frames). For a video of N frames $x^{1:N} = (x^1, x^2, \ldots, x^N)$, the generation is factorized as:

$$p(x^{1:N}) = \prod_{i=1}^{N} p(x^i \mid x^{<i}), \tag{2}$$

where each conditional is modeled by a diffusion process. In practice, recent causal diffusion transformers (Huang et al., 2025b; Yang et al., 2025; Liu et al., 2025) implement this conditioning through a KV cache, which stores key-value projections of previously generated frames or chunks. At step t, the transformer $\mathcal{G}_\theta$ denoises a noisy latent $x_t^{(\sigma)}$ conditioned on prior cached entries:

$$\hat{x}_t = \mathcal{G}_\theta(x_t^{(\sigma)},\ \sigma,\ \mathbf{K}_{<t},\ \mathbf{V}_{<t}). \tag{3}$$

The new key-value pairs are appended to the cache for subsequent steps.

4 Method

4.1 Overview

Our framework, **WorldKV**, operates on top of sliding-window inference and introduces two complementary components addressing the two bottlenecks of Full KV inference: attention computation and storage (Fig. 2 (a), (b)). **World Retrieval** (Sec. 4.2) stores evicted KV cache chunks in GPU/CPU memory and retrieves only viewpoint-relevant caches at revisit time, bounding the active attention window to preserve real-time inference speed. **World Compression** (Sec. 4.3) prunes redundant tokens within each chunk via key-key similarity, compressing each 3-frame chunk to approximately half its size and fitting roughly 2× chunks under a fixed memory budget without out-of-memory failures.

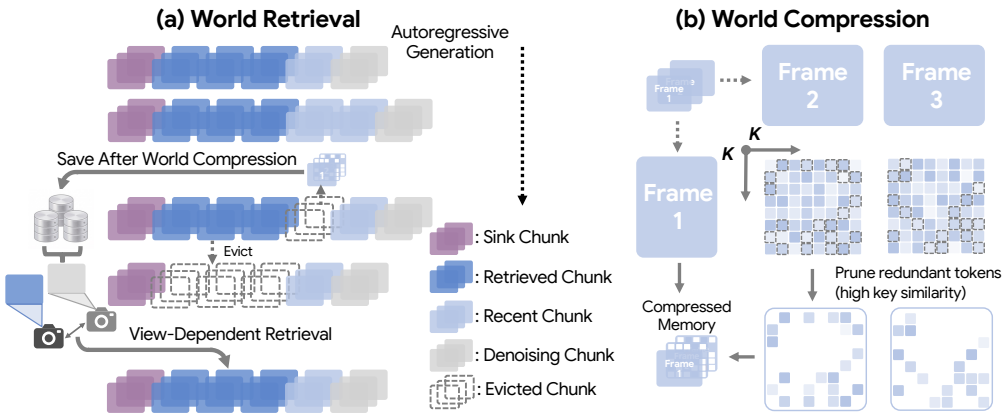


Figure 4: **Overview of WorldKV.** (a) World Retrieval stores KV-cache chunks after compression and retrieves view-relevant chunks back into the attention window at revisit time. (b) World Compression designates the first frame of each chunk as the anchor, computes key similarity against the remaining frames, and prunes tokens redundant with the anchor.

4.2 World Retrieval

View-Correspondent Attention under Camera/Action Revisits. We first analyze how autoregressive world models distribute attention over historical KV caches under camera/action input. We generate a sequence of 11 chunks following the trajectory "Right (chunks C0–C3) $\rightarrow$ Stop (C4) $\rightarrow$ Left (C5–C8) $\rightarrow$ Stop (C9) $\rightarrow$ Right (C10)" and visualize the chunk-level attention maps for Matrix-Game-2.0 (He et al., 2025) and LingBot-World-Fast (Team et al., 2026b) in Fig. 3. The maps reveal a clear view-correspondence pattern across both models. As the camera turns Left at C5–C8 and sweeps back toward the initial scene direction, attention rises on C0–C2, whose cached views overlap with the current viewpoint, as indicated by (1). At C9, where the camera stays near the initial viewpoint, attention concentrates on C0, the input image corresponding to that view, as marked by (2). When the camera turns right again at C10, attention shifts toward C5–C8, the chunks generated during the previous left-turn trajectory, as highlighted by (3). Additional attention maps are provided in Appendix K. These patterns suggest that attending to a compact set of viewpoint-relevant KV chunks, rather than simply the most recent caches, can preserve much of the important context provided by Full KV attention, motivating **World Retrieval**.

World Retrieval Mechanism. Motivated by the view-correspondence pattern observed above, World Retrieval operates as follows. Under sliding-window inference, KV cache chunks evicted from the active attention window are stored in GPU/CPU memory rather than discarded, each indexed by the camera/action state $\mathbf{a}_i$ at the time of its generation (absolute pose for camera models, cumulative discrete actions for keyboard action models). As illustrated in Fig. 4(a), the sliding window is partitioned into four regions: 1) *sink* KV caches from the initial frames that serve as a visual anchor, 2) *retrieved* KV caches selected from stored history, 3) *recent* KV caches from the immediately preceding frames, and 4) *denoising* chunk currently being generated. World Retrieval operates on the retrieved region: at generation time, given the current camera/action state $\mathbf{a}_{\text{cur}}$, the top-k most relevant chunks are selected from the stored history to fill this region:

$$\mathcal{R} = \text{Top-k}\left(\text{sim}(\mathbf{a}_{\text{cur}}, \mathbf{a}_i) \mid i = 1, \ldots, M\right), \tag{4}$$

where M is the number of chunks, k is the retrieval budget, and $\text{sim}(\cdot, \cdot)$ is a relevance function.

The framework is *retrieval-algorithm agnostic*: $\text{sim}(\cdot, \cdot)$ can be instantiated as camera/action-based similarity, query-based importance score, or other relevance methods. In this work, we evaluate camera/action-based and query-based retrieval in Appendix C; both substantially outperform sliding-window inference, demonstrating that the framework generalizes across retrieval signals.

4.3 World Compression

Motivation. World Retrieval requires storing all evicted KV caches in GPU/CPU memory for potential future retrieval. However, this storage cost is substantial: on LingBot-World-Fast, a single

chunk of 3 latent frames occupies approximately 3.4GB across all transformer layers, accumulating to over 200GB for a one-minute rollout, which exceeds even the VRAM capacity of a B200 GPU (Fig. 2 (a)). We observe that temporally adjacent frames share substantial visual content (Appendix B): camera viewpoint, scene layout, and object appearance change minimally over consecutive frames, producing near-duplicate KV caches that encode largely overlapping information. World Compression exploits this redundancy to reduce per-chunk storage while preserving the most distinctive KV caches. Beyond storage savings, this enables broader retrieval coverage within a fixed attention budget. As we show in Sec. 5.4 and Appendix D, broader coverage improves revisit fidelity.

Key-Key Similarity as a Redundancy Measure. World Compression requires a criterion for identifying redundant tokens within a short temporal chunk. We use Key-Key cosine similarity as a redundancy signal: we compare non-anchor frame keys against the anchor-frame keys, and find that keys from spatiotemporally overlapping regions exhibit high cosine similarity while keys from newly revealed or dynamic regions diverge (Appendix B). This finding is also consistent with prior evidence that keys in video diffusion transformers encode spatiotemporal correspondence (Yuan et al., 2025). We therefore prune high-similarity non-anchor tokens as redundant with the anchor, while retaining low-similarity tokens that carry distinctive content.

World Compression Mechanism. Given a chunk consisting of F consecutive frames, World Compression designates the first frame as the *anchor* and compresses the remaining $F-1$ frames against it. Let $\mathbf{K}^{(a)} \in \mathbb{R}^{T\times d}$ denote the T key vectors from the anchor frame at a given layer. For each non-anchor frame f, we measure the redundancy of each key $\mathbf{k}_j^{(f)}$ as its average cosine similarity to all anchor-frame keys:

$$s_j^{(f)} = \frac{1}{T}\sum_{i=1}^{T} \frac{{\mathbf{k}_j^{(f)}}^{\top}\mathbf{k}_i^{(a)}}{\|\mathbf{k}_j^{(f)}\| \cdot \|\mathbf{k}_i^{(a)}\|}. \tag{5}$$

We pool these scores across all non-anchor frames and retain the bottom $P\%$ among the pooled non-anchor tokens by similarity, since low similarity indicates content not captured by the anchor, such as newly revealed regions under camera motion. The compressed chunk consists of all anchor-frame tokens plus the retained tokens. With $F = 3$ and $P = 25\%$ retention across the $2T$ non-anchor tokens, each chunk shrinks from $3T$ to approximately $1.5T$ tokens, achieving $2\times$ storage efficiency.

Compression is applied once per chunk at storage time and operates independently per layer: each layer retains its own set of distinctive tokens, since token importance varies across layers. At retrieval time, each layer attends to its own retained tokens within the inserted chunk. Beyond storage efficiency, compression allows $2\times$ as much history to be retrieved within the fixed attention window, and we show in Sec. 5.4 that this broader coverage improves revisit fidelity.

5 Experiments

5.1 Experimental settings

Benchmark. To evaluate the world memory, we construct a benchmark of scene–trajectory pairs spanning diverse visual domains (e.g., indoor, outdoor, urban, natural). Initial frames are sourced from real-world videos, game recordings, and AI-generated images. For each scene, we design a long-horizon action sequence with at least one loop-closure, where the camera returns to a previously observed location, enabling direct evaluation of revisit consistency. We consider two settings distinguished by how that loop closure is reached: in the *Retracing* setting, the camera returns by reversing its path, while in the *Non-Retracing* setting, it returns along a different path, such as an asymmetric or curved one, reapproaching the location from a different direction. Each setting contains 60 scene-trajectory pairs. Trajectories and composition statistics are detailed in Appendix H.

Base Models. We evaluate on two autoregressive video world models at different scales: (1) LingBot-World-Fast (Team et al., 2026b) is a 14B-parameter model distilled from a long-video teacher capable of generating one-minute sequences; it natively operates with Full KV cache attention. (2) Matrix-Game-2.0 (He et al., 2025) is a 1.3B-parameter model that was not trained on long-context video; it natively operates with a sliding window of 6 latent frames.

Table 1: **Quantitative comparison.** We apply WorldKV to video world models, measuring revisit consistency between revisited and first-visit frames, on retracing and non-retracing trajectories. Throughputs are measured on 4×H200 GPUs by default; values in () are measured on 4×B200 GPUs.

Method	Throughput ↑ (FPS)	LPIPS ↓	PSNR ↑	SSIM ↑	FID ↓
Retracing					
With Memory Training					
WorldPlay (8B)	4.95	0.496	14.556	0.470	113.317
Yume-1.5 (5B)	11.41	0.566	13.167	0.467	141.704
Without Memory Training					
LingBot-World-Fast (14B)					
- Sliding Window	5.05 (8.10)	0.581	12.184	0.375	144.036
- Full KV (Baseline)	2.36 (3.83)	**0.441**	**15.901**	**0.472**	85.705
- **WorldKV (Ours)**	4.78 (7.71)(+2.42)	0.455(+.014)	15.660(-.241)	0.463(-.009)	**75.644**(-10.061)
Matrix-Game-2.0 (1.3B)					
- Sliding Window (Baseline)	18.87	0.594	11.422	0.280	157.261
- Full KV	7.82	0.529	13.748	0.364	124.912
- **WorldKV (Ours)**	16.25(-2.62)	**0.462**(-.132)	**14.101**(+2.679)	**0.405**(+.125)	**93.561**(-63.700)
Non-Retracing					
With Memory Training					
WorldPlay (8B)	4.95	0.536	14.453	0.463	103.322
Yume-1.5 (5B)	11.41	0.568	13.127	0.430	145.781
Without Memory Training					
LingBot-World-Fast (14B)					
- Sliding Window	5.05 (8.10)	0.596	11.313	0.408	138.272
- Full KV (Baseline)	2.36 (3.83)	**0.531**	**14.619**	0.473	101.960
- **WorldKV(Ours)**	4.78 (7.71)(+2.42)	0.533(+.002)	14.489(-.130)	**0.485**(+.012)	**97.351**(-4.609)
Matrix-Game-2.0 (1.3B)					
- Sliding Window (Baseline)	18.87	0.613	10.988	0.247	162.044
- Full KV	7.82	0.554	12.555	0.281	146.291
- **WorldKV(Ours)**	16.25(-2.62)	**0.546**(-.067)	**13.091**(+2.103)	**0.312**(+.065)	**140.798**(-21.246)

Baselines. For each base model, we compare against its native inference mode (Full KV cache attention for LingBot-World-Fast, sliding-window inference for Matrix-Game-2.0). We compare against WorldPlay (Sun et al., 2025) and Yume-1.5 (Mao et al., 2025), which were trained with memory modules. Since these models differ in backbone, scale, and training data, we include them as reference points rather than comparisons of the memory mechanism. Our main comparisons remain within the same backbone with different KV cache policies. A comparison with WorldPlay-style FOV retrieval on the same backbone is provided in Appendix C.

Implementation Details. We use a sliding window of 18 latent frames partitioned into sink (3 frames), retrieval (9 frames), recent (3 frames), and denoising (3 frames). World Compression retains the anchor frame in full and keeps 25% of tokens in non-anchor frames, compressing each 3-frame chunk to 1.5 frames. Full implementation details are provided in Appendix I.

Metrics. We measure PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018) between each revisit frame and the corresponding first-visit frame generated at the same viewpoint. For FID (Heusel et al., 2017), we compute the distributional distance between the set of revisit frames and the set of first-visit reference frames. Higher PSNR/SSIM and lower LPIPS/FID indicate better memory fidelity. We also report throughput (FPS) measured at the last chunk of each rollout.

5.2 Quantitative Results

Table 1 summarizes the main comparison on both retracing and non-retracing trajectories. WorldKV maintains throughput close to sliding-window inference, while Full KV cache attention drops to less than half this throughput due to linearly growing context length.

Table 2: **Ablation on World Compression.** (Top) Intra-chunk compression (frames retained per 3-frame chunk). (Bottom) Inter-chunk compression (chunks compressed into a fixed 3-chunk budget).

	LingBot-World-Fast				Matrix-Game-2.0			
Intra-Chunk	**LPIPS↓**	**PSNR↑**	**SSIM↑**	**FID↓**	**LPIPS↓**	**PSNR↑**	**SSIM↑**	**FID↓**
$3 \to 1.0$	0.494	14.807	0.435	97.155	0.474	13.951	0.385	94.424
$3 \to 1.25$	0.463	15.599	0.467	90.124	0.463	14.158	0.397	95.376
$3 \to 1.5$	0.455	15.660	0.463	75.644	0.462	14.101	0.405	93.561
$3 \to 2.0$	0.456	15.654	0.461	76.230	0.453	14.258	0.417	96.000
$3 \to 2.5$	0.456	15.685	0.466	77.482	0.450	14.216	0.417	95.478
$3 \to 3.0$	0.459	15.594	0.467	75.005	0.443	14.630	0.433	93.536

	LingBot-World-Fast				Matrix-Game-2.0			
Inter-Chunk	**LPIPS↓**	**PSNR↑**	**SSIM↑**	**FID↓**	**LPIPS↓**	**PSNR↑**	**SSIM↑**	**FID↓**
$3 \to 3$	0.468	15.436	0.454	91.389	0.496	13.416	0.369	105.988
$6 \to 3$	**0.455**	**15.660**	**0.463**	**75.644**	**0.462**	**14.101**	**0.405**	**93.561**
$9 \to 3$	0.482	14.982	0.430	101.226	0.499	13.403	0.360	108.460

On LingBot-World-Fast (Team et al., 2026b), which was distilled from a long-video teacher, Full KV cache attention already provides strong memory. On retracing trajectories, WorldKV closely approaches Full KV performance on LPIPS, PSNR, and SSIM and surpasses it on FID at roughly $2\times$ the throughput. On non-retracing trajectories, the gap narrows further: WorldKV is comparable to Full KV on LPIPS and PSNR while surpassing it on SSIM and FID, as restricting attention to view-relevant chunks becomes more beneficial when a larger fraction of the history is irrelevant to the current viewpoint (Appendix A).

On Matrix-Game-2.0 (He et al., 2025), WorldKV outperforms both sliding-window and Full KV cache attention across all metrics on both trajectory types. Full KV underperforms because Matrix-Game-2.0 was trained on short sequences: the accumulated KV cache contains degraded KV caches from out-of-distribution generation, and attending to all of them introduces compounding errors (Yi et al., 2025). WorldKV avoids this by retrieving only the KV caches relevant to the current scene.

WorldPlay (Sun et al., 2025) and Yume-1.5 (Mao et al., 2025) are included as reference points to assess how WorldKV changes each base model's performance relative to its original KV cache policy, rather than to compare memory mechanisms across backbones. On LingBot-World-Fast, Full KV cache already outperforms both memory-trained models on all metrics and trajectories, and WorldKV largely preserves this advantage. On Matrix-Game-2.0, sliding-window falls behind both models on all metrics, whereas WorldKV narrows this gap and surpasses both on LPIPS and FID on retracing.

5.3 Qualitative results

Figure 5 shows qualitative comparisons under multi-revisit trajectories. On LingBot-World-Fast, WorldKV closely matches Full KV, recovering scene appearance with high fidelity, and Appendix A shows cases where WorldKV reconstructs revisited scenes more faithfully than Full KV. On Matrix-Game-2.0, where Full KV degrades over long horizons, WorldKV produces sharper and more consistent results than Full KV. Sliding-window drifts visibly on both backbones, as evicted caches prevent access to the original viewpoint. Full comparisons including WorldPlay and Yume-1.5, with additional qualitative results, are provided in Appendix M.

5.4 Ablation studies

We ablate World Compression from two perspectives. First, the upper part of Table 2 varies the *intra-chunk* compression ratio, i.e., how many frame-equivalents are retained from each 3-frame chunk. The notation $3 \to r$ means that a 3-frame chunk is compressed to r frame-equivalents, ranging from anchor-only ($3 \to 1.0$) to no compression ($3 \to 3.0$); the latter corresponds to applying only World Retrieval(WR). Second, the lower part varies the *inter-chunk* coverage under a fixed 3-chunk-equivalent budget. $6 \to 3$ compresses 6 chunks into the same budget as 3 chunks, increasing history coverage without enlarging the attention window; the $3 \to 3$ row corresponds to the WR-only baseline. Additional ablations on the number of retrieved chunks are provided in Appendix D.

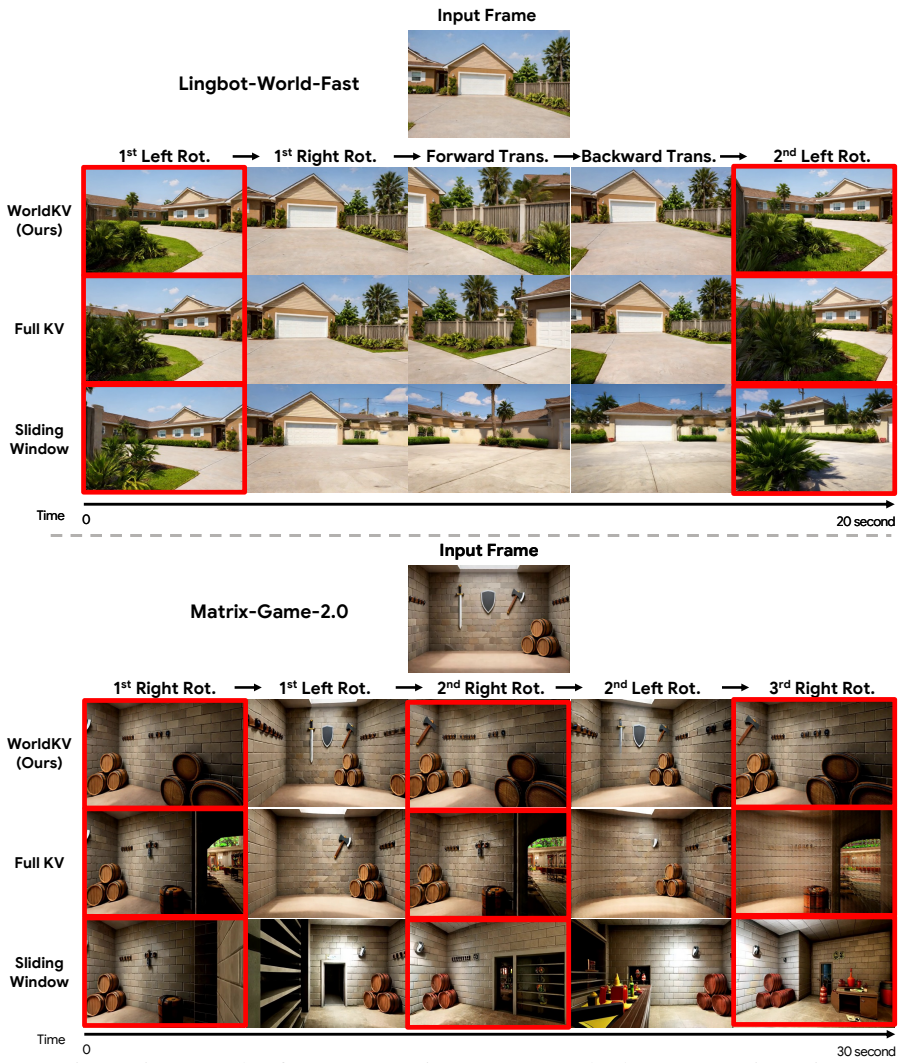


Figure 5: Frame-by-frame comparison across methods on two trajectories.

Intra-Chunk Compression Ratio Ablation. Table 2 (Top) shows the effect of varying the compression ratio. On both models, retaining only the anchor frame ($3 \rightarrow 1.0$) yields lower performance, confirming that non-anchor tokens contain distinctive information not captured by the anchor alone. Moderate compression ($3 \rightarrow 1.5$ or $3 \rightarrow 2.0$) achieves strong performance, while retaining more tokens gives limited gains. This suggests that World Compression preserves informative KV cache at practical compression ratios.

Inter-Chunk Compression Ratio Ablation. Table 2 (Bottom) compares compression scopes under a fixed 3-chunk retrieval budget. On both models, 6 chunks $\rightarrow$ 3 chunks outperforms uncompressed retrieval ($3 \rightarrow 3$), showing that broader historical coverage is more useful than preserving fewer chunks at full resolution. This agrees with Appendix D, where retrieving more chunks improves memory fidelity. In contrast, aggressive compression ($9 \rightarrow 3$) degrades performance by retaining only anchor frames and discarding distinctive non-anchor information.

6 Conclusion

We introduced WorldKV, a training-free framework for efficient world memory in autoregressive video world models through KV cache retrieval and compression. Building on our finding that these models already hold view-correspondent emergent memory in their KV caches, WorldKV achieves revisit fidelity comparable to or better than Full KV cache attention across two world models of different scales, at lower memory and attention cost and without any fine-tuning.

AI USE STATEMENT

In this work, we used generative AI tools for polishing writing and generating a few synthetic images for initial frames. We have not used generative AI tools for retrieval and discovery, research ideation or execution, drafting sections of the paper, and proving mathematical claims is not applicable to this work. Additionally, we used generative AI tools for formulating questions for surveys. We have reviewed all AI-assisted work. All AI-assisted edits and refinements were manually reviewed by the authors to ensure that they accurately reflected the intended technical content. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This paper studies efficient long-horizon inference with long-term memory for autoregressive video world models. By reducing the cost of maintaining and retrieving long-term visual memory, WorldKV may improve the practicality of interactive simulation, gaming, embodied AI, and robotic training environments, while reducing the memory and computation required by full-history attention. Because WorldKV is an inference-time memory-management method, it does not directly introduce new content-generation capabilities; however, it can make persistent interactive generation more efficient and accessible. Potential risks therefore follow those of generative video and interactive world models more broadly, including misuse for misleading synthetic content, unlabeled simulated media, or more realistic persistent virtual environments. Responsible deployment should include provenance, disclosure, and appropriate access controls when such systems are used in public-facing applications.

REPRODUCIBILITY STATEMENT

We provide the experimental details necessary to reproduce our main results. Section 5.1, Appendix H, and Appendix I describe the benchmark construction, evaluated base models and baselines, implementation details, retrieval and compression configurations, evaluation metrics, and hardware used for throughput measurement. Additional implementation details and analyses, including detailed latency and memory profiling, are provided in the Appendix.

REFERENCES

Zefan Cai, Wen Xiao, Hanshi Sun, Cheng Luo, Yikai Zhang, Ke Wan, Yucheng Li, Yeyang Zhou, Li-Wen Chang, Jiuxiang Gu, et al. R-kv: Redundancy-aware kv cache compression for reasoning models. *arXiv preprint arXiv:2505.24133*, 2025.

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.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, Qianli Ma, Seungjun Nah, Loic Magne, Jiannan Xiang, Yuqi Xie, Ruijie Zheng, Dantong Niu, You Liang Tan, K.R. Zentner, George Kurian, Suneel Indupuru, Pooya Jannaty, Jinwei Gu, Jun Zhang, Jitendra Malik, Pieter Abbeel, Ming-Yu Liu, Yuke Zhu, Joel Jang, and Linxi "Jim" Fan. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

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.

Xianglong He, Chunli Peng, Zexiang Liu, Boyang Wang, Yifan Zhang, Qi Cui, Fei Kang, Biao Jiang, Mengyin An, Yangyang Ren, et al. Matrix-game 2.0: An open-source real-time and streaming interactive world model. *arXiv preprint arXiv:2508.13009*, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Neural Information Processing Systems*, 2017. URL https://api.semanticscholar.org/CorpusID:326772.

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.

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. *arXiv preprint arXiv:2506.08009*, 2025b.

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.

Jiaqi Li, Junshu Tang, Zhiyong Xu, Longhuang Wu, Yuan Zhou, Shuai Shao, Tianbao Yu, Zhiguo Cao, and Qinglin Lu. Hunyuan-gamecraft: High-dynamic interactive game video generation with hybrid history condition, 2025a. URL https://arxiv.org/abs/2506.17201.

Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory. *arXiv preprint arXiv:2506.18903*, 2025b.

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. *arXiv preprint arXiv:2404.14469*, 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.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Xiaofeng Mao, Zhen Li, Chuanhao Li, Xiaojie Xu, Kaining Ying, Tong He, Jiangmiao Pang, Yu Qiao, and Kaipeng Zhang. Yume-1.5: A text-controlled interactive world generation model. *arXiv preprint arXiv:2512.22096*, 2025.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Georgy Savva, Oscar Michel, Daohan Lu, Suppakit Waiwitlikhit, Timothy Meehan, Dhairya Mishra, Srivats Poddar, Jack Lu, and Saining Xie. Solaris: Building a multiplayer video world model in minecraft. *arXiv preprint arXiv:2602.22208*, 2026.

Junyoung Seo, Hyunwook Choi, Minkyung Kwon, Jinhyeok Choi, Siyoon Jin, Gayoung Lee, Junho Kim, JoungBin Lee, Geonmo Gu, Dongyoon Han, et al. Grounding world simulation models in a real-world metropolis. *arXiv preprint arXiv:2603.15583*, 2026.

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.

Jiaming Tang, Yilong Zhao, Kan Zhu, Guangxuan Xiao, Baris Kasikci, and Song Han. Quest: Query-aware sparsity for efficient long-context llm inference. *arXiv preprint arXiv:2406.10774*, 2024.

InSpatio Team, Donghui Shen, Guofeng Zhang, Haomin Liu, Haoyu Ji, Hujun Bao, Hongjia Zhai, Jialin Liu, Jing Guo, Nan Wang, Siji Pan, Weihong Pan, Weijian Xie, Xianbin Liu, Xiaojun Xiang, Xiaoyu Zhang, Xinyu Chen, Yifu Wang, Yipeng Chen, Zhenzhou Fan, Zhewen Le, Zhichao Ye, and Ziqiang Zhao. Inspatio-world: A real-time 4d world simulator via spatiotemporal autoregressive modeling, 2026a. URL https://arxiv.org/abs/2604.07209.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, et al. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026b.

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.

Zhou Wang, Alan Conrad Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13:600–612, 2004. URL https://api.semanticscholar.org/CorpusID:207761262.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv*, 2023.

Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. Worldmem: Long-term consistent world simulation with memory, 2025. URL https://arxiv.org/abs/2504.12369.

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*, 2025.

Deheng Ye, Fangyun Zhou, Jiacheng Lv, Jianqi Ma, Jun Zhang, Junyan Lv, Junyou Li, Minwen Deng, Mingyu Yang, Qiang Fu, et al. Yan: Foundational interactive video generation. *arXiv preprint arXiv:2508.08601*, 2025.

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.

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. *arXiv preprint arXiv:2506.03141*, 2025.

Tianyu Yuan, Yuanbo Yang, Lin-Zhuo Chen, Yao Yao, and Zhuzhong Qian. Denoise to track: Harnessing video diffusion priors for robust correspondence. *arXiv preprint arXiv:2512.04619*, 2025.

Ted Zadouri, Markus Hoehnerbach, Jay Shah, Vijay Thakkar, and Tri Dao. Flashattention-4: Algorithm and kernel pipelining co-design for asymmetric hardware scaling. *Proceedings of Machine Learning and Systems*, 8:912–926, 2026.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018. URL https://api.semanticscholar.org/CorpusID:4766599.

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.

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.

A Cases Where WorldKV Outperforms Full KV Cache Attention

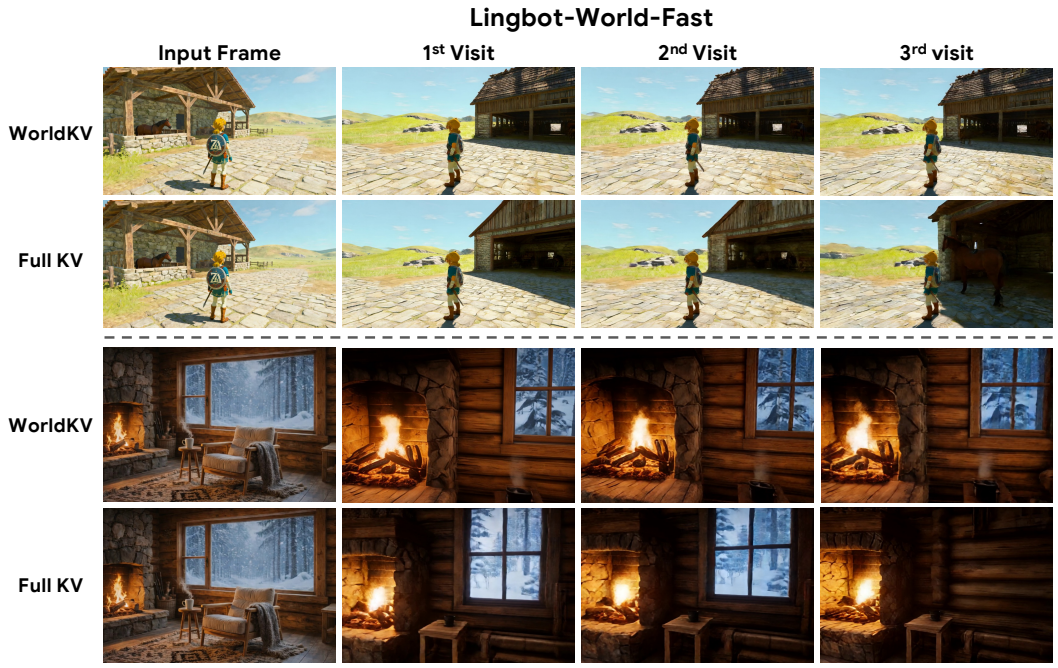


Figure 6: Each column shows the same viewpoint revisited at different times during a long-horizon rollout. Compared with Full KV cache attention, WorldKV preserves scene-specific details more faithfully across repeated visits by selectively retrieving viewpoint-relevant chunks and pruning redundant caches within each chunk.

While Full KV cache attention generally provides strong memory fidelity on LingBot-World-Fast (Team et al., 2026b), Fig. 6 shows cases where WorldKV reconstructs revisited viewpoints more faithfully than attending to the full history. We hypothesize that this is related to attention dilution: as the KV cache grows, the model must attend over many entries, including redundant or viewpoint-irrelevant ones, which may weaken effective access to the entries that encode the revisited scene. This interpretation is motivated by related findings in long-context language models. Lost in the Middle (Liu et al., 2024) shows that relevant information can be underutilized even when it is present in long contexts. Similarly, KV cache compression methods such as SnapKV (Li et al., 2024) and R-KV (Cai et al., 2025) show that compact caches can match, and in some cases outperform, full-cache attention by reducing irrelevant or redundant context. Although these results are obtained in language-model decoding, they suggest that full-history attention is not always optimal when much of the context is irrelevant or redundant.

In our setting, WorldKV follows this principle in dense spatiotemporal memory: it restricts the active attention window to retrieved viewpoint-relevant chunks and prunes near-duplicate entries within each chunk. To our knowledge, this is the first empirical observation of selective KV retrieval and compression outperforming Full KV cache attention in autoregressive video world models. These cases suggest that selective retrieval and compression can provide a cleaner effective context than full-history attention in some long-horizon world-model rollouts.

B Key-Key Similarity Visualization

Figure 7 visualizes the distinctive tokens selected by Key-Key similarity. For each scene, we use the first frame as the anchor and highlight the bottom 12.5% least-similar tokens from the 2nd and 3rd frames, based on cosine similarity to the anchor-frame keys. In camera-rotation cases (Fig. 7(a), (b), (c)), low-similarity tokens concentrate on newly revealed regions absent from the anchor frame, such as the left or right image boundaries exposed by camera motion. This indicates that high key

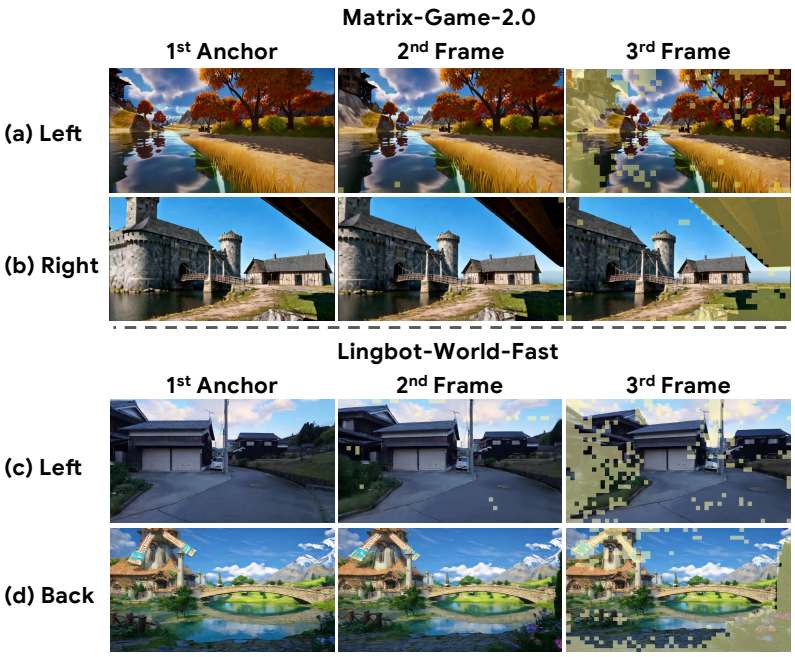


Figure 7: **Key-Key similarity visualization on Matrix-Game-2.0 and LingBot-World-Fast.** Yellow patches indicate tokens in the 2nd and 3rd frames with the lowest 12.5% cosine similarity to the anchor-frame keys. These low-similarity tokens correspond to information not present in the anchor, including newly revealed regions under camera motion and dynamic object changes, and are retained by World Compression.

similarity corresponds to redundant visual content, while low similarity identifies newly visible information that should be preserved.

Beyond newly revealed spatial content, Key-Key similarity also captures dynamic changes. In Fig. 7(d), where the camera moves backward, low-similarity tokens appear not only on newly visible scene regions but also on the rotating blades of the windmill. This suggests that key-similarity-based pruning can preserve distinctive temporal information from moving objects, rather than only static spatial differences. Overall, these observations support Key-Key similarity as an effective cue for redundancy within a chunk.

C Retrieval Algorithm Ablations

As described in Sec. 4.2, World Retrieval is retrieval-algorithm agnostic. Table 3 compares two retrieval strategies against the sliding-window baseline. The first is the camera/action-based retrieval used in our main experiments (Sec. 5.1), which selects chunks by camera pose or accumulated discrete action similarity. The second is query-based retrieval, inspired by Deep Forcing (Yi et al., 2025), which ranks stored chunks by their attention scores with respect to the current denoising query and therefore requires no explicit geometric signal such as camera pose or action. Both strategies substantially outperform sliding-window inference on both models, indicating that useful scene memory can be recovered from stored KV history across different retrieval signals. Camera/action-based retrieval outperforms query-based retrieval on both models, suggesting that viewpoint correspondence provides a particularly strong retrieval signal for interactive world models. We therefore use camera/action-based retrieval as our default and leave improved retrieval algorithms for future work.

Comparison with WorldPlay-style FOV retrieval. We also implement the FOV-based retrieval formulation of WorldPlay (Sun et al., 2025) on LingBot-World-Fast, which is conditioned on camera. The scenes, trajectories, context-window size, sink and recent regions, retrieval budget, and World Compression are identical across variants; only the retrieval function changes. As shown in Table 3, camera- and FOV-based retrieval perform nearly identically, and all three relevance signals (camera-,

Table 3: Retrieval Algorithm Ablation

Method	LPIPS ↓	PSNR ↑	SSIM ↑	FID ↓
LingBot-World-Fast (Camera-conditioned)				
Sliding	0.581	12.184	0.375	144.036
Camera / Action-based	0.455	15.660	0.463	75.644
Query-based	0.490	15.065	0.445	83.201
FOV-based	0.453	15.530	0.459	76.430
Matrix-Game-2.0 (Keyboard/Mouse Action-conditioned)				
Sliding	0.594	11.422	0.280	157.261
Camera / Action-based	0.462	14.101	0.405	93.561
Query-based	0.488	13.579	0.363	109.723

FOV-, and query-based) substantially outperform sliding-window inference. This experiment is not intended to claim that our camera-distance score is superior to the FOV score of WorldPlay. Rather, it supports the framework-level claim that World Retrieval is retrieval-algorithm-agnostic: view-relevant memory in the native KV cache is accessible through multiple relevance signals without retrieval-aware training.

D Increasing Retrieval Chunk Size

Fig. 8 shows the effect of increasing the number of retrieved chunks on memory fidelity. On both Matrix-Game-2.0 and LingBot-World-Fast, LPIPS, PSNR, and SSIM generally improve as more chunks are retrieved, indicating that broader access to historical KV caches improves revisit consistency. This trend further motivates World Compression: beyond reducing GPU/CPU storage cost, compression allows more historical chunks to fit within a fixed attention-window budget, expanding retrieval coverage.

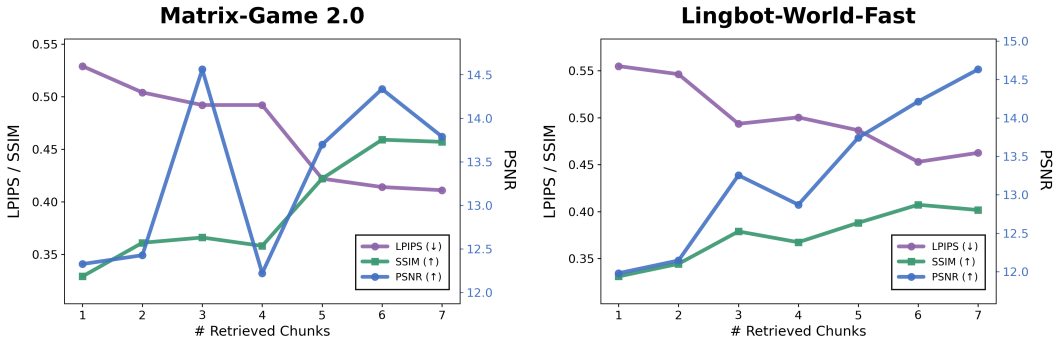


Figure 8: **Effect of increasing the number of retrieved chunks on memory fidelity.** Retrieving more past KV cache chunks generally improves reconstruction metrics on both models, motivating World Compression as a way to fit more historical chunks within a fixed attention-window budget.

E Comparison with Context-as-Memory and WorldPlay

Recent works, Context-as-Memory (Yu et al., 2025) and WorldPlay (Sun et al., 2025), also aim to improve world memory consistency through history retrieval, but differ in how retrieved history is represented and incorporated during generation.

Prior work: Reconstructing context under memory-specific training. Context-as-Memory (Yu et al., 2025) retrieves view-relevant frames by FOV overlap and concatenates them as conditioning context, re-encoding them with the VAE at every generation step. WorldPlay (Sun et al., 2025) ranks past chunks by geometric relevance, similar to WorldMem (Xiao et al., 2025), and reconstructs the context by recomputing the KV cache at each chunk from the latents. In both cases, the retrieved history must be reconstructed before it can be used, so the retrieval cost is incurred repeatedly rather

Table 4: Comparison of memory retrieval mechanisms for long-horizon autoregressive video generation. Unlike prior work, WorldKV requires no training and instead exploits an innate property of the model.

Aspect	Context-as-Memory	WorldPlay	WorldKV (Ours)
Stored historical representation	Past context frames	Latent chunks	Compressed KV cache
Model adaptation	Retrieval-aware training	Memory-aware distillation (Context Forcing)	**Training-free**
Context consumption	Re-encoded by the VAE at every step	Recomputed by the model	Inserted directly into attention
Retrieval criterion	Camera trajectories + FOV overlap	WorldMem-style Retrieval	Camera/action-based, FOV-based, query-based retrieval
Contribution	Training mechanism & retrieval algorithm	Distillation & Training mechanism	**Innate property** (emergent view correspondent)

than once, which directly reduces real-time throughput. Both methods also require training: retrieval-aware training in Context-as-Memory (Yu et al., 2025), memory-aware distillation in WorldPlay (Sun et al., 2025).

WorldKV: Retaining and compressing states the model has already computed. In contrast, our method operates *training-free* at inference time. Rather than reconstructing context, World Retrieval reinserts view-relevant KV cache chunks into native attention without re-encoding or recomputation. Additionally, World Compression exploits the spatio-temporal correspondence encoded in video diffusion keys to prune redundant tokens within each retained chunk via key–key similarity. This compression mechanism has no direct counterpart in prior memory-based world models(Yu et al. (2025), Sun et al. (2025)) and approximately doubles the retrievable history under a fixed memory budget. Together, KV cache reuse and compression enable retrieval of more historical context within a fixed attention budget while maintaining real-time throughput. WorldKV leverages the view-correspondent attention patterns already present in the pretrained models (Sec. 4.2), requiring no memory-specific training. Its retrieval mechanism accommodates multiple relevance criteria, including camera/action-based, FOV-based, and query-based scoring (Appendix C).

Summary. Context-as-Memory (Yu et al., 2025) retrieves and re-encodes historical frames using a model trained with retrieval-aware conditioning, also WorldPlay (Sun et al., 2025) retrieves latent chunks and recomputes their KV states using a model distilled with memory-aware context. WorldKV instead directly reuses and compresses previously computed KV states, without additional training or context recomputation.

F Latency and memory-footprint breakdown

We profile the per-component cost of WorldKV to verify that the overhead of retrieval and compression remains small relative to the attention cost they save. Measurements are taken with LingBot-World-Fast (Team et al., 2026b) as the base model on 4×B200 GPUs with FlashAttention-4 (Zadouri et al., 2026), where each generation chunk produces 3 latent frames (approximately 12 output RGB frames). All values are reported at the final chunk of each rollout, i.e. under the largest KV bank the rollout reaches. We report two configurations: a GPU-resident KV bank, and a CPU-resident KV bank in which retrieved chunks are transferred to the device on demand. Throughput in the GPU-resident configuration corresponds to the setting used in the main paper.

Table 5 yields two observations. First, retrieval search is negligible: at roughly 0.1% of per-chunk latency even at the largest bank size, top-k selection over stored chunk descriptors does not become a bottleneck as rollouts lengthen. Second, self-attention cost is unchanged between configurations (approximately 180 ms), confirming that the bounded attention window of WorldKV keeps attention cost constant regardless of how much history is stored. Offloading the bank to host memory removes

Table 5: **Per-component latency and memory footprint.** Retrieval search accounts for roughly 0.1% of per-chunk latency, and self-attention cost is unchanged between configurations.

Measurement	GPU-resident KV bank	CPU-resident KV bank
Retrieval search	1.963 ms	2.385 ms
CPU↔GPU KV movement	0 ms	644.169 ms
Store-time compression	31.076 ms	30.880 ms
FA4 self-attention	179.813 ms	181.202 ms
Total chunk latency	1556.698 ms	2547.217 ms
Throughput	7.71 FPS	4.71 FPS
KV cache memory footprint	36.55 GB / GPU	146.2 GB (host)

Table 6: **Human preference study.** Each entry reports the percentage of comparisons in which WorldKV was preferred over the corresponding baseline. Values above 50% favor WorldKV.

vs. Baseline	Scene-level revisit	Object-level persistence	Visual quality
Matrix-Game-2.0			
Sliding Window	97.5%	97.5%	77.5%
Full KV	85.0%	85.0%	87.5%
Yume-1.5	60.0%	57.5%	42.5%
WorldPlay	42.5%	40.0%	47.5%
LingBot-World-Fast			
Sliding Window	100.0%	100.0%	100.0%
Full KV	52.5%	50.0%	52.5%
Yume-1.5	85.0%	82.5%	65.0%
WorldPlay	60.0%	55.0%	52.5%

the per-GPU cache footprint at the cost of transfer latency, which dominates the chunk budget and reduces throughput below real time; we therefore keep the bank GPU-resident by default.

G Human Preference Study

Reconstruction metrics (PSNR, LPIPS, SSIM, FID) computed against the originally generated frame measure self-consistency, but do not distinguish the reproduction of generation artifacts from the preservation of scene content. We therefore conducted a human preference study along three axes: *i) scene-level revisit*, whether a revisited location preserves the overall layout, large-scale structures, and viewpoint geometry of the previously seen scene; *ii) object-level persistence*, whether individual objects retain their identity and appearance without disappearing, being replaced, or newly appearing; and *iii) visual quality*, whether the generated video is visually coherent, plausible, and stable, with minimal artifacts or distortions, independently of its consistency with earlier frames. We conducted the study with 20 participants. Each participant evaluated 16 video comparisons and answered three questions for each comparison, one for each evaluation axis. Figure 13 shows the interface presented to participants. The results are summarized in Table 6.

The study is broadly consistent with our quantitative results. On Matrix-Game-2.0, WorldKV is preferred over both sliding-window and Full KV inference on all three axes, matching Table 1 where it outperforms Full KV on all four metrics. On LingBot-World-Fast, WorldKV is preferred over sliding-window inference, while human raters show no clear preference between WorldKV and Full KV attention. The WorldPlay and Yume-1.5 rows are reference points rather than the basis of our claim.

H Benchmark Composition and Non-Retracing Trajectories

We characterize the 60 retracing scene–trajectory pairs of our benchmark along three axes. *Large translation*: the trajectory includes sustained forward or lateral motion that carries the camera substantially away from its origin, rather than rotation in place. *Dynamic objects*: moving or deformable content is present in the traversed region, including third-person agents, vehicles, and

Table 7: Composition of the 60 scene–trajectory pairs in the Retracing setting. Categories are not mutually exclusive.

Category	Proportion
Large translation	33%
Dynamic objects	20%
Repeated / ambiguous structures	5%

NPCs. *Repeated or ambiguous structures*: the scene contains at least two spatially distinct locations that are visually indistinguishable from the generated frames alone, such as forests and open fields. Table 7 reports the proportion of pairs in each category.

Each trajectory contains at least one loop-closure event. Since retracing places the revisited view under a camera pose close to the original one, it does not test whether retrieval remains effective when a location is reapproached from a different direction. We therefore additionally evaluate the **Non-Retracing** setting, described below.

Non-Retracing trajectories. Since revisit consistency can only be measured when the camera returns to a previously observed location, every non-retracing trajectory contains at least one loop closure; unlike in the Retracing setting, the revisited location is reapproached along a different path. This setting consists of five trajectories: (1) an asymmetric sequence combining forward, lateral, and backward motion; (2) a translation-dominant rectangular loop; (3) a sequence interleaving forward motion, 90° rotations, backward motion, and lateral translation; (4) a lateral loop combined with an extended out-and-back revisit; and (5) a curved out-and-back path with a continuously changing viewing direction. Results on these trajectories are reported in Sec. 5.2.

I Action-to-Pseudo Pose Conversion & Camera Pose-based Retrieval Algorithm

World Retrieval ranks chunks by camera poses and actions: we use the input poses for LingBot-World-Fast (Team et al., 2026b), and for Matrix-Game-2.0 (He et al., 2025), which accepts discrete keyboard and mouse inputs, we accumulate WASD and yaw/pitch commands into pseudo-camera poses. While these are not calibrated to scene geometry, they capture the relative camera motion induced by the action sequence, making them effective for retrieval.

Action-to-Pseudo Camera Pose Conversion. For Matrix-Game-2.0 (He et al., 2025), which provides discrete keyboard and mouse inputs rather than camera poses, we construct a pseudo-camera trajectory from the action sequence. Let $\mathbf{u}_t$ denote the planar displacement induced by the WASD input, and let $(\Delta\theta_t, \Delta\phi_t)$ denote the mouse-induced yaw and pitch changes. We accumulate the rotations and integrate the displacement along the current viewing direction:

$$\theta_t = \theta_{t-1} + \Delta\theta_t, \qquad \phi_t = \phi_{t-1} + \Delta\phi_t, \qquad \mathbf{x}_t = \mathbf{x}_{t-1} + R(\theta_t)\mathbf{u}_t, \tag{6}$$

where $R(\theta_t)$ is a planar rotation matrix. Each latent chunk is represented by its mean pseudo-translation $\overline{\mathbf{x}}_j$ and pseudo-rotation $\overline{\mathbf{r}}_j = (\overline{\theta}_j, \overline{\phi}_j)$.

Although not geometrically calibrated, this pseudo-trajectory captures the relative camera motion induced by the actions and serves as a surrogate for camera-pose-based retrieval. We convert $(\overline{\theta}_j, \overline{\phi}_j)$ into a rotation matrix $\bar{R}_j$ and set $\bar{t}_j = \overline{\mathbf{x}}_j$, using $(\bar{t}_j, \bar{R}_j)$ as the pseudo-camera pose of chunk j in the retrieval algorithm below.

Camera Pose-based Retrieval Algorithm. World Retrieval ranks previously generated KV cache chunks according to their geometric relevance to the current chunk using the corresponding camera trajectory. We associate each chunk with its mean camera center $\bar{t}_j$ and orientation $\bar{R}_j$, and measure translation and orientation differences between a candidate historical chunk j and the current chunk q as

$$d_t(j,q) = \|\bar{t}_j - \bar{t}_q\|_2^2, \qquad d_R(j,q) = \cos^{-1}\left(\mathrm{clip}\left(\frac{\mathrm{tr}(\bar{R}_j^\top \bar{R}_q) - 1}{2}, -1, 1\right)\right). \tag{7}$$

Table 8: **VBench evaluation.**

Method	Motion Smoothness ↑	Overall Consistency ↑	Imaging Quality ↑	Aesthetic Quality ↑	Subject Consistency ↑	Background Consistency ↑
Matrix-Game-2.0						
- Sliding Window	97.67	–	58.63	52.49	91.41	93.76
- **WorldKV**	97.77	–	60.33	55.16	91.52	93.91
- Full KV	97.74	–	56.09	50.94	91.24	93.70
LingBot-World-Fast						
- Sliding Window	97.07	10.43	62.46	57.85	93.31	95.44
- **WorldKV**	97.04	10.51	65.40	59.69	93.46	95.59
- Full KV	96.99	10.59	65.47	59.58	93.53	95.70

We normalize the two distances separately over the candidate set and combine them with equal weight:

$$D(j, q) = \hat{d}_t(j, q) + \hat{d}_R(j, q), \tag{8}$$

where $\hat{d}_t$ and $\hat{d}_R$ denote d_t and d_R normalized by their respective maxima over the candidate set.

We retrieve the chunks with the smallest scores under the retrieval budget, excluding persistent attention-sink and recent chunks already covered by the local KV cache.

Pose Normalization. For model conditioning, trajectories are expressed relative to the initial pose and converted into frame-to-frame transformations, with translations normalized by the maximum frame-to-frame translation norm and rotations left unnormalized. World Retrieval instead operates on absolute (pseudo-)poses, and the retrieval distances above are normalized by their maxima over the candidate set.

J Video Quality Evaluation

To verify that WorldKV does not compromise visual quality while maintaining or improving world memory, we additionally evaluate long-horizon video quality using VBench-Long (Huang et al., 2024). Following prior works on long video generation (Huang et al., 2025b; Yi et al., 2025), we report six quality dimensions: Motion Smoothness, Overall Consistency, Imaging Quality, Aesthetic Quality, Subject Consistency, and Background Consistency. Since Overall Consistency measures video–text alignment and Matrix-Game-2.0 is not conditioned on text, we omit this dimension for Matrix-Game-2.0.

Table 8 summarizes the results. On Matrix-Game-2.0 (He et al., 2025), WorldKV achieves the best performance across all five dimensions. Notably, Full KV underperforms even Sliding Window in frame-level quality (i.e., imaging and aesthetic quality), indicating that retaining the entire context can instead degrade generation quality. By retaining only a bounded set of relevant KV chunks, WorldKV avoids this degradation, consistent with its advantage over Full KV in world memory (Table 1). On LingBot-World-Fast (Team et al., 2026b), WorldKV achieves performance comparable to Full KV across all dimensions, while outperforming Sliding Window on most of the dimensions. These results demonstrate that, as with world memory, WorldKV maintains or even improves visual quality relative to Full KV.

K Additional Attention Visualization

We provide additional attention map visualizations in Fig. 12, complementing those in Fig. 3. We consider three trajectories comprising left/right rotations and forward/backward translations. Across these trajectories, attention over historical chunks consistently follows viewpoint overlap, further supporting the view-correspondent attention pattern observed in Section 4.2.

For the two rotational revisit trajectories (Fig. 12(i)–(ii)), the camera first turns in one direction, reverses toward the initial viewpoint, and then turns again in the original direction. Consistent with Fig. 3, as the camera returns toward a previously observed viewpoint, attention increases on the historical chunks generated near that viewpoint.

In (iii), following the trajectory "Right (chunks C0–C3) → Stop (C4) → Left (C5–C8) → Stop (C9) → Forward (C10)", the attention pattern during the initial right and left turns resembles that of (ii). However, as the camera moves forward at C10 toward a region not covered by previous chunks, the current viewpoint shares little overlap with the cached views except for the initial view. Accordingly, attention remains low on most past chunks, concentrating only on C0 and the most recent chunks.

Overall, these patterns show that historical chunks whose viewpoints overlap with the current viewpoint receive high attention, further supporting the view-correspondent attention pattern observed in Section 4.2.

L WorldKV in Inspatio-World

To demonstrate that WorldKV generalizes beyond the two base models used in our main evaluation, we apply it to Inspatio-World (Team et al., 2026a), a video-to-video 4D world model that generates novel-view sequences conditioned on an input video. Inspatio-World natively maintains memory for the input video by placing it in the sink region of the attention window, but has no mechanism to preserve memory for newly generated scenes. Like Matrix-Game-2.0 (He et al., 2025), the model was not trained on long video sequences. As shown in Fig. 9, applying WorldKV enables Inspatio-World (Team et al., 2026a) to preserve long-term memory consistency across revisits. This shows that WorldKV is not tied to specific models and can apply broadly to KV cache-based autoregressive world models.



Figure 9: Applying WorldKV to Inspatio-World (Team et al., 2026a), a video-to-video 4D world model. Top: input video with a fixed camera. Middle: Inspatio-World generates novel-view sequences from the input video but loses scene memory upon revisit. Bottom: with WorldKV applied, the same scene content is preserved consistently across views without any fine-tuning.

M Additional Qualitative Comparisons

Fig.10 presents the full qualitative comparison, including WorldPlay (Sun et al., 2025) and Yume-1.5 (Mao et al., 2025), complementing the comparisons in the main paper. Additional qualitative results of WorldKV are presented in Fig. 11. These examples show that our training-free retrieval and compression framework maintains consistent scene revisits across long-horizon trajectories, achieving qualitative results comparable to Full KV cache attention while attending only to a bounded context window. We also provide video comparisons in the supplementary material. Additional qualitative results and video demonstrations are available on our anonymous project page: `https://wkviclr.github.io/worldkv__/`.



Figure 10: Frame-by-frame comparison across methods on two trajectories.

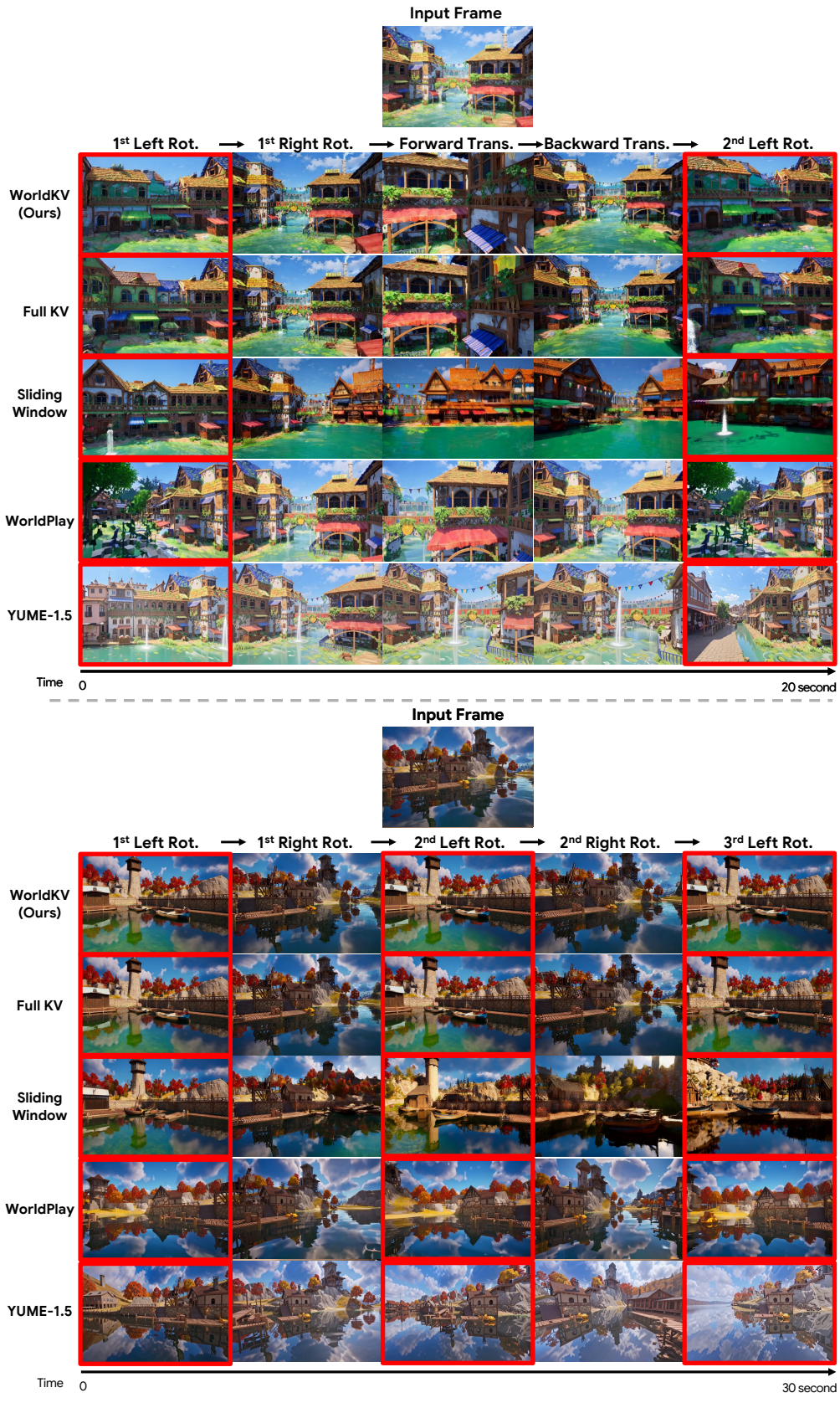


Figure 11: Frame-by-frame comparison across methods on two trajectories.

N LIMITATIONS & FUTURE WORK

As a KV cache management method, WorldKV assumes that attention over past context is causal or block-causal and that key–value states persist in a reusable cache; world models that break either assumption fall outside our scope. World Retrieval may fail to retrieve the relevant historical context in highly dynamic scenes or when camera/action similarity is not a reliable indicator of visual relevance. The query-based criterion (Appendix C) may help mitigate this limitation by estimating visual relevance from attention.

WorldKV improves the efficiency and fidelity of long-horizon world-model inference, while remaining an inference-time memory-management method. Since it operates on the KV cache of a frozen backbone, its visual fidelity remains bounded by the generation quality of the underlying pretrained world model. In rollouts substantially longer than those seen during training, autoregressive video generation may still accumulate visual artifacts caused by error accumulation. WorldKV focuses on efficient preservation and retrieval of past visual memory, and future work may combine it with training strategies for more stable multi-minute world generation.

As shown in Fig. 2 (a), CPU offloading offers a complementary direction for reducing VRAM cost: all KV caches are stored in CPU memory, and only the chunks needed for current attention are loaded onto the GPU. This bounds VRAM consumption regardless of rollout length. However, the host-device transfer latency at retrieval time currently prevents real-time generation. We leave reducing this offloading latency to future work, which would enable real-time multi-minute world generation under bounded VRAM.

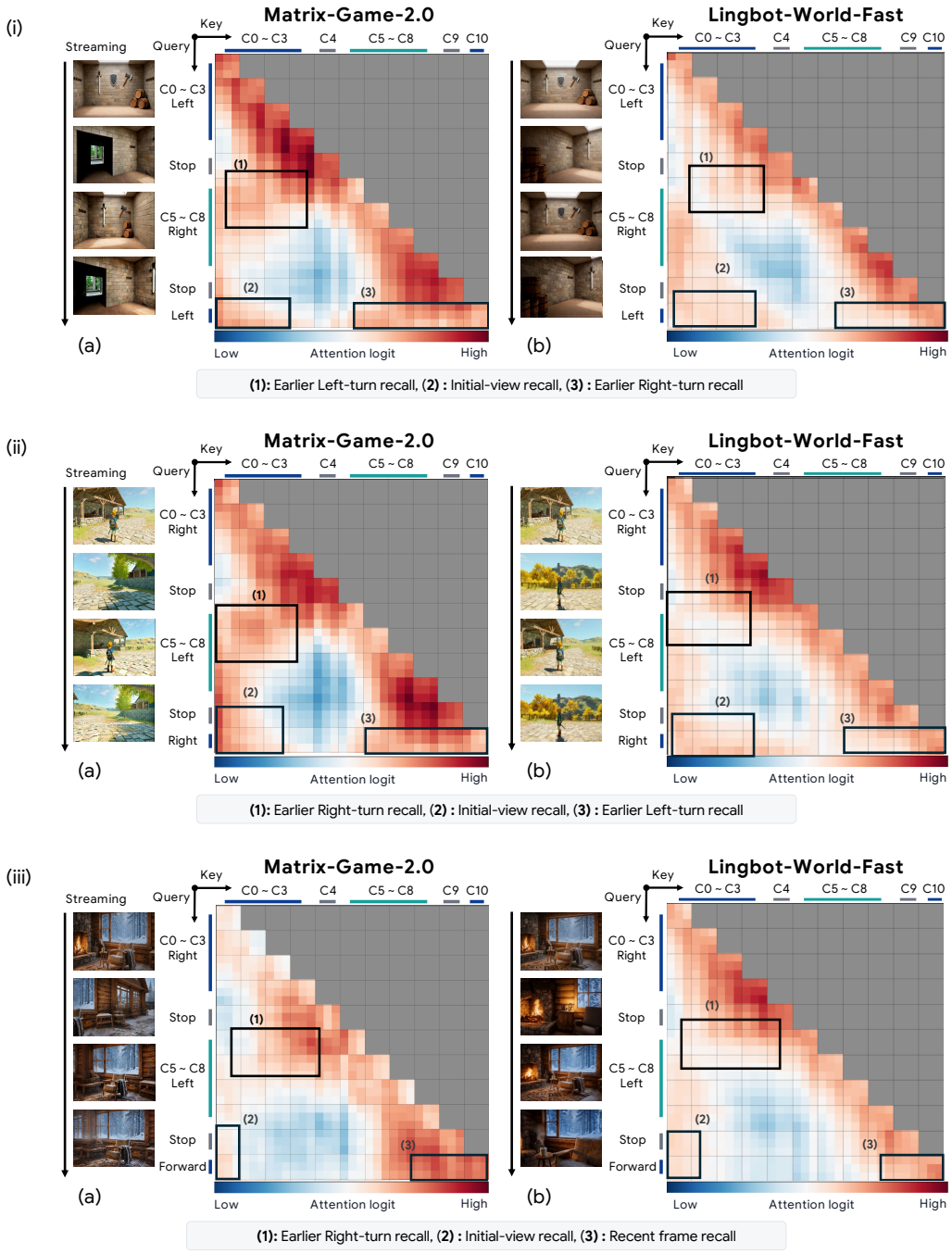


Figure 12: **Additional attention maps for the action sequences.** These patterns motivate using camera/action as the retrieval criterion for selecting relevant past KV caches.

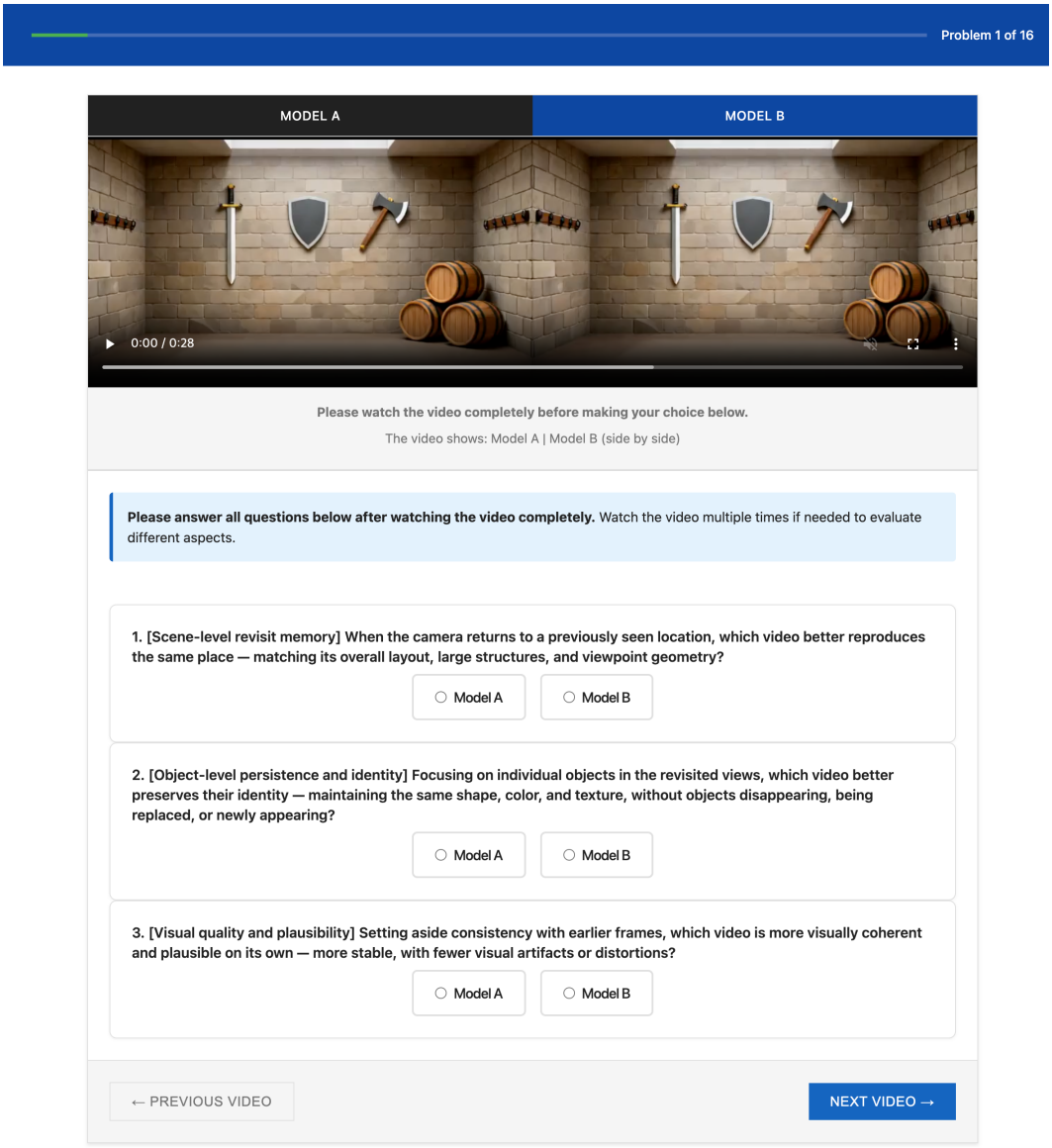


Figure 13: Example of the user interface for the user study.