TRACEWORLD: VIEW-ADAPTIVE MEMORY ALLOCATION FOR SPATIALLY CONSISTENT VIDEO WORLD MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Camera-controlled video world models aim to simulate coherent visual environments across changing viewpoints. When the camera revisits a previously observed region, maintaining this coherence requires scene information that may no longer be available in recent temporal context. We introduce TraceWorld, a video world model with view-adaptive key-value (KV) memory allocation for camera-controlled generation. To incorporate camera geometry while preserving the pretrained visual pathway, TraceWorld adds a decoupled camera-aware attention branch. Pose-affinity retrieval then selects historical latent frames and reuses their cached KV states across both attention branches. A learned allocation policy divides a fixed context budget between recent and retrieved KV states, conditioned on inter-chunk camera-pose changes and the pose affinity between target views and stored history. Together, these components balance local temporal continuity with scene recall without constructing an explicit 3D scene memory. To evaluate scene consistency under camera revisits, we introduce the Scene Recall Benchmark (SRBench), comprising 200 camera-return cases across eight motion categories. On WorldScore, TraceWorld achieves the highest object control and 3D consistency scores among the compared methods. On SRBench, it achieves the highest revisit consistency and margin while remaining competitive in stability and camera following.

1 INTRODUCTION

Video world models generate visual sequences conditioned on camera motion or actions, supporting applications in embodied intelligence, interactive content creation, and environment exploration. Recent video diffusion transformers produce increasingly realistic videos (Peebles & Xie, 2023; OpenAI, 2026; Kong et al., 2024; Yang et al., 2024; Wan et al., 2025), and autoregressive generation extends their temporal range by synthesizing successive causal chunks (Gao et al., 2024; Henschel et al., 2025; Jin et al., 2025; Valevski et al., 2025). However, generating a longer video does not necessarily preserve a persistent world. When the camera leaves a region and later returns, previously observed structures and appearance may drift or be replaced. Maintaining a coherent environment therefore requires both local temporal continuity and the ability to recover scene information across distant observations.

Memory allows a video world model to retain earlier observations and reuse them when the camera returns to previously seen regions. The context needed for generation, however, changes between exploration and revisits. Recent frames support smooth continuation into new regions, whereas earlier observations from matching viewpoints provide scene references needed at revisits. Effective memory use therefore requires adapting this balance to camera motion and the availability of relevant historical observations.

Figure 1 contrasts two common ways to make past scene information available to generation. Video-based methods retrieve historical observations according to temporal, content, or camera relevance, including camera-overlap retrieval (Yu et al., 2025b). Other methods maintain point clouds or latent 3D representations that supply scene references for target views (Wu et al., 2026; Zhao et al., 2026; Wang et al., 2026c), requiring additional geometry construction and rendering or update procedures.

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.

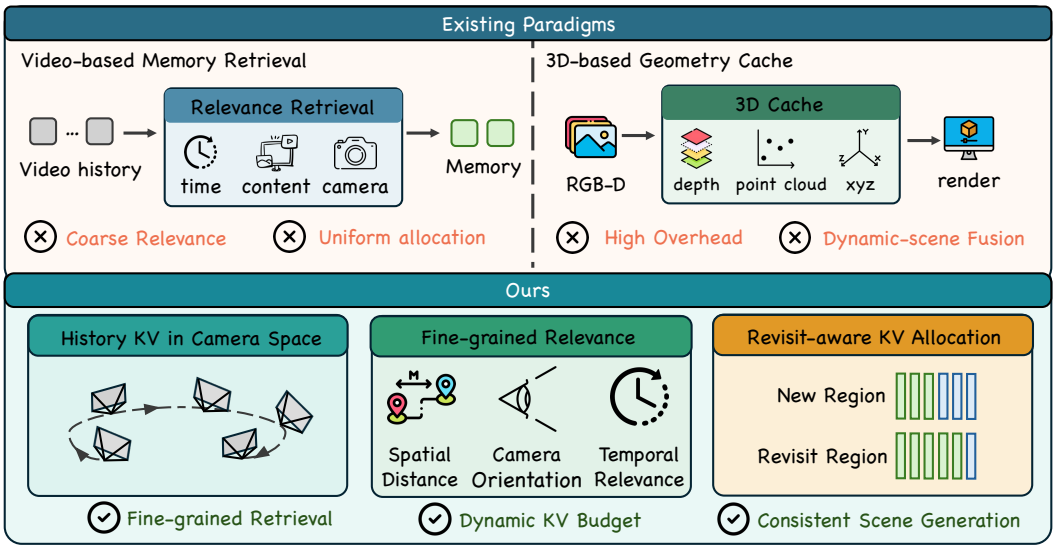

Figure 1: Comparison of memory designs. Video-based methods retrieve historical observations, while geometry-based methods maintain explicit 3D scene caches. TraceWorld reuses native KV states and adaptively allocates a fixed context budget between recent observations and history retrieved by camera pose affinity. The new-region and revisit examples illustrate how context needs change with camera motion.

Our approach reuses the generator's native key-value (KV) states as memory and learns how to divide a fixed active context budget between recent observations and retrieved history. Camera pose affinity identifies relevant historical states within this allocation framework, providing access to earlier views without maintaining a separate 3D scene representation.

We introduce **TraceWorld**, a video world model with view-adaptive KV memory allocation for camera-controlled autoregressive generation. Our key idea is to treat the generator's native KV states as retrievable memory and learn how to divide a fixed historical context budget between recent and retrieved states. The policy uses inter-chunk camera-pose changes and historical retrieval relevance to balance local temporal continuity with scene recall.

To be specific, we augment the pretrained visual pathway with a decoupled camera-aware attention branch using PRoPE (Li et al., 2025; Team et al., 2026) to condition generation on the target camera trajectory. Both branches maintain their own KV caches, with each historical latent frame's states associated with its source camera pose. To retrieve relevant history, we rank these frames by camera-center proximity and forward and up direction agreement with the target views. Frames already retained as recent context are excluded from retrieval to avoid duplicate selection. Both branches use the same selected frame indices and attend to the corresponding KV states in their respective caches. This design reuses native attention states to provide temporally distant scene references without constructing an explicit 3D scene memory.

Our evaluation examines both general world-generation quality and the recovery of previously observed scenes. We complement WorldScore (Duan et al., 2025) with the Scene Recall Benchmark (SRBench), whose 200 camera-return cases across eight motion categories test whether scene content is preserved when the camera returns to earlier viewpoints. TraceWorld performs in the leading group on general world generation, with particular strengths in object control and geometric consistency. Its advantage becomes more pronounced at revisits, where it recovers prior scene content more consistently while maintaining competitive temporal stability and camera following.

Our contributions are threefold:

- We introduce TraceWorld, a video world model with a learned view-adaptive allocation policy that dynamically divides a fixed historical KV budget between recent context and retrieved history, balancing local continuity with scene recall.

- We combine pose-affinity retrieval with decoupled visual and camera-aware attention to select and integrate historical latent-frame KV states, providing relevant scene references without an explicit 3D scene memory.
- We construct the 200-case Scene Recall Benchmark (SRBench) for camera-return consistency and demonstrate improved scene recall with competitive temporal stability through evaluation on SRBench and WorldScore.

2 Related Work

2.1 Video Generation and World Modeling

Video generation has progressed from UNet-based latent diffusion architectures (Blattmann et al., 2023) to scalable Diffusion Transformers (DiTs) (Peebles & Xie, 2023; OpenAI, 2026; Kong et al., 2024; Yang et al., 2024; Wan et al., 2025; Google DeepMind, 2026; Kuaishou, 2026; Seedance et al., 2026). Interactive video world modeling further requires generation to respond to camera commands or actions. Each new chunk or frame must therefore be generated from the available history and controls without relying on future observations, motivating a causal formulation across successive chunks. Bidirectional models (Zheng et al., 2024b; Minimax, 2026) jointly model tokens within a video clip through full spatiotemporal attention, while autoregressive models extend a rollout by generating successive segments conditioned on earlier ones.

Autoregressive frameworks (Gao et al., 2024; Henschel et al., 2025; Jin et al., 2025; Valevski et al., 2025) generate frames, clips, or latent chunks in causal order. This formulation supports continued interaction and extends generation beyond the backbone's native temporal window, but local errors can accumulate and historical scene information can fade over successive chunks. Recent methods (Chen et al., 2024a; 2025; Liu et al., 2026; Zhou et al., 2025a; Zhang et al., 2026) improve context propagation, frame packing, streaming generation, or history-guided denoising. Our work focuses on how to allocate the available historical context between recent frames for local continuity and earlier observations relevant to the current views.

2.2 Camera-Controlled Video Generation

Camera-controlled video generation synthesizes videos that follow user-specified trajectories. Existing methods introduce camera conditions through motion embeddings, control branches, trajectory-aware adapters, or generative rendering formulations (Wang et al., 2024; He et al., 2024; Van Hoorick et al., 2024; Bai et al., 2025). AC3D (Bahmani et al., 2025) further studies where and when to inject pose conditions into video diffusion transformers. To improve consistency across changing viewpoints, CamI2V uses epipolar attention for cross-frame feature exchange, while Cavia integrates attention across multiple camera-controlled views (Zheng et al., 2024a; Xu et al., 2024a). Geometry-aware approaches also inject camera-ray geometry into attention, as in ViewRope (Xiang et al., 2026) and PRoPE (Li et al., 2025), to align visual representations with projective geometry under camera motion. CameraCtrl II (He et al., 2025) extends camera-controlled generation to iterative scene exploration by conditioning new clips on previously generated content and newly specified trajectories.

Following this line of camera conditioning, we adopt a dual-branch attention design that augments the pretrained visual branch with a decoupled camera-aware branch. The camera-aware branch incorporates camera geometry through PRoPE, and its output is added to the visual branch output to guide generation along the target camera trajectory.

2.3 Memory in Video World Models

Existing video world models use different forms of memory to extend context beyond a short temporal window. Context-as-Memory (CaM) (Yu et al., 2025b) stores historical video observations in observation space and retrieves them according to camera overlap between past and target views.

Another line of work constructs persistent spatial representations. Video World Models with Long-term Spatial Memory (Wu et al., 2026) organize history into working, spatial, and episodic memory. Spatia (Zhao et al., 2026) maintains an updatable 3D point-cloud memory and conditions generation

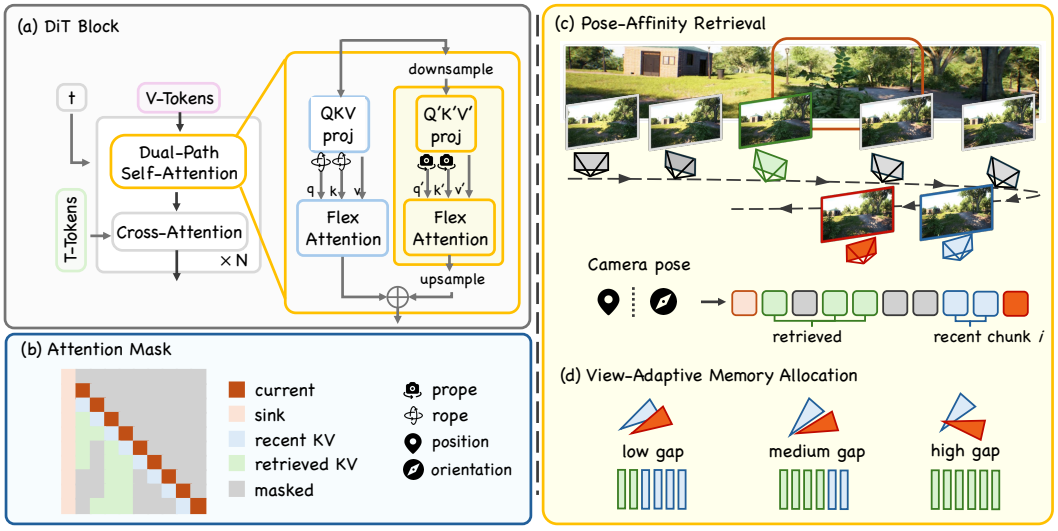

Figure 2: Overview of our framework. (a) A decoupled PRoPE branch augments the base visual pathway with camera-aware attention. (b) Attention masks retain the selected history and prevent access to future frames. (c) Pose-affinity retrieval ranks historical latent frames by their camera poses relative to the current views. (d) A learned view-adaptive memory allocation policy adaptively divides a fixed historical budget between recent context and retrieved context.

on renderings produced along the target camera trajectory. Mirage (Wang et al., 2026c) stores scene information in a latent 3D representation and retrieves spatial features for subsequent generation. These methods differ in their memory representations and update procedures, but all maintain a spatial state outside the generator's causal attention cache. Related approaches also introduce geometric supervision without maintaining an inference-time scene memory. For example, World-R1 (Wang et al., 2026b) uses 3D-aware reinforcement learning to improve geometric consistency during post-training.

Our method instead uses the generator's historical key-value states as memory. Each latent-frame KV entry is associated with its source camera, historical entries are ranked by pose affinity, and a learned policy divides the active context between recent and retrieved states.

3 Method

3.1 Overview

Given an input image, a text prompt, and a target camera trajectory, our model generates a video autoregressively as a sequence of latent chunks. During a revisit, earlier observations from similar viewpoints can provide scene details that are absent from the recent context. We therefore adaptively balance recent context and relevant history within a fixed active context budget.

We retain the key-value (KV) states of previously generated latent frames as historical memory. As illustrated in Figure 2, a decoupled camera-aware attention branch complements the pretrained visual pathway, with each branch maintaining its own KV cache. For each chunk, a learned policy divides a fixed historical KV budget between recent and retrieved context based on inter-chunk camera-pose changes and historical pose-affinity statistics. We select the allocated number of most recent frames and fill the remaining slots with frames from the remaining history. Both branches then attend to the corresponding KV states in their respective caches. This provides long-range scene references within the generator's native attention mechanism, without an explicit 3D scene representation.

3.2 Camera-Aware Attention and Pose-Affinity Retrieval

Adaptive allocation requires both a way to reuse past observations and a measure of their relevance to the current views. We first describe how camera-aware attention incorporates cached KV states, then introduce the pose-affinity score used to select frames for retrieval.

Decoupled camera-aware attention. To control the generated views along the target camera trajectory, we introduce an additional camera-aware branch using PRoPE (Li et al., 2025; Team et al., 2026) alongside the pretrained visual branch (Figure 2(a)). To reduce the added attention cost, we downsample each latent frame's feature map by fourfold average pooling along both height and width before the camera-aware QKV projections. After PRoPE attention and output projection, bilinear upsampling restores the original height and width, and the result is added to the base-branch output.

Following PRoPE's conjugate-attention formulation, the query and output side transformations are determined by the target camera, whereas the key and value transformations are determined by each historical frame's source camera.

We omit layer and attention-head indices. The superscripts base and cam identify the two branches, and a tilde marks states computed from the downsampled features. The queries $\mathbf{Q}_t$ come from the current chunk; keys and values include the selected past frames and the allowed current-chunk context. We write $\mathrm{Attn}_{\mathrm{PRoPE}}$ for camera-aware attention, including the PRoPE transformations and its output projection. For chunk t, the fused output is

$$\begin{aligned}\mathbf{O}_t = \; & \mathrm{Attn}\big(\mathbf{Q}_t^{\mathrm{base}}, \mathbf{K}^{\mathrm{base}}, \mathbf{V}^{\mathrm{base}}; \mathbf{A}_t^{\mathrm{base}}\big)\, \mathbf{W}_O^{\mathrm{base}} \\ & + \mathcal{U}_{\mathrm{bilinear}}\Big(\mathrm{Attn}_{\mathrm{PRoPE}}\Big(\widetilde{\mathbf{Q}}_t^{\mathrm{cam}}, \widetilde{\mathbf{K}}^{\mathrm{cam}}, \widetilde{\mathbf{V}}^{\mathrm{cam}}; \widetilde{\mathbf{A}}_t^{\mathrm{cam}}\Big)\Big),\end{aligned} \tag{1}$$

where $\mathcal{U}_{\mathrm{bilinear}}$ is the bilinear upsampling operator that restores the camera-aware output to the base branch's feature-map size and token sequence length, and $\mathbf{W}_O^{\mathrm{base}}$ is the base-branch output projection. Both branches use the same selected frames. For subsequent chunks, we additionally retain the first latent frame's KV tokens as attention sinks. The attention masks $\mathbf{A}_t^{\mathrm{base}}$ and $\widetilde{\mathbf{A}}_t^{\mathrm{cam}}$ allow access to these sink tokens, the selected recent and retrieved KV states, and the current-chunk tokens permitted by the causal mask, while blocking unselected history and future frames.

Pose-affinity retrieval. We associate each cached latent frame with its source camera's unit forward and up directions and camera center. For a query frame q and a historical frame i, the pose-affinity score $s(q, i)$ combines forward-direction agreement, up-direction agreement, and camera-center proximity. Historical frames receive higher scores when their camera directions align with those of the target view. The distance term assigns higher scores to historical frames whose camera centers are closer to the target camera.

Let Q_t denote the set of target latent-frame indices in the current chunk t. We define the retrieval score $s_t(i)$ of historical frame i as its highest pose affinity with any target frame in the chunk:

$$s_t(i) = \max_{q \in Q_t} s(q, i). \tag{2}$$

We rank historical frames by this score. The allocation policy determines how many frames to retrieve and how many recent frames to retain.

3.3 View-Adaptive Memory Allocation

Recent and retrieved context serve different needs. Retrieved frames with matching camera poses can recover scene details during revisits, whereas recent frames support smooth continuation when exploring regions with limited historical coverage. We therefore learn a view-adaptive policy that divides a fixed historical KV budget between recent and retrieved context based on inter-chunk camera-pose changes and historical pose-affinity (Figure 2(d)).

We summarize inter-chunk camera motion in g_t using the displacement between the mean camera centers of consecutive chunks and changes in their forward and up directions. The displacement is normalized by the median displacement of preceding chunk transitions. Since camera motion alone does not indicate whether relevant history exists, we also compute h_t, which contains the highest pose-affinity score among retrieval candidates and the mean score of the top-ranked candidates.

These candidates exclude the most recent frames that are retained regardless of the allocation. We compute h_t before predicting the retrieval size, so its candidate set does not depend on the allocation decision.

Let C be the historical capacity and $C_{\text{ret}} < C$ the maximum number of retrieved latent frames. A lightweight MLP f_θ predicts a probability for each integer retrieval size from 0 to C_{ret}:

$$\mathbf{p}_t = \text{softmax}(f_\theta([g_t; h_t])), \qquad k_t = \underset{k \in \{0,\dots,C_{\text{ret}}\}}{\arg\max}\ p_{t,k}. \tag{3}$$

Here, $p_{t,k}$ is the predicted probability of retrieving k frames, and $[g_t; h_t]$ concatenates the motion and retrieval-relevance features. The selected size k_t leaves $C - k_t$ slots for recent context.

Let $\mathcal{H}_t$ denote the available historical KV memory. We first retain the KV states of the $C - k_t$ most recent latent frames as $\mathcal{M}_t^{\text{rec}}$, then select the top k_t candidates from the remaining memory by pose affinity:

$$\mathcal{M}_t = \mathcal{M}_t^{\text{rec}} \ \cup\ \text{TopK}_{k_t}(\mathcal{H}_t \setminus \mathcal{M}_t^{\text{rec}}; s_t). \tag{4}$$

$\mathcal{M}_t$ is the active historical KV memory, combining recent and retrieved context without duplicates. Selection operates at latent-frame granularity: $s_t(i)$ ranks the KV states associated with frame i, and each selection includes all of that frame's KV tokens in both branches. Each branch reads the selected frames' KV states from its own cache.

3.4 Training and Inference

Training. Training proceeds in two stages. We first freeze the pretrained bidirectional Wan2.2-5B backbone and train the PRoPE branch for camera conditioning. We then jointly train the two attention branches and the memory-allocation policy autoregressively with teacher forcing, using chunks of three latent frames and the backbone's flow-matching objective. Appendix 6.4 describes how gradients propagate through the discrete allocation decision. Training uses a mixture of synthetic, gameplay, and real-world videos with per-frame camera annotations. Dataset statistics are reported in Appendix Table 4.

Inference. Before each chunk, we score the available history, predict the retrieval size, and assemble the active KV memory using Equations 3–4. After denoising, a clean forward pass appends the new branch-specific KV states and source-camera metadata to the historical memory. The first chunk bypasses retrieval, while subsequent chunks repeat the same selection and caching procedure.

4 Experiments

Our experiments assess general world-generation performance and the ability to recover previously observed scenes. We introduce the evaluation datasets and metrics in Section 4.1, then compare against video generation and explicit 3D methods on WorldScore before examining camera revisits in Section 4.2. Section 4.3 provides visual comparisons of scene exploration and return views, and Section 4.4 examines the memory components through ablations.

4.1 Datasets and Metrics

We evaluate on WorldScore (Duan et al., 2025) for general world generation and SRBench for scene recall.

WorldScore. We report WorldScore-Static, which assesses controllability, scene consistency, and visual quality, and WorldScore-Dynamic, which additionally evaluates motion accuracy, magnitude, and smoothness, along with their individual metrics.

SRBench. SRBench contains 200 camera-return cases, with 25 cases for each of eight motions: move left/right, pan left/right, push in/pull out, and orbit left/right. Each trajectory is followed in reverse to revisit earlier viewpoints. We evaluate revisit consistency (RCS) for visual similarity at revisited poses, margin relative to temporally matched and pose-separated control pairs, stability using adjacent-frame PSNR, and camera following for trajectory adherence. For fair cross-method

Table 1: Results on WorldScore, including static and dynamic aggregate scores and individual control, quality, consistency, and motion scores. Higher is better in every column. Bold and underlined entries indicate the best and second-best available scores, respectively, among all listed methods. Dashes denote unreported values.

Method	WorldScore Static↑	WorldScore Dynamic↑	Static Subset Details							Dynamic Subset Details		
			Camera Control	Object Control	Content Align.	3D Consist.	Photometric Consist.	Style Consist.	Subjective Quality	Motion Acc.	Motion Mag.	Motion Smooth.
VideoCrafter2 (Chen et al., 2024b)	52.57	47.49	28.92	39.07	72.46	65.14	61.85	43.79	56.74	47.12	30.40	29.39
EasyAnimate (Xu et al., 2024b)	52.85	51.65	26.72	54.50	50.76	67.29	47.35	73.05	50.31	75.00	31.16	40.32
Allegro (Zhou et al., 2024)	55.31	51.97	24.84	57.47	51.48	70.50	69.89	65.60	47.41	54.39	**40.28**	37.81
CogVideoX-I2V (Yang et al., 2024)	62.15	59.12	38.27	40.07	36.73	86.21	88.12	83.22	62.44	69.56	26.42	60.15
Vchitect-2.0 (Fan et al., 2025)	42.28	38.47	26.55	49.54	65.75	41.53	42.30	25.69	44.58	33.59	33.81	21.31
LTX-Video (HaCohen et al., 2024)	55.44	56.54	25.06	53.41	39.73	78.41	88.92	53.50	49.08	**76.22**	29.95	71.09
Wan2.1 (Wan et al., 2025)	57.56	52.85	23.53	40.32	45.44	78.74	78.36	77.18	59.38	54.27	33.26	38.05
WonderJourney (Yu et al., 2024)	63.75	44.63	84.60	37.10	35.54	80.60	79.03	62.82	66.56	-	-	-
InvisibleStitch (Engstler et al., 2025)	61.12	42.78	**93.20**	36.51	29.53	88.51	**89.19**	32.37	58.50	-	-	-
WonderWorld (Yu et al., 2025a)	72.69	50.88	92.98	51.76	71.25	86.87	85.56	70.57	49.81	-	-	-
Voyager (Huang et al., 2025)	**77.62**	54.53	85.95	66.92	68.92	81.56	85.99	84.89	**71.09**	-	-	-
Spatia (Zhao et al., 2026)	72.63	66.82	75.66	52.32	69.95	86.40	89.10	80.09	54.86	54.83	24.75	80.26
Ours	74.29	**68.63**	75.03	**72.50**	**74.48**	**93.40**	73.00	**85.07**	46.53	56.31	28.31	**81.69**

comparison, we cap the main evaluation at the common target horizon of 81 frames. Shorter outputs are evaluated on their available prefix without temporal padding. Additional longer rollouts of TraceWorld are presented in Appendix 6.6, and full metric definitions are provided in Appendix 6.5.

Implementation details. Our model generates videos at 1280×704 resolution and 16 fps, with three VAE latent timesteps per chunk. We set the active historical budget to $C = 12$ latent frames. For adaptive allocation, the retrieval limit is $C_{\text{ret}} = 6$, and the remaining $C - k_t$ slots are assigned to recent context. Each slot represents one latent timestep and contains its complete branch-specific spatial KV span. These counts refer to latent frames; the full stored history is distinct from the active budget.

4.2 Quantitative Results

Compared methods. We first evaluate on WorldScore against two groups of methods. General video generation models include VideoCrafter2 (Chen et al., 2024b), EasyAnimate (Xu et al., 2024b), Allegro (Zhou et al., 2024), CogVideoX-I2V (Yang et al., 2024), Vchitect-2.0 (Fan et al., 2025), LTX-Video (HaCohen et al., 2024), and Wan2.1 (Wan et al., 2025). Methods that build explicit 3D scene representations include WonderJourney (Yu et al., 2024), InvisibleStitch (Engstler et al., 2025), WonderWorld (Yu et al., 2025a), Voyager (Huang et al., 2025), and Spatia (Zhao et al., 2026). Comparing these groups tests how our approach, which uses pose-affinity retrieval of native attention states, performs alongside both general video models and systems with explicit geometry. Rankings use the available results in each table, and unreported values are shown as dashes.

Results on WorldScore. Table 1 places our method among the leading systems in broad world generation: it ranks first on WorldScore-Dynamic (68.63) and second on WorldScore-Static (74.29). Its strongest advantages are in object control and 3D consistency, where it exceeds the best competing scores by 5.58 and 4.89 points, respectively. The object control metric assesses whether the requested objects appear in the generated scene, while 3D consistency evaluates the stability of scene geometry across views. The joint result is consistent with improved adherence to scene specifications and more coherent structure under camera motion. It also supports native KV states as a useful basis for scene-consistent generation without constructing an explicit 3D scene representation.

Results on SRBench. For SRBench, we select Voyager and Spatia as the strongest competing methods on WorldScore-Static and WorldScore-Dynamic, respectively. We also include CogVideoX-I2V, the strongest general video baseline on both aggregates, and LTX-Video, which leads in motion accuracy. TraceWorld achieves the highest RCS and margin on SRBench (Table 2), outperforming both Voyager and Spatia. The joint improvement indicates better scene recovery that is specific to revisited poses, supporting the use of view-relevant history beyond recent context. TraceWorld ranks second in stability and camera following, behind Voyager, combining leading scene recall with competitive local smoothness and trajectory adherence. Appendix 6.6 provides qualitative examples of 484-frame rollouts.

Table 2: Results on the 200-case SRBench. Higher is better for all four scores. The best and second-best results are shown in **bold** and underlined, respectively.

Method	Revisit Consistency↑	Margin↑	Stability↑	Camera Following↑
CogVideoX-I2V (Yang et al., 2024)	0.5487	0.0307	0.6479	0.6371
LTX-Video (HaCohen et al., 2024)	0.4662	0.0846	0.5824	0.5814
Voyager (Huang et al., 2025)	0.6200	0.1549	**0.7660**	**0.9037**
Spatia (Zhao et al., 2026)	0.6882	0.1987	0.7181	0.8751
Ours	**0.6979**	**0.2086**	0.7258	0.8836

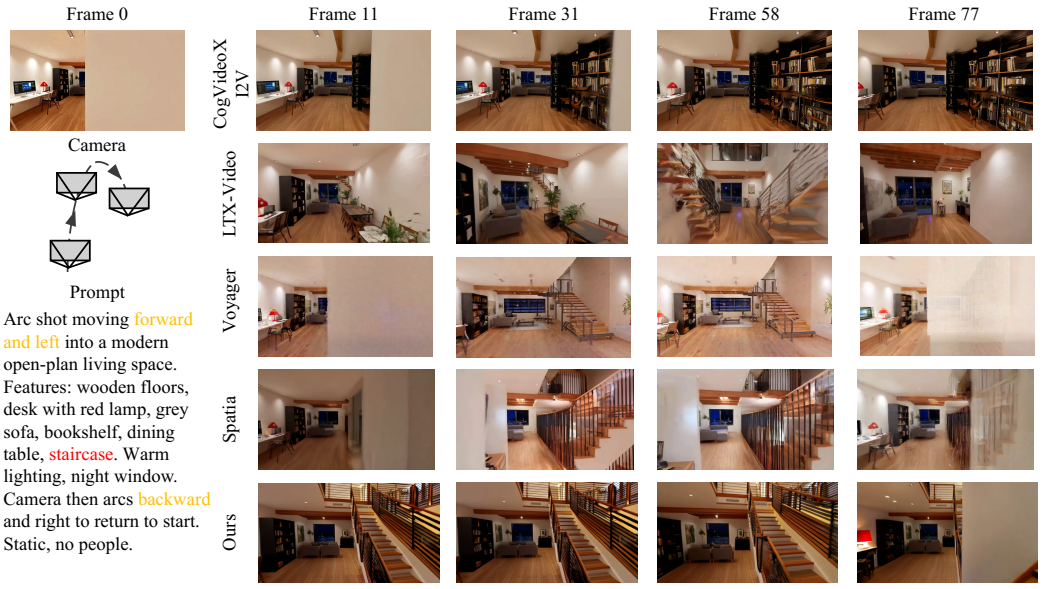

Figure 3: SRBench comparison on an indoor camera-return trajectory. The left column shows the initial frame, camera path, and prompt. The remaining columns show frames 11, 31, 58, and 77 from CogVideoX-I2V, LTX-Video, Voyager, Spatia, and TraceWorld. The camera moves forward and left into the room before returning toward the starting view, allowing comparison of scene layout, structural detail, and the final return view.

4.3 Qualitative Results

Figure 3 compares the five methods on a furnished indoor scene with a camera-return trajectory from SRBench. Additional comparisons are provided in Appendix 6.6. The prompt specifies a wooden floor, a desk with a red lamp, a grey sofa, a bookshelf, and a staircase. This example tests both the generation of newly revealed room structure and the preservation of recognizable scene content as the camera moves back toward its initial view.

CogVideoX-I2V (Yang et al., 2024) increasingly exposes a large bookshelf, but its final displayed frame remains an open view into the room rather than recovering the initial wall occlusion. LTX-Video (HaCohen et al., 2024) exhibits pronounced structural distortion around the staircase and walls at frame 58. Voyager (Huang et al., 2025) and Spatia (Zhao et al., 2026) show more recognizable stair structures, although their final frames contain blurred or ghosted boundaries near the foreground wall and railing. In the TraceWorld sequence, the staircase and dark railings remain clearly recognizable across the intermediate views, alongside the sofa and rear window.

At frame 77, TraceWorld brings the foreground wall back into view while retaining the red lamp, bookshelf, sofa, and rear window as recognizable scene landmarks. This behavior illustrates scene consistency across the prescribed viewpoint changes. The returned image is not an exact reproduction of frame 0: its framing, illumination, and some local details differ. We therefore interpret this

Table 3: Memory ablations under the same 12-frame active history for all memory-enabled variants. Higher is better; **bold** and underlined indicate the best and second-best scores.

Configuration	Revisit Consistency↑	Margin↑	Stability↑	Camera Following↑
Baseline	0.4951	0.0910	0.5925	0.8209
Baseline + Recent KV	0.5141	0.0988	0.6610	0.8210
Baseline + FOV-overlap retrieval	0.5526	0.1269	0.6268	0.8252
Baseline + Pose-affinity retrieval	0.5957	0.1245	0.6486	0.8170
Baseline + FOV-overlap retrieval + Adaptive allocation	0.6380	<u>0.2013</u>	<u>0.7227</u>	0.8444
Baseline + Pose-affinity retrieval + Fixed allocation	<u>0.6763</u>	0.1842	0.7097	<u>0.8652</u>
Baseline + Pose-affinity retrieval + Adaptive allocation	**0.6979**	**0.2086**	**0.7258**	**0.8836**

example as qualitative evidence of scene recall, consistent with the RCS and margin results, rather than as evidence of pixel-perfect reconstruction or precise camera alignment.

4.4 Ablation Studies

We compare recent-only, FOV-overlap, and pose-affinity KV selection under the same 12-frame history budget. Our baseline is Wan2.2-5B trained with teacher forcing and the same decoupled PRoPE branch, but without cross-chunk KV caching. Inspired by the camera-overlap retrieval strategy in CaM (Yu et al., 2025b), we construct an FOV-overlap baseline within our framework. In the recent-only and retrieval-only variants, all 12 history slots are filled with KV states from the most recent frames or from frames selected by FOV overlap or pose affinity, respectively. The fixed-allocation variant uses six recent and six retrieved frames. In the adaptive variants, our proposed allocation policy dynamically selects up to six retrieved frames and fills the remaining slots with recent frames.

KV selection strategies. We first compare recent KV, FOV-overlap retrieval, and pose-affinity retrieval without mixing recent and retrieved context. Recent KV selects history solely by temporal proximity. Among the three strategies, it has the lowest revisit consistency and margin but the highest stability: relying on recent consecutive frames supports local continuity while providing limited access to scene references needed at revisits. Compared with FOV-overlap retrieval, pose-affinity retrieval increases revisit consistency from 0.5526 to 0.5957 and stability from 0.6268 to 0.6486, while margin changes from 0.1269 to 0.1245 and camera following from 0.8252 to 0.8170. Thus, pose-based ranking improves scene recall and temporal stability without adaptive allocation, although it does not improve all metrics.

Adaptive allocation. We next separate the effect of adaptive allocation from that of simply mixing recent and retrieved context. Fixed 6+6 pose-affinity allocation improves all four metrics over recent-only and retrieval-only selection, confirming their complementarity. Adaptive allocation further raises revisit consistency from 0.6763 to 0.6979 and margin from 0.1842 to 0.2086, while also improving stability and camera following. Similar gains with FOV-overlap retrieval show that adaptive balancing benefits both retrieval strategies, with pose affinity and adaptive allocation achieving the best overall results.

5 Conclusion and Limitations

We presented TraceWorld, a video world model that uses view-adaptive KV memory allocation to maintain scene consistency across changing viewpoints. A learned policy divides a fixed historical context budget between recent and retrieved KV states according to camera motion and historical relevance. Pose-affinity retrieval and camera-aware attention support this allocation by selecting and integrating useful scene references without constructing an explicit 3D scene memory. Our experiments demonstrate strong geometric consistency and scene recall, while comparisons with fixed allocation show the benefit of adapting the recent/retrieved balance.

The current framework still has limitations. The active attention context is bounded, but cumulative KV storage and retrieval scoring grow with rollout length. The method also assumes reliable camera poses, and evaluation primarily covers static or quasi-static scenes. Extending it to substantial scene dynamics and noisy or unavailable camera poses remains future work.

AI DISCLOSURE

In this work, we used generative AI tools to assist with language editing and polishing. All AI-assisted content was carefully reviewed and revised by the authors, who take full responsibility for the accuracy, originality, and scientific integrity of the final manuscript.

ETHICS STATEMENT

TraceWorld studies camera-controlled video generation with improved scene consistency across revisited viewpoints. Potential applications include interactive content creation and virtual environment exploration. Like other generative video models, it may inherit biases from its pretrained backbone and training data, and could be misused to create misleading visual content. Improved scene consistency does not guarantee factual accuracy or physical correctness, and generated videos should not be treated as evidence of real events. Responsible use should include clear disclosure of generated content and respect for privacy, intellectual property, and the usage terms of source data and models.

REPRODUCIBILITY STATEMENT

Section 3 describes the model architecture, pose-affinity retrieval, view-adaptive memory allocation, and training and inference procedures. Appendices 6.2, 6.3, and 6.4 provide implementation details of camera-aware attention, retrieval scores and weights, allocation descriptors, and gradient propagation through the discrete allocation decision. Training data sources and statistics are reported in Table 4, and the benchmark evaluation metrics are detailed in Appendix 6.5. These descriptions specify the main components and evaluation procedures needed to reproduce our experiments.

REFERENCES

Sherwin Bahmani, Ivan Skorokhodov, Guocheng Qian, Aliaksandr Siarohin, Willi Menapace, Andrea Tagliasacchi, David B. Lindell, and Sergey Tulyakov. Ac3d: Analyzing and improving 3d camera control in video diffusion transformers, 2025. URL https://arxiv.org/abs/2411.18673.

Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. Recammaster: Camera-controlled generative rendering from a single video. In *International Conference on Computer Vision*, pp. 14834–14844. IEEE, 2025.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

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, 2024a.

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.

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7310–7320. IEEE, 2024b.

Haoyi Duan, Hong-Xing Yu, Sirui Chen, Li Fei-Fei, and Jiajun Wu. Worldscore: A unified evaluation benchmark for world generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 27713–27724, 2025.

Paul Engstler, Andrea Vedaldi, Iro Laina, and Christian Rupprecht. Invisible stitch: Generating smooth 3d scenes with depth inpainting. In *International Conference on 3D Vision*, pp. 457–468. IEEE, 2025.

Weichen Fan, Chenyang Si, Junhao Song, Zhenyu Yang, Yinan He, Long Zhuo, Ziqi Huang, Ziyue Dong, Jingwen He, Dongwei Pan, et al. Vchitect-2.0: Parallel transformer for scaling up video diffusion models. *arXiv preprint arXiv:2501.08453*, 2025.

Kaifeng Gao, Jiaxin Shi, Hanwang Zhang, Chunping Wang, and Jun Xiao. Vid-gpt: Introducing gpt-style autoregressive generation in video diffusion models. *arXiv preprint arXiv:2406.10981*, 2024.

Google DeepMind. Veo 3 video model. `https://deepmind.google/models/veo/`, 2026.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation, 2024.

Hao He, Ceyuan Yang, Shanchuan Lin, Yinghao Xu, Meng Wei, Liangke Gui, Qi Zhao, Gordon Wetzstein, Lu Jiang, and Hongsheng Li. Cameractrl ii: Dynamic scene exploration via camera-controlled video diffusion models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13416–13426. IEEE, 2025.

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

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*, 2025.

Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. In *International Conference on Learning Representations*, volume 2025, pp. 23378–23402, 2025.

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

Kuaishou. Kling video model. `https://kling.kuaishou.com`, 2026.

Ruilong Li, Brent Yi, Junchen Liu, Hang Gao, Yi Ma, and Angjoo Kanazawa. Cameras as relative positional encoding. *Advances in Neural Information Processing Systems*, 2025.

Zhen Li, Chuanhao Li, Xiaofeng Mao, Shaoheng Lin, Ming Li, Shitian Zhao, Zhaopan Xu, Xinyue Li, Yukang Feng, Jianwen Sun, et al. Sekai: A video dataset towards world exploration. *Advances in Neural Information Processing Systems*, 38, 2026.

Jinlai Liu, Jian Han, Bin Yan, Hui Wu, Fengda Zhu, Xing Wang, Yi Jiang, Bingyue Peng, and Zehuan Yuan. Infinitystar: Unified spacetime autoregressive modeling for visual generation. *Advances in Neural Information Processing Systems*, 38:170054–170072, 2026.

Minimax. Minimax h3: An open model breaking the boundaries between tasks and modalities. `https://www.minimax.io/blog/minimax-h3`, 2026.

OpenAI. Sora 2 video model. `https://openai.com/research/sora-2`, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026.

DreamX Team, Yancheng Bai, Rui Chen, Xiangxiang Chu, Rujing Dang, Hao Dou, Bingjie Gao, Qiwen Gu, Siyu Hong, Jiachen Lei, et al. Dreamx-world 1.0: A general-purpose interactive world model. *arXiv preprint arXiv:2606.16993*, 2026.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *International Conference on Learning Representations (ICLR)*, volume 2025, pp. 73754–73776, 2025.

Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. *European Conference on Computer Vision*, pp. 313–331, 2024.

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.

Jiahao Wang, Yufeng Yuan, Rujie Zheng, Youtian Lin, Jian Gao, Lin-Zhuo Chen, Yajie Bao, Chang Zeng, Yanxi Zhou, Xiao-Xiao Long, et al. Spatialvid: A large-scale video dataset with spatial annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42592–42603, 2026a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Weijie Wang, Xiaoxuan He, Youping Gu, Yifan Yang, Zeyu Zhang, Yefei He, Yanbo Ding, Xirui Hu, Donny Y Chen, Zhiyuan He, et al. World-r1: Reinforcing 3d constraints for text-to-video generation. *arXiv preprint arXiv:2604.24764*, 2026b.

Weijie Wang, Haoyu Zhao, Yifan Yang, Feng Chen, Zeyu Zhang, Yefei He, Zicheng Duan, Donny Y. Chen, Yuqing Yang, and Bohan Zhuang. Latent spatial memory for video world models. *arXiv preprint arXiv:2606.09828*, 2026c.

Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.

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.

Chendong Xiang, Jiajun Liu, Jintao Zhang, Xiao Yang, Zhengwei Fang, Shizun Wang, Zijun Wang, Yingtian Zou, Hang Su, and Jun Zhu. Geometry-aware rotary position embedding for consistent video world model, 2026. URL `https://arxiv.org/abs/2602.07854`.

Dejia Xu, Yifan Jiang, Chen Huang, Liangchen Song, Thorsten Gernoth, Liangliang Cao, Zhangyang Wang, and Hao Tang. Cavia: Camera-controllable multi-view video diffusion with view-integrated attention. *arXiv preprint arXiv:2410.10774*, 2024a.

Jiaqi Xu, Xinyi Zou, Kunzhe Huang, Yunkuo Chen, Bo Liu, MengLi Cheng, Xing Shi, and Jun Huang. Easyanimate: A high-performance long video generation method based on transformer architecture. *arXiv preprint arXiv:2405.18991*, 2024b.

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

Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6658–6667. IEEE, 2024.

Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5916–5926. IEEE, 2025a.

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. pp. 1–11, 2025b.

Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models. *Advances in Neural Information Processing Systems*, 38:30546–30566, 2026.

Jinjing Zhao, Fangyun Wei, Zhening Liu, Hongyang Zhang, Chang Xu, and Yan Lu. Spatia: Video generation with updatable spatial memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4245–4257, 2026.

Guangcong Zheng, Teng Li, Rui Jiang, Yehao Lu, Tao Wu, and Xi Li. Cami2v: Camera-controlled image-to-video diffusion model. *arXiv preprint arXiv:2410.15957*, 2024a.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024b.

Jensen Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. In *2025 IEEE/CVF International Conference on Computer Vision*, pp. 12405–12414. IEEE, 2025a.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

Yang Zhou, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Haoyu Guo, Zizun Li, Kaijing Ma, Xinyue Li, Yating Wang, Haoyi Zhu, et al. Omniworld: A multi-domain and multi-modal dataset for 4d world modeling. *arXiv preprint arXiv:2509.12201*, 2025b.

Yuan Zhou, Qiuyue Wang, Yuxuan Cai, and Huan Yang. Allegro: Open the black box of commercial-level video generation model. *arXiv preprint arXiv:2410.15458*, 2024.

6 APPENDIX

6.1 TRAINING DATA DETAILS

Our training corpus combines synthetic scenes, gameplay videos, and real-world footage to cover complementary camera-motion and appearance regimes. Synthetic clips provide controlled trajectories, gameplay videos contribute navigation through varied virtual environments, and real-world clips broaden the range of scene content, lighting, and capture conditions. Each clip includes per-frame camera extrinsics and intrinsics, which are used both to condition the PRoPE branch and to index the historical KV memory. The corpus contains 408,660 clips and 1,230.2 hours in total; clips range from 5 to 17 seconds at resolutions between 720p and 1080p. We train both stage on 16 h20 gpus.

Table 4: Training data statistics. All clips are annotated with per-frame camera extrinsics and intrinsics.

Source	Dataset	Clips	Hours	Avg. (s)
Synthetic	UE-Data	66,528	92.4	5.0
Gameplay	Sekai-Walking (Li et al., 2026)	12,213	17.0	5.0
	Sekai-Drone (Li et al., 2026)	1,570	2.2	5.0
	OmniWorld (Zhou et al., 2025b)	4,718	16.5	12.6
Real-world	RealEstate (Zhou et al., 2018)	26,401	55.9	7.6
	SpatialVidHQ (Wang et al., 2026a)	297,230	1,046.2	12.7
Total		**408,660**	**1,230.2**	

6.2 IMPLEMENTATION DETAILS OF CAMERA-AWARE ATTENTION

Spatial resolution. Let $\mathbf{X}_l$ denote the input visual features at layer l, with spatial size $H_l \times W_l$. The base branch operates at this native resolution. Before the camera-aware QKV projections, we apply average pooling independently to each latent frame, reducing both spatial dimensions by a factor of four while leaving the temporal dimension unchanged. The camera-aware QKV projections, PRoPE transformations, and FlexAttention therefore operate on a grid of size $(H_l/4) \times (W_l/4)$.

Let $\widetilde{\mathbf{O}}_l^{\mathrm{cam}}$ denote the camera-aware output on this reduced grid, after PRoPE attention and output projection. We restore its native spatial resolution using bilinear interpolation:

$$\mathbf{O}_l^{\mathrm{cam}} = \mathcal{U}_{\mathrm{bilinear}}\left(\widetilde{\mathbf{O}}_l^{\mathrm{cam}}; H_l, W_l\right). \tag{5}$$

Interpolation is applied independently to each latent frame and feature channel. The resulting output is added to the base-branch output at the same spatial resolution.

Camera parameters on the reduced grid. Spatial downsampling changes the visual feature grid but not the camera parameters. All camera-aware tokens belonging to a latent frame share that frame's camera pose. We reuse the original frame pose and fixed normalized intrinsics when applying PRoPE, with spatial token locations corresponding to the reduced grid. Camera poses are neither pooled nor interpolated.

Consequently, the source-camera projective operator is shared across the spatial tokens of a source frame. Keys and values use the camera parameters of their source frames, while queries use those of their target frames.

Query-independent PRoPE caching. We use the conjugate-attention formulation of PRoPE. In the following equations, we omit layer and attention-head indices. The symbols $\mathbf{Q}_i$, $\mathbf{K}_j$, and $\mathbf{V}_j$ denote camera-aware query, key, and value features belonging to latent frames i and j on the reduced spatial grid. We omit their tilde and cam superscript for readability. A hat denotes features after a PRoPE transformation.

Let $\mathbf{P}_i$ and $\mathbf{P}_j$ denote the projective operators constructed from the target camera of frame i and the source camera of frame j, respectively. PRoPE transforms the features as

$$\begin{aligned}\widehat{\mathbf{Q}}_i &= \mathrm{apply_fn}_q\left(\mathbf{Q}_i; \mathbf{P}_i^\top\right),\\ \widehat{\mathbf{K}}_j &= \mathrm{apply_fn}_{kv}\left(\mathbf{K}_j; \mathbf{P}_j^{-1}\right),\\ \widehat{\mathbf{V}}_j &= \mathrm{apply_fn}_{kv}\left(\mathbf{V}_j; \mathbf{P}_j^{-1}\right).\end{aligned} \tag{6}$$

Here, $\mathrm{apply_fn}_q$ and $\mathrm{apply_fn}_{kv}$ denote PRoPE's feature transformations using the supplied query-side and source-side projective operators. The corresponding output transformation $\mathrm{apply_fn}_o$ maps the attention result back using the target-camera operator.

Let $\widehat{\mathbf{Q}}_t$ collect the transformed queries of the current chunk, and let $\widehat{\mathbf{K}}$ and $\widehat{\mathbf{V}}$ collect the transformed keys and values supplied to attention. These include historical, sink, and current-chunk states, with visibility controlled by the camera-branch attention mask. The output is computed as

$$\begin{aligned}\widehat{\mathbf{O}}_t &= \mathrm{Attn}\left(\widehat{\mathbf{Q}}_t, \widehat{\mathbf{K}}, \widehat{\mathbf{V}}; \widetilde{\mathbf{A}}_t^{\mathrm{cam}}\right)\mathbf{W}_O^{\mathrm{cam}},\\ \widetilde{\mathbf{O}}_i^{\mathrm{cam}} &= \mathrm{apply_fn}_o\left(\widehat{\mathbf{O}}_i; \mathbf{P}_i\right).\end{aligned} \tag{7}$$

Here, $\mathbf{W}_O^{\mathrm{cam}}$ is the camera-branch output projection, and $\widehat{\mathbf{O}}_i$ contains the output tokens of frame i within $\widehat{\mathbf{O}}_t$. The output transformation is applied separately using each query frame's camera. The resulting $\widetilde{\mathbf{O}}_i^{\mathrm{cam}}$ remains on the reduced grid and is subsequently upsampled for fusion with the base branch. These operations expand the $\mathrm{Attn}_{\mathrm{PRoPE}}$ operator in Equation 1.

The camera parameters used to transform historical keys and values depend only on their source frames, not on future queries. The transformed states $\widehat{\mathbf{K}}_j$ and $\widehat{\mathbf{V}}_j$ are therefore written directly to the camera-aware KV cache. When frame j is selected later, its cached states can be reused without a new transformation for the current camera. Only the current query and output transformations depend on the target camera.

Unlike conventional rotary positional encodings that transform queries and keys, this PRoPE formulation also transforms values using the source-camera operator. This source-dependent value transformation remains compatible with KV caching.

Branch-specific masks and KV states. The two branches use the same frame-level memory selection and causal attention rules. These rules are expanded over their respective spatial grids, giving the native-resolution mask $\mathbf{A}_t^{\mathrm{base}}$ and the reduced-resolution mask $\widetilde{\mathbf{A}}_t^{\mathrm{cam}}$. Their dimensions differ, but their effective visibility includes the same selected historical frames, sink frames, and causally permitted current frames. Appendix 6.4 describes how causal masks and allocation gates implement this selection during training.

Each historical memory slot represents one latent frame. For a native grid of size $H_l \times W_l$, the base cache stores $H_l W_l$ spatial KV states per frame, while the camera-aware cache stores $(H_l/4)(W_l/4)$ states. Both branches select the same latent-frame indices and read the complete spatial KV states of each selected frame from their own caches.

6.3 Implementation Details of Pose-Affinity Retrieval

Camera descriptors and pairwise score. Let Q_t denote the target latent-frame indices in chunk t. We write $\mathcal{I}_t = \mathrm{ids}(\mathcal{H}_t)$ for the indices associated with the available historical KV memory, where ids returns the source-frame indices of a KV collection. Throughout these appendices, $\mathcal{H}_t$ and $\mathcal{I}_t$ refer to history eligible for budget allocation; the first latent frame is retained separately as an attention sink and is excluded from these collections.

For each latent frame i, we extract its camera center $\mathbf{o}_i$ from the translation column of its camera-to-world matrix. The normalized third and second columns of the rotation block give its forward direction $\mathbf{f}_i$ and up direction $\mathbf{u}_i$, respectively, following the implementation's camera-axis convention.

For target frame $q \in Q_t$ and historical frame $i \in \mathcal{I}_t$, the pose-affinity score is

$$s(q,i) = \lambda_f \langle \mathbf{f}_q, \mathbf{f}_i \rangle + \lambda_u \langle \mathbf{u}_q, \mathbf{u}_i \rangle + \lambda_p \exp\left(-\frac{\|\mathbf{o}_q - \mathbf{o}_i\|_2}{\sigma_t} \right). \tag{8}$$

Here, λ_f, λ_u, and λ_p weight forward-direction agreement, up-direction agreement, and camera-center proximity, respectively. We set $\lambda_f = 1.0$ and $\lambda_u = \lambda_p = 0.2$ in all experiments.

Distance scale. The distance scale σ_t is the median of camera-center distances $\|\mathbf{o}_q - \mathbf{o}_i\|_2$ greater than 10^{-6} over $q \in Q_t$ and $i \in \mathcal{I}_t$. If no distance exceeds this threshold, we use $\sigma_t = 1$. This scale normalizes camera-center proximity relative to the distances between the current target views and available history.

Chunk aggregation and selection. Equation 2 defines the retrieval score of historical frame i as $s_t(i) = \max_{q \in Q_t} s(q,i)$. A historical frame therefore receives a high score when it matches at least one target view in the current chunk.

After the allocation policy determines the retrieval size, we retain the allocated number of recent frames and exclude them from retrieval. The first latent frame is retained separately as an attention sink and does not participate in this ranking. We select the highest-scoring frames from the remaining history and combine them with the recent frames. Frame indices are deduplicated and sorted in temporal order before their KV states are gathered. When no historical frames are available, retrieval is skipped.

KV retrieval granularity. Each selected frame corresponds to one VAE latent timestep. Its index retrieves the complete native-resolution spatial KV states from the base cache and the complete reduced-resolution spatial KV states from the camera-aware cache. The branches therefore use the same selected frame indices while reading their own KV representations.

6.4 Implementation Details of View-Adaptive Allocation

Notation. We use the camera descriptors and historical index set defined in Appendix 6.3. The historical capacity is C latent-frame units, and $0 < C_{\text{ret}} < C$ is the maximum number assigned to pose-affinity retrieval. Sink tokens and current-chunk tokens are handled separately and do not count toward C.

The allocation procedure below applies when $|\mathcal{I}_t| > C$. When $|\mathcal{I}_t| \le C$, all available historical states are used and allocation is bypassed.

View-change descriptor. For chunk t, let $\bar{\mathbf{o}}_t$ denote its mean camera center, and let $\bar{\mathbf{f}}_t$ and $\bar{\mathbf{u}}_t$ denote its normalized mean forward and up directions, respectively:

$$\begin{aligned} \bar{\mathbf{o}}_t &= \frac{1}{|Q_t|} \sum_{q \in Q_t} \mathbf{o}_q, \\ \bar{\mathbf{f}}_t &= \mathrm{norm}_\epsilon \left(\sum_{q \in Q_t} \mathbf{f}_q \right), \\ \bar{\mathbf{u}}_t &= \mathrm{norm}_\epsilon \left(\sum_{q \in Q_t} \mathbf{u}_q \right). \end{aligned} \tag{9}$$

Here, $\mathrm{norm}_\epsilon(\mathbf{x}) = \mathbf{x} / \max(\|\mathbf{x}\|_2, \epsilon)$, with $\epsilon = 10^{-6}$ used for numerical stability throughout this subsection. If a directional sum has norm below ϵ, we instead use the corresponding unit direction of the chunk's first latent frame.

For $t \ge 1$, let Δ_t denote the displacement between the mean camera centers of consecutive chunks:

$$\Delta_t = \|\bar{\mathbf{o}}_t - \bar{\mathbf{o}}_{t-1}\|_2. \tag{10}$$

Let $\widetilde{\Delta}_t$ denote the reference displacement scale and d_t the normalized displacement. We compute the reference scale from preceding chunk transitions, using Δ_1 at the first transition:

$$\begin{aligned}\widetilde{\Delta}_t &= \begin{cases}\Delta_1, & t = 1,\\ \mathrm{median}_{1\le j<t}\,\Delta_j, & t \ge 2,\end{cases}\\ d_t &= \frac{\Delta_t}{\max(\widetilde{\Delta}_t, \epsilon)}.\end{aligned} \tag{11}$$

The initialization avoids an empty reference set, and the denominator floor handles negligible preceding translation.

Let θ_t^f and θ_t^u denote the inter-chunk angular changes in the forward and up directions, respectively, normalized by π. Using $\mathrm{clip}(x, a, b) = \min(b, \max(a, x))$, we compute

$$\begin{aligned}\theta_t^f &= \frac{1}{\pi}\arccos\left(\mathrm{clip}(\bar{\mathbf{f}}_t^\top \bar{\mathbf{f}}_{t-1}, -1, 1)\right),\\ \theta_t^u &= \frac{1}{\pi}\arccos\left(\mathrm{clip}(\bar{\mathbf{u}}_t^\top \bar{\mathbf{u}}_{t-1}, -1, 1)\right).\end{aligned} \tag{12}$$

The local view-change descriptor is

$$g_t = [d_t, \theta_t^f, \theta_t^u]^\top. \tag{13}$$

It describes local camera motion but does not indicate whether matching historical views are available.

Retrieval-relevance descriptor. We compute retrieval relevance before predicting the retrieval size. Let $\mathcal{I}_t^{\mathrm{cand}}$ denote the historical indices in $\mathcal{I}_t$ excluding the $C - C_{\mathrm{ret}}$ most recent latent frames. These recent frames are retained under every allowed allocation, so this candidate set is independent of the predicted k_t.

The retrieval-relevance descriptor h_t contains the maximum candidate score and the mean score of the C_{ret} highest-ranked candidates:

$$h_t = \begin{bmatrix} \max\limits_{i\in\mathcal{I}_t^{\mathrm{cand}}} s_t(i) \\ \dfrac{1}{C_{\mathrm{ret}}} \displaystyle\sum_{i\in \mathrm{TopK}_{C_{\mathrm{ret}}}(\mathcal{I}_t^{\mathrm{cand}}; s_t)} s_t(i) \end{bmatrix}. \tag{14}$$

Here, TopK returns the indices of the specified number of highest-scoring candidates. Because allocation is used only when $|\mathcal{I}_t| > C$, the candidate set contains at least C_{ret} frames.

The maximum captures the strongest historical match, while the top-ranked mean summarizes relevance across several frames. These are geometric affinity scores, not calibrated probabilities of scene overlap. Together with g_t, they distinguish similar local camera motions with different historical matches.

Retrieval-size prediction. Let $\mathcal{K} = \{0, \ldots, C_{\mathrm{ret}}\}$ denote the possible retrieval sizes. The allocation predictor f_θ is a lightweight MLP with learnable parameters θ. It takes the concatenated features $[g_t; h_t]$ and predicts a probability for each size:

$$\begin{aligned}\mathbf{p}_t &= \mathrm{softmax}\left(f_\theta([g_t; h_t])\right),\\ k_t &= \arg\max_{k\in\mathcal{K}} p_{t,k}.\end{aligned} \tag{15}$$

Here, $p_{t,k}$ is the entry of $\mathbf{p}_t$ corresponding to retrieval size k. The selected size k_t assigns k_t slots to retrieved frames and $C - k_t$ slots to recent frames. The predictor is trained through the generation objective, without manually labeled retrieval sizes.

Candidate memory plans. For each candidate size $k \in \mathcal{K}$, let $\mathcal{M}_t^{\mathrm{rec}}(k)$ contain the KV states of the $C - k$ most recent historical latent frames. The corresponding active memory plan is

$$\mathcal{M}_t(k) = \mathcal{M}_t^{\mathrm{rec}}(k) \cup \mathrm{TopK}_k\left(\mathcal{H}_t \setminus \mathcal{M}_t^{\mathrm{rec}}(k); s_t\right). \tag{16}$$

When applied to a KV collection, TopK ranks source frames by $s_t(i)$ and returns their complete KV states in both branches. Selection, union, and exclusion operate by source-frame identity. For $k = 0$, the retrieval result is empty. Each plan therefore contains exactly C distinct historical latent-frame units.

The selected memory in the main text is $\mathcal{M}_t = \mathcal{M}_t(k_t)$, with $\mathcal{M}_t^{\mathrm{rec}} = \mathcal{M}_t^{\mathrm{rec}}(k_t)$. The separately retained sink and current-chunk states are not part of these budgeted memory plans.

Straight-through training. To train the discrete allocation predictor, we associate each candidate plan with a binary frame-membership mask:

$$m_{t,i}^{(k)} = \mathbb{I}\left[i \in \mathrm{ids}(\mathcal{M}_t(k))\right]. \tag{17}$$

Here, i indexes a historical latent frame, and $\mathbb{I}[\cdot]$ equals one when its condition holds and zero otherwise. The soft, hard, and straight-through masks are

$$\begin{aligned} m_{t,i}^{\mathrm{soft}} &= \sum_{k \in \mathcal{K}} p_{t,k} m_{t,i}^{(k)}, \\ m_{t,i}^{\mathrm{hard}} &= m_{t,i}^{(k_t)}, \\ m_{t,i}^{\mathrm{ST}} &= \mathrm{sg}\left(m_{t,i}^{\mathrm{hard}} - m_{t,i}^{\mathrm{soft}}\right) + m_{t,i}^{\mathrm{soft}}, \end{aligned} \tag{18}$$

where sg denotes stop-gradient. The forward pass uses the hard selection, while the backward pass propagates surrogate gradients through the soft mask.

During training, we gather the union of the candidate plans, $\bigcup_{k \in \mathcal{K}} \mathcal{M}_t(k)$. Each frame-level mask is broadcast to all KV tokens belonging to that frame in each branch. This allows alternative plans to contribute allocation gradients without changing the hard forward selection.

Gated attention. We apply the allocation mask to attention weights. Let z_{rj}^b denote the attention logit between query token r and key token j in branch $b \in \{\mathrm{base}, \mathrm{cam}\}$. Token indices are local to each branch's spatial grid.

Let $A_{t,rj}^b$ denote the additive causal mask entry: it is zero for causally permitted query–key pairs and $-\infty$ otherwise. During allocation training, this mask does not remove candidate-history tokens solely because they are absent from the hard-selected plan. Allocation selection is instead applied through a separate gate.

Let $i(j)$ denote the source latent frame of historical key token j, and let $\gamma_{t,j}$ denote its allocation gate. For historical tokens, $\gamma_{t,j} = m_{t,i(j)}^{\mathrm{ST}}$. For sink and current-chunk tokens, $\gamma_{t,j} = 1$, with visibility determined by the causal mask. The attention weights are

$$\alpha_{rj}^b = \frac{\gamma_{t,j} \exp(z_{rj}^b + A_{t,rj}^b)}{\sum_{j'} \gamma_{t,j'} \exp(z_{rj'}^b + A_{t,rj'}^b)}. \tag{19}$$

Here, α_{rj}^b is the normalized weight assigned to key j by query r, and the denominator sums over all supplied keys in branch b, including candidate-history, sink, and current-chunk tokens.

The causal mask and allocation gate jointly implement the effective visibility described by $\mathbf{A}_t^{\mathrm{base}}$ and $\widetilde{\mathbf{A}}_t^{\mathrm{cam}}$ in the main text. In the forward pass, attention uses only the selected historical plan and causally permitted sink and current-chunk tokens. In the backward pass, the soft allocation path allows the generation loss to update f_θ through $\mathbf{p}_t$.

Inference. At inference, we construct only the selected historical memory $\mathcal{M}_t = \mathcal{M}_t(k_t)$, and each branch reads its corresponding KV states. The candidate union and straight-through masks are unnecessary. The first chunk has no historical context and bypasses allocation; subsequent chunks also bypass allocation while the available history fits within C. The fixed budget bounds the selected historical context used by attention, while the stored KV history remains available for subsequent retrieval.

6.5 SRBENCH METRICS

We report Revisit Consistency (RCS), Margin, Stability, and Camera Following separately, without an overall aggregate score. These metrics assess scene recall, revisit specificity, short-term temporal smoothness, and adherence to the prescribed camera trajectory, respectively.

Visual-metric preprocessing. RGB frames are converted to floating-point values in $[0, 1]$ and resized to 352×640 using bicubic interpolation for LPIPS, PSNR, and SSIM.

For DINO similarity, these frames are further resized to 518×518 using bicubic interpolation and normalized using the model-specific mean and standard deviation. We use DINOv2 ViT-S/14, loaded through `timm` with pretrained weights and `num_classes=0`. The resulting pooled CLS feature is L2-normalized before computing pairwise similarity.

Revisit-pair construction. Revisit annotations are constructed from the full prescribed 81-frame camera trajectory, independently of generated-video content or length. Let c index a benchmark case and t denote the turning-frame index of its trajectory. For each post-turning frame $i \in \{t + 1, \ldots, 80\}$, we search pre-turning candidates $j \in \{0, \ldots, t - \Delta_{\min}\}$, with $\Delta_{\min} = 6$.

Let $d_{\mathrm{rot}}(i, j)$ denote the rotation geodesic distance in degrees between prescribed camera poses i and j, and let $d_{\mathrm{trans}}(i, j)$ denote their Euclidean camera-center distance. The full-trajectory extents are

$$A_{\mathrm{rot}} = \max_{0 \le k \le 80} d_{\mathrm{rot}}(0, k), \qquad A_{\mathrm{trans}} = \max_{0 \le k \le 80} d_{\mathrm{trans}}(0, k). \tag{20}$$

We use the safe normalization denominators

$$\begin{aligned} \bar{A}_{\mathrm{rot}} &= \begin{cases} A_{\mathrm{rot}}, & A_{\mathrm{rot}} > 0, \\ 1.0, & A_{\mathrm{rot}} = 0, \end{cases} \\ \bar{A}_{\mathrm{trans}} &= \begin{cases} A_{\mathrm{trans}}, & A_{\mathrm{trans}} > 0, \\ 10^{-8}, & A_{\mathrm{trans}} = 0. \end{cases} \end{aligned} \tag{21}$$

The distance used to identify the closest revisit candidate is

$$d(i, j) = \frac{d_{\mathrm{rot}}(i, j)}{\bar{A}_{\mathrm{rot}}} + \frac{d_{\mathrm{trans}}(i, j)}{\bar{A}_{\mathrm{trans}}}. \tag{22}$$

For each post-turning frame i, we select the candidate j that minimizes $d(i, j)$. Exact ties are resolved in favor of the smaller j.

The selected pair is annotated as a true revisit only if

$$\begin{aligned} d_{\mathrm{rot}}(i, j) &< \max(3^\circ, 0.05 A_{\mathrm{rot}}), \\ d_{\mathrm{trans}}(i, j) &< \max(0.05, 0.05 A_{\mathrm{trans}}). \end{aligned} \tag{23}$$

The translation floor is expressed in the coordinate units of the benchmark trajectory. Evaluation additionally requires a temporal separation $i - j \ge 10$. Let $\mathcal{P}_c^{\mathrm{rev}}$ denote the true revisit pairs retained for evaluation in case c.

Control-pair construction. For each annotated true revisit pair $p = (j, i)$, with $j < i$, we construct a matched control group $\mathcal{C}_c(p)$ using only the prescribed camera trajectory. The control pose-separation score is

$$d_{\mathrm{pose}}(a, b) = \max \left\{ \frac{d_{\mathrm{rot}}(a, b)}{2 \max(3^\circ, 0.05 A_{\mathrm{rot}})}, \frac{d_{\mathrm{trans}}(a, b)}{2 \max(0.05, 0.05 A_{\mathrm{trans}})} \right\}. \tag{24}$$

We require $d_{\mathrm{pose}}(a, b) \ge 1$. Thus, at least one of the rotational or translational separations must reach twice its corresponding revisit tolerance. This maximum-based separation criterion differs from the summed distance $d(i, j)$ used to find revisit matches.

Candidate controls are drawn from the entire 81-frame trajectory, including frames on either side of the turning point. Each candidate satisfies $0 \le a < b \le 80$, $b - a \ge 10$, and is not an annotated true revisit pair. Its temporal gap must satisfy

$$|(b - a) - (i - j)| \le \delta, \tag{25}$$

where δ denotes the temporal-gap tolerance in frames.

We select at most five controls per revisit pair. The tolerance starts at $\delta = 2$ and increases to 4 and then 8 only when fewer than five candidates qualify. If fewer than five remain at $\delta = 8$, we retain the available candidates. A control group may therefore contain between zero and five pairs.

Candidates are sorted in ascending lexicographic order by

$$\left(|(b-a)-(i-j)|\,, d_{\text{pose}}(a,b), a, b\right), \tag{26}$$

and the first five are selected, or all candidates if fewer are available. This prioritizes temporal-gap agreement, followed by smaller pose separation among candidates meeting the threshold. Frame indices resolve remaining ties.

Control construction is deterministic and does not depend on generated-video features, model outputs, or random sampling. The selected temporal tolerance and pose-separation values are stored in the annotations. In the 200-case benchmark annotations, 95.3% of true revisit pairs have at least one control, while five cases have no controls for any revisit pair.

We preserve each pair-to-control association. Repeated control pairs may be shared across groups; their visual scores are computed once and reused without changing their contributions to the matched groups.

Pair-level visual metrics. Let I_a denote RGB frame a after visual-metric preprocessing. Let $m_k(a,b)$ denote the pairwise score between frames a and b for component $k \in \{\text{DINO}, \text{LPIPS}, \text{PSNR}, \text{SSIM}\}$.

DINO similarity is the cosine similarity of normalized frame embeddings:

$$m_{\text{DINO}}(a,b) = \left\langle \hat{\phi}_{\text{DINO}}(I_a), \hat{\phi}_{\text{DINO}}(I_b) \right\rangle, \tag{27}$$

where $\hat{\phi}_{\text{DINO}}(I)$ denotes the L2-normalized DINOv2 feature of frame I.

LPIPS uses the VGG-based model on frames remapped from $[0,1]$ to $[-1,1]$:

$$m_{\text{LPIPS}}(a,b) = \text{LPIPS}_{\text{VGG}}(2I_a - 1, 2I_b - 1). \tag{28}$$

PSNR uses mean squared error over all pixels and channels:

$$m_{\text{PSNR}}(a,b) = 10 \log_{10} \frac{1}{\max(\text{MSE}(I_a, I_b), 10^{-10})}. \tag{29}$$

The lower clamp prevents division by zero for identical frames.

The SSIM score $m_{\text{SSIM}}(a,b)$ uses a custom batched implementation with a 7×7 uniform window and stabilizing constants $C_1 = 0.01^2$ and $C_2 = 0.03^2$. Scores are averaged over all pixels and channels.

Raw revisit metrics. For each case c, the raw revisit score for component k is

$$\text{RCS}_{c,k} = \frac{1}{|\mathcal{P}_c^{\text{rev}}|} \sum_{(a,b)\in\mathcal{P}_c^{\text{rev}}} m_k(a,b). \tag{30}$$

RCS uses all valid true revisit pairs, regardless of whether matched controls are available. A case with no valid revisit pairs is assigned NaN.

Higher raw DINO, PSNR, and SSIM indicate greater similarity, whereas lower raw LPIPS indicates greater similarity. These raw component averages are distinct from the final normalized RCS score.

Matched control scores and revisit margins. For a revisit pair $p = (j,i)$ with a nonempty control group, let $\overline{m}_{c,k}(p)$ denote its mean control score:

$$\overline{m}_{c,k}(p) = \frac{1}{|\mathcal{C}_c(p)|} \sum_{(a,b)\in\mathcal{C}_c(p)} m_k(a,b), \qquad k \in \{\text{DINO}, \text{LPIPS}\}. \tag{31}$$

The denominator is the actual control-group size rather than the maximum size of five.

Let $\mathcal{P}_c^{\mathrm{ctrl}}$ denote the subset of evaluated revisit pairs with at least one matched control:

$$\mathcal{P}_c^{\mathrm{ctrl}} = \{p \in \mathcal{P}_c^{\mathrm{rev}} : |\mathcal{C}_c(p)| > 0\}. \tag{32}$$

The raw per-case margins are

$$\begin{aligned} \mathrm{MARGIN_DINO}_c &= \frac{1}{|\mathcal{P}_c^{\mathrm{ctrl}}|} \sum_{p=(j,i)\in\mathcal{P}_c^{\mathrm{ctrl}}} [m_{\mathrm{DINO}}(j,i) - \overline{m}_{c,\mathrm{DINO}}(p)], \\ \mathrm{MARGIN_LPIPS}_c &= \frac{1}{|\mathcal{P}_c^{\mathrm{ctrl}}|} \sum_{p=(j,i)\in\mathcal{P}_c^{\mathrm{ctrl}}} [\overline{m}_{c,\mathrm{LPIPS}}(p) - m_{\mathrm{LPIPS}}(j,i)]. \end{aligned} \tag{33}$$

The LPIPS difference is reversed because smaller LPIPS indicates greater similarity. Pairs without controls are excluded rather than assigned zero. If $\mathcal{P}_c^{\mathrm{ctrl}}$ is empty, the per-case margins are assigned NaN.

Each eligible revisit pair receives equal weight within its case, independently of its control-group size. Larger margins indicate greater similarity at revisited views than at temporally matched, pose-separated controls.

Temporal stability proxy. Let T denote the number of frames actually included in the visual-metric sequence. For $T \geq 2$, the raw stability component is

$$\mathrm{FLICKER_PSNR}_c = \frac{1}{T-1} \sum_{i=0}^{T-2} m_{\mathrm{PSNR}}(i, i+1). \tag{34}$$

Stability uses only mean adjacent-frame PSNR. It does not include the temporal slope of similarity to the first frame, since that similarity can change under camera motion even when the scene remains consistent.

Adjacent-frame PSNR is a short-term smoothness proxy. It does not compensate for camera motion and may favor static or slow-moving videos. It should therefore be interpreted alongside Margin and Camera Following. Long-range scene recall is evaluated by RCS and Margin.

Camera-following metrics. We use the official VGGT-1B camera head (Wang et al., 2025) to estimate camera extrinsics. Camera preprocessing operates on the original decoded RGB frames rather than the 352×640 visual-metric images. Each frame is centered on a black square canvas and resized to 518×518 using bicubic interpolation. This preserves the aspect ratio without cropping or anisotropic stretching.

Let T_c denote the number of frames included in camera evaluation for case c. The selected frames pass through the VGGT aggregator and camera head, and the final prediction is decoded with `pose_encoding_to_extri_intri`. Predictions are compared with the corresponding frame indices of the commanded trajectory.

The predicted extrinsics are 3×4 camera-from-world matrices in the OpenCV convention. We append the homogeneous row to obtain 4×4 matrices and convert the commanded poses to the same convention.

For one case, let $G_i, \hat{G}_i \in \mathrm{SE}(3)$ denote the commanded and predicted camera-from-world rigid transformations at frame i, respectively. We express each trajectory relative to its first camera:

$$\bar{G}_i = G_i G_0^{-1}, \qquad \bar{\hat{G}}_i = \hat{G}_i \hat{G}_0^{-1}. \tag{35}$$

Let R_i and $\hat{R}_i$ denote the rotation blocks of $\bar{G}_i$ and $\bar{\hat{G}}_i$. The mean rotation geodesic error in degrees is

$$\mathrm{CAM_ROT_ERR_DEG} = \frac{180}{\pi T_c} \sum_{i=0}^{T_c-1} \arccos\left(\mathrm{clip}\left(\frac{\mathrm{tr}(R_i^\top \hat{R}_i) - 1}{2}, -1, 1\right)\right). \tag{36}$$

Here, tr denotes the matrix trace, and $\mathrm{clip}(x, a, b) = \min(b, \max(a, x))$. The case index is omitted from the camera formulas for readability.

Let $\mathbf{o}_i$ and $\hat{\mathbf{o}}_i$ denote the commanded and predicted camera centers, given by the translation columns of $\bar{G}_i^{-1}$ and $(\hat{\bar{G}}_i)^{-1}$, respectively. Both first-frame centers are zero. The trajectory extents are

$$L = \max_{0 \le i < T_c} \|\mathbf{o}_i - \mathbf{o}_0\|_2, \qquad \hat{L} = \max_{0 \le i < T_c} \|\hat{\mathbf{o}}_i - \hat{\mathbf{o}}_0\|_2. \tag{37}$$

We use the numerical threshold $\epsilon_{\text{cam}} = 10^{-6}$. For $L \ge \epsilon_{\text{cam}}$, a single scale factor α aligns the predicted trajectory extent to the commanded extent:

$$\begin{aligned} \alpha &= \begin{cases} L/\hat{L}, & \hat{L} \ge \epsilon_{\text{cam}}, \\ 1, & \hat{L} < \epsilon_{\text{cam}}, \end{cases} \\ \text{CAM_ATE} &= \frac{1}{T_c L} \sum_{i=0}^{T_c - 1} \|\alpha \hat{\mathbf{o}}_i - \mathbf{o}_i\|_2. \end{aligned} \tag{38}$$

Here, CAM_ATE is the normalized mean camera-center trajectory error. For $L < \epsilon_{\text{cam}}$, the trajectory is treated as pure rotation and the translation term is not applicable.

Rotation error is evaluated directly after first-frame anchoring. No additional rotation is fitted from camera centers, so the same rotation-error definition applies to pure rotations and collinear translations.

Camera Following score and valid cases. Let S_{rot} and S_{ate} denote the normalized rotation and translation scores:

$$\begin{aligned} S_{\text{rot}} &= 1 - \frac{\min(\text{CAM_ROT_ERR_DEG}, 45)}{45}, \\ S_{\text{ate}} &= 1 - \min(\text{CAM_ATE}, 1). \end{aligned} \tag{39}$$

The per-case Camera Following score is

$$S_{\text{CAM}} = \begin{cases} \frac{1}{2}(S_{\text{rot}} + S_{\text{ate}}), & L \ge \epsilon_{\text{cam}}, \\ S_{\text{rot}}, & L < \epsilon_{\text{cam}}. \end{cases} \tag{40}$$

Pure-rotation cases therefore contribute their rotation score alone.

If VGGT fails to return a valid camera trajectory, the case is marked invalid for Camera Following and excluded from its average. The reported score is the equally weighted mean of valid per-case scores after normalization and component combination. All compared methods and ablation configurations have 200 valid camera cases.

Cross-case aggregation. For each raw visual metric, we compute the equally weighted mean of its non-NaN per-case values. Cases without a defined value are excluded rather than assigned zero. RCS and Margin may therefore use different sets of valid cases.

For RCS, pairwise scores are averaged within each case before averaging across cases. For Margin, aggregation proceeds from control-group means to pair-level differences, then to per-case means, and finally to the cross-case mean. The same procedure is used for summaries grouped by motion direction, speed relation, or depth bucket.

Fixed normalization and reported scores. The table below specifies the fixed normalization of each raw metric. For the visual metrics, names without a case index denote the cross-case raw means defined above. For example, RCS_DINO is the cross-case mean of $\text{RCS}_{c,\text{DINO}}$.

Let x denote the raw value and $n(x)$ its normalization. These transforms are not fitted to the compared methods, so adding another method does not change them. For RCS, Margin, and Stability, normalization is applied to the cross-case raw means. Camera Following instead uses per-case normalization and combination before averaging, as defined above.

Raw metric	Normalization $n(x)$
RCS_DINO	x
RCS_LPIPS	$1 - x$
RCS_PSNR	$\min(x, 50)/50$
RCS_SSIM	x
MARGIN_DINO	$\text{clip}(x, 0, 1)$
MARGIN_LPIPS	$\text{clip}(x, 0, 1)$
FLICKER_PSNR	$\min(x, 50)/50$
CAM_ROT_ERR_DEG	$1 - \min(x, 45)/45$
CAM_ATE	$1 - \min(x, 1)$

The final RCS score is the equally weighted mean of the four normalized RCS components. The final Margin score is the equally weighted mean of the normalized DINO and LPIPS margins. Stability is the normalized adjacent-frame PSNR summary. The final Camera Following score is the mean of the valid per-case S_{CAM} values.

LPIPS and camera errors are reversed so that higher transformed values indicate better performance. The identity and LPIPS-reversal transforms do not explicitly clip to $[0, 1]$ and therefore do not guarantee this range when raw inputs lie outside it.

6.6 Additional SRBench Results

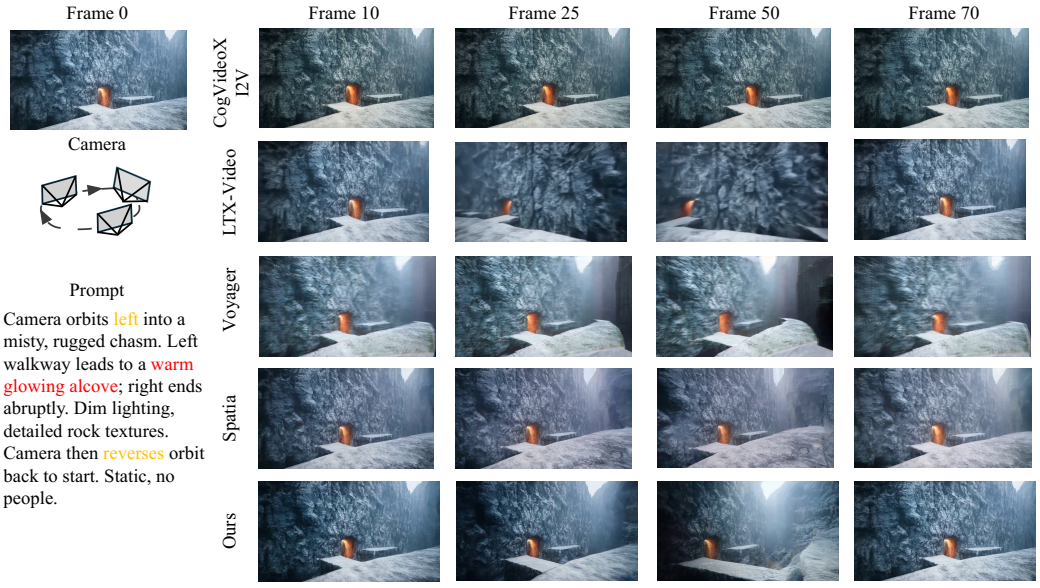

Figure 4: Additional SRBench comparison on an orbit-return trajectory. The camera is instructed to orbit left into a misty rocky chasm and then reverse toward the starting view. The scene contains narrow walkways and a warmly lit alcove. The left column shows the initial frame, camera-path illustration, and prompt; the remaining columns show frames 10, 25, 50, and 70 for the five compared methods. The sequence allows comparison of the rock structures, walkway layout, and alcove across intermediate and return views.

Figures 4 and 5 provide additional qualitative comparisons from the SRBench, complementing the indoor camera-return example in Figure 3. Each figure shows the initial frame, the prescribed camera motion, and generated frames from CogVideoX-I2V (Yang et al., 2024), LTX-Video (HaCohen et al., 2024), Voyager (Huang et al., 2025), Spatia (Zhao et al., 2026), and our method. The examples cover an orbit-return trajectory through a rocky chasm and a pan-return trajectory inside a living space, illustrating the persistence of scene landmarks during camera motion and upon return.

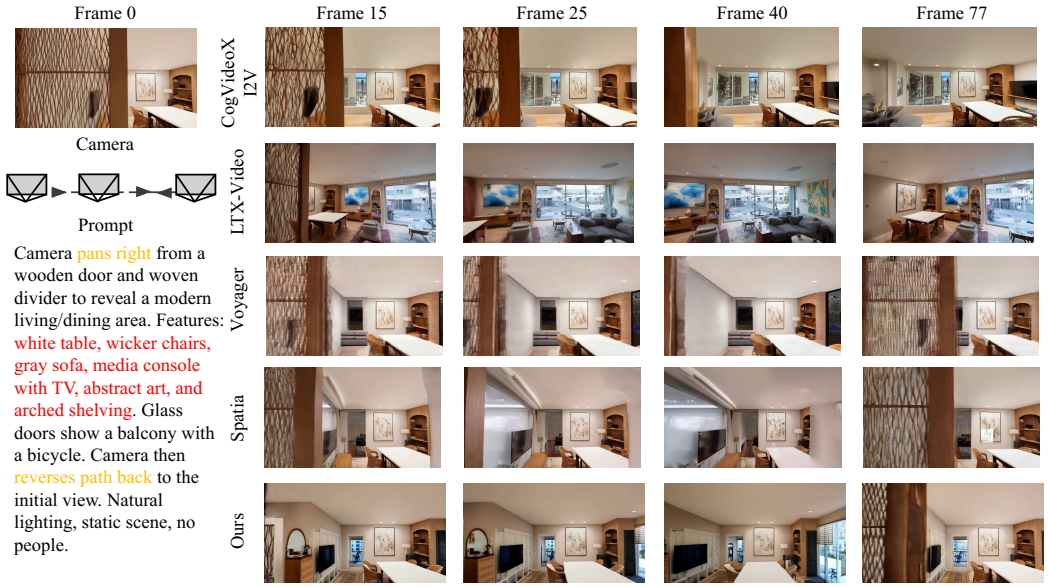

Figure 5: Additional SRBench comparison on an indoor pan-return trajectory. The camera is instructed to pan right from a wooden door and woven divider into a living and dining area, then reverse toward the initial view. The left column shows the initial frame, camera-path illustration, and prompt; the remaining columns show frames 15, 25, 40, and 77 for the five compared methods. The divider, dining table, wall art, and shelving provide scene landmarks for comparing intermediate views and the final return view.

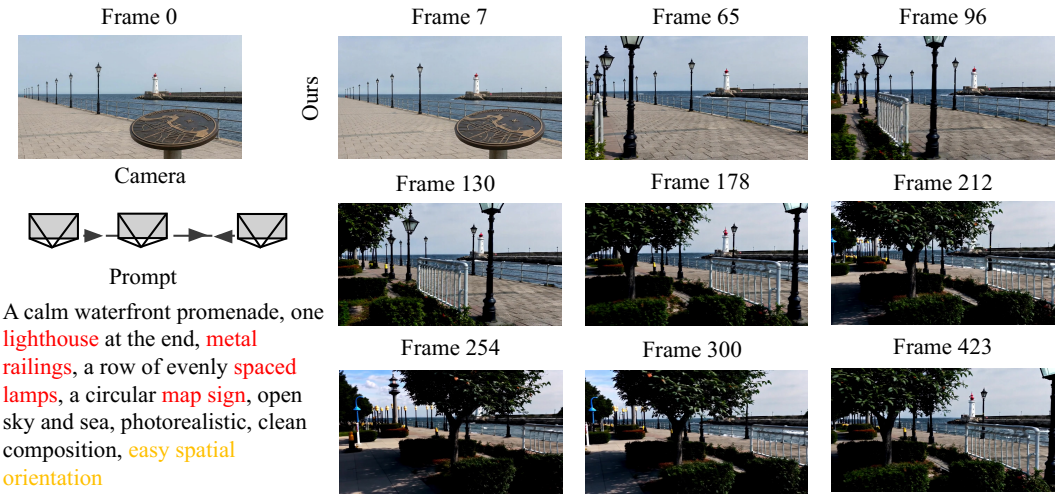

Figure 6: Selected frames from a 484-frame camera-controlled rollout generated by TraceWorld on a waterfront promenade at 16 fps (approximately 30 seconds). The left column shows the input image, prescribed camera path, and prompt. The remaining panels show selected frames from the rollout. The lighthouse, railings, street lamps, trees, and planters provide visual landmarks for inspecting scene continuity over the extended trajectory.

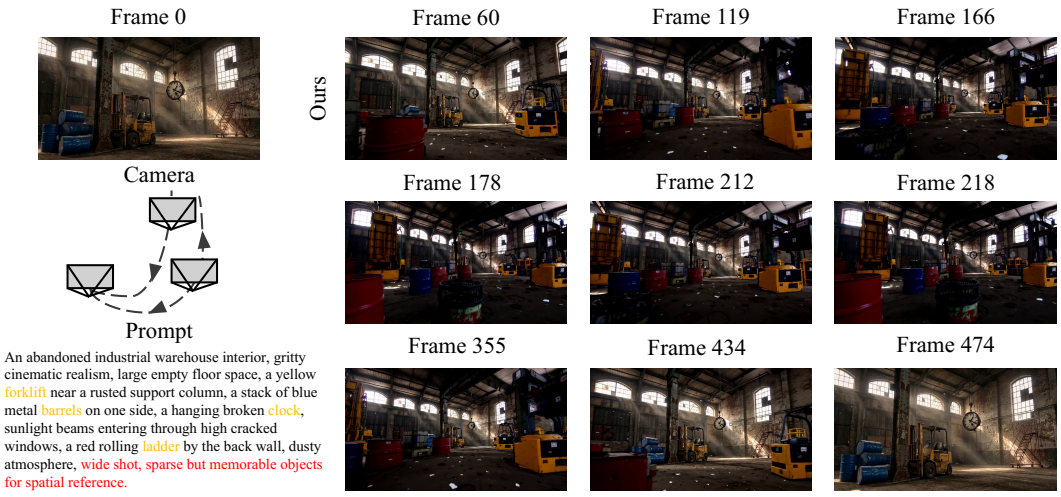

Figure 7: Selected frames from a 484-frame camera-controlled rollout generated by TraceWorld in an industrial warehouse at 16 fps (approximately 30 seconds). The left column shows the input image, prescribed camera path, and prompt. The remaining panels show selected views during camera motion and the return toward the initial viewpoint. The forklift, barrels, hanging clock, and red ladder provide visual landmarks for inspecting scene continuity and recall over the extended trajectory.

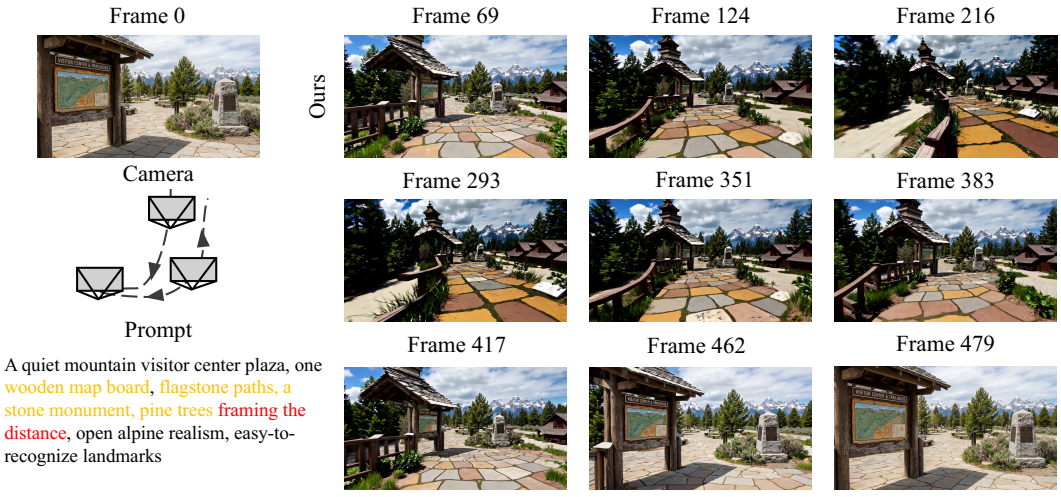

Figure 8: Selected frames from a 484-frame camera-controlled rollout generated by TraceWorld in a mountain visitor-center plaza at 16 fps (approximately 30 seconds). The left column shows the input image, prescribed camera path, and prompt. The remaining panels show selected views as the camera moves away from and returns toward the initial viewpoint. The wooden map board, flagstone paths, stone monument, and surrounding trees provide visual landmarks for inspecting scene continuity and recall over the extended trajectory.

Extended rollouts. Figures 6, 7, and 8 present three extended camera-controlled rollouts generated by TraceWorld, covering a waterfront promenade, an industrial warehouse, and a mountain visitor-center plaza, respectively. Each rollout contains 484 frames generated at 16 fps (approximately 30 seconds), with selected frames shown in the figures. Recognizable scene landmarks across the displayed views allow qualitative inspection of scene continuity and recall over extended camera trajectories. These examples complement the 81-frame quantitative evaluation with longer-horizon qualitative results.