CONSISTENT VIDEO WORLD MODEL WITH GEOMETRY-AWARE ROTARY POSITION EMBEDDING

Chendong Xiang[1*] **Jiajun Liu**[2,*] **Jintao Zhang**[1,*] **Xiao Yang**[1] **Zhengwei Fang**[1]
Shizun Wang[3] **Zijun Wang**[4] **Yingtian Zou**[5] **Hang Su**[1†] **Jun Zhu**[1†]
[1] Dept. of Comp. Sci. and Tech., Institute for AI, BNRist Center, THBI Lab, Tsinghua-Bosch Joint ML Center, Tsinghua University
[2] Gaoling School of Artificial Intelligence, Renmin University of China
[3] National University of Singapore [4] Peking University [5] Shanghai Jiao Tong University
{suhangss, dcszj}@tsinghua.edu.cn

ABSTRACT

Predictive world models that simulate future observations under explicit camera control are fundamental to interactive AI. Despite rapid advances, current systems lack spatial persistence: they fail to maintain stable scene structures over long trajectories, frequently hallucinating details when cameras revisit previously observed locations. We identify that this geometric drift stems from reliance on screen-space positional embeddings, which conflict with the projective geometry required for 3D consistency. We introduce **ViewRope**, a geometry-aware encoding that injects camera-ray directions directly into video transformer self-attention layers. By parameterizing attention with relative ray geometry rather than pixel locality, ViewRope provides a model-native inductive bias for retrieving 3D-consistent content across temporal gaps. We further propose **Geometry-Aware Frame-Sparse Attention**, which exploits these geometric cues to selectively attend to relevant historical frames, improving efficiency without sacrificing memory consistency. We also present **ViewBench**, a diagnostic suite measuring loop-closure fidelity and geometric drift. Our results demonstrate that ViewRope improves long-term consistency while reducing computational costs.

1 INTRODUCTION

Developing pose-conditioned visual world models—predictive simulators that generate future observations under an *explicit viewpoint trajectory*—is fundamental to interactive AI systems (Ha & Schmidhuber, 2018; Hafner et al., 2023; Yang et al., 2025b; Zhang et al., 2025c). Despite remarkable progress in open-domain video diffusion models (Wan et al., 2025; Kong et al., 2025; Google DeepMind, 2025; Bao et al., 2024), current generators fail to maintain *long-term geometric consistency*: as viewpoints evolve, they do not preserve stable scene structures that can be revisited. This deficiency becomes most apparent in *loop-closure* trajectories (e.g., *rotate-away-rotate-back*), where the camera returns to a previously observed viewpoint after traversing elsewhere. A consistent world model should reconstruct identical structures and appearances upon revisiting. Instead, existing generators frequently hallucinate new details or drift, revealing the absence of a reliable mechanism for retaining and retrieving 3D-consistent content over time.

Existing approaches typically address this challenge through two strategies. The first enlarges context via external retrieval or memory—selecting historical frames based on field-of-view overlap or maintaining explicit 3D spatial structures (Yu et al., 2025a; Wu et al., 2025; Huang et al., 2025a; Oshima et al., 2025). However, these mechanisms rely on pixel-level concatenation or external data structures rather than being integrated into the model's internal representation natively; this often incurs substantial compute and can become brittle when histories are long or camera motion is complex. The second employs geometry-first pipelines such as 3D Gaussian Splatting (Kerbl et al.,

[*] Equal Contributions.
[†] Correspondence Author.

2023) and specialized novel-view synthesis transformers (Jin et al., 2024; Gao et al., 2024a; Li et al., 2025b)—which enforce strict 3D consistency but typically sacrifice open-domain flexibility.

We trace this failure to a bottleneck in *positional modeling*. Most video transformers parameterize space–time structure in *screen coordinates* (x, y, t) via learned absolute/relative embeddings (Su et al., 2023). However, under camera rotation and translation, correspondence is dictated by projective geometry: the same 3D point can map to widely separated image locations across time, and conversely, nearby pixels need not be co-visible. As a result, screen-space positional bias is misaligned with the invariances required for view-consistent generation, inducing *geometric drift* that compounds over long trajectories and becomes most evident at loop closure when the camera revisits previously observed viewpoints (Sec. 4). The central challenge is thus to equip transformers with a mechanism to identify and reuse *the same physical content* across temporally distant tokens whose image-plane coordinates are decorrelated by camera motion—without resorting to explicit 3D scene reconstruction or compromising open-domain generative flexibility.

Our key insight is that long-horizon view consistency is governed by *angular correspondence of viewing directions*, rather than locality in the image plane—and that this geometric prior can be embedded *directly into the attention mechanism* without external memory structures. Under calibrated camera motion, two tokens are likely to be informative to each other when their associated camera rays are *co-visible* (i.e., intersect the same physical content), even if they are separated by long temporal gaps and occupy unrelated pixel coordinates. Motivated by this principle, we propose **ViewRope**, a geometry-aware positional encoding that injects patch-level viewing-ray directions into self-attention through ray-based rotary transformations of the query/key features. In contrast to standard 2D/3D RoPE, which encodes pixel-space offsets, ViewRope parameterizes attention as a function of *relative ray geometry*, enabling the attention mechanism to retrieve and reuse consistent 3D content from long histories without external modules. Unlike explicit memory approaches that maintain external data structures, ViewRope realizes geometric correspondence *implicitly* through attention itself, offering a lightweight and complementary mechanism. To further support long-context generation, we introduce **Geometry-Aware Frame-Sparse Attention**, which leverages geometry-conditioned relevance to select a small set of co-visible historical frames, replacing quadratic dense attention with geometry-driven sparsity while preserving loop-closure fidelity.

To evaluate view consistency directly, we introduce **ViewBench**, a diagnostic benchmark tailored to camera-conditioned long-horizon generation. Unlike generic perceptual metrics (e.g., FVD/IS) that primarily measure frame quality, ViewBench targets loop-closure trajectories such as *rotate-away-rotate-back* and quantifies revisit fidelity and geometric drift. Experiments show that our approach improves view-consistency on ViewBench, particularly at moderate rotation angles (30°–75°), while remaining efficient, narrowing the gap between geometry-rigid 3D pipelines and open-domain diffusion generators. Our contributions are summarized as follows:

- **ViewRope**: a geometric positional encoding that injects *patch-level* camera-ray directions into attention, yielding a model-native inductive bias for long-term geometric consistency.
- **Geometry-Aware Frame-Sparse Attention**: an efficient, geometry-conditioned retrieval mechanism selecting co-visible past frames, enabling consistent long-video generation with low latency.
- **ViewBench**: a targeted evaluation suite for quantifying view-consistency and loop-closure behavior in camera-conditioned video generation models.

2 Related Work

2.1 Conditioning Transformers on Camera Geometry

Camera conditioning is essential for binding geometric viewpoint information to visual tokens in multiview and video transformers. A dominant approach is to encode camera parameters into raymaps—per-pixel 6D embeddings containing ray origins and directions or Plücker coordinates Mildenhall et al. (2020); Zhang et al. (2024); Gao et al. (2024b); Jin et al. (2024); Weber et al. (2025). Concatenating these raymaps to input tokens allows models to condition on both intrinsics and extrinsics Zhang et al. (2024). However, raymaps typically rely on a global reference frame, which can be arbitrary and hinder generalization Mildenhall et al. (2019); Guizilini et al. (2024).

To mitigate this, recent methods employ relative attention-level encodings that leverage the geometric relationship between views. CaPE and GTA embed relative SE(3) pose by applying transformations directly to attention mechanisms Safin et al. (2023); Miyato et al. (2024); Kong et al. (2024). PRoPE Li et al. (2025b) models the complete relative projective transformation, encoding both camera intrinsics and extrinsics within the attention mechanism to better ground visual tokens in 3D space. While these methods improve view synthesis and avoid global frame dependency, they operate at *per-camera* granularity—all patches within a single view share the same encoding—lacking fine-grained within-view geometric modeling. In concurrent work, RayRoPE (Wu et al., 2026) predicts per-token depth and computes expected RoPE under depth uncertainty, adding depth awareness at the cost of extra parameters. ViewRope is complementary: it prioritizes simplicity with direction-only SO(3) encoding while achieving strong loop-closure performance; integrating depth-aware encoding is a promising future direction.

2.2 Interactive World Models

Recent interactive world models enable controllable simulation of physical environments by conditioning video generation on user actions and historical context Che et al. (2024); Zhang et al. (2025c); Li et al. (2025a). To support real-time interaction, the field has seen a paradigm shift from bidirectional diffusion to causal or autoregressive architectures Valevski et al. (2024); Yin et al. (2025); Yang et al. (2025a); Huang et al. (2025b), often utilizing KV-caching and distillation to accelerate inference. While scaling training on massive gameplay datasets enables foundation models like Matrix-Game (Zhang et al., 2025c) and Hunyuan-GameCraft Li et al. (2025a) to achieve high-dynamic controllability, maintaining long-term spatial consistency—particularly during scene revisits—remains a critical bottleneck (Lian et al., 2025).

To address this limitation, recent works introduce explicit memory mechanisms. Context-as-Memory (Yu et al., 2025a) retrieves historical frames based on field-of-view overlap and concatenates them into the generation context. Memory Forcing (Huang et al., 2025a) incorporates a geometry-indexed spatial memory to enforce coherence across extended horizons. More ambitiously, Wu et al. (Wu et al., 2025) propose augmenting world models with explicit 3D point-cloud memory inspired by human spatial cognition, while WorldPack (Oshima et al., 2025) compresses trajectory history via packing and selective retrieval. These memory-based approaches demonstrate improved loop-closure consistency but rely on external data structures that are not native to the attention mechanism. Our work instead embeds geometric correspondence *directly* into positional encoding, enabling implicit memory retrieval through attention without auxiliary modules.

2.3 Sparse Attention with Long Sequence

The quadratic complexity of the self-attention mechanism with respect to sequence length poses a significant challenge for modeling long sequences. To address this bottleneck, recent works have explored sparse attention mechanisms that reduce computational cost by attending to only a subset of tokens. In both language and vision domains, various sparse attention methods have been proposed. These approaches typically rely on learnable schemes (Gao et al., 2025; DeepSeek-AI et al., 2025), pattern-based selection (Lai et al., 2025; Xi et al., 2025), or low-cost dynamic estimation (Zhang et al., 2025b;a; Zhu et al., 2025b;a). However, in the setting of Autoregressive Diffusion for video generation, sparse attention remains relatively underexplored. Current state-of-the-art approaches typically rely on Sliding Window Attention to handle long temporal sequences. For instance, LongLive (Yang et al., 2025a) utilizes short window attention combined with frame sinks to maintain efficiency and consistency during real-time interactive long video generation.

3 Method

3.1 Problem Formulation

We study *pose-conditioned video generation* as a visual world model. Given an initial observation $\mathbf{x}_0$ (or a short context $\mathbf{x}_{\leq 0}$) and a target camera trajectory, the model generates a future video $\mathbf{x}_{1:T}$ that is consistent with the specified viewpoint evolution.

Let $\mathcal{C}_{1:T} := \{(\mathbf{R}_t, \mathbf{P}_t, \mathbf{K}_t)\}_{t=1}^{T}$ denote the camera trajectory, where $\mathbf{R}_t \in SO(3)$, $\mathbf{P}_t \in \mathbb{R}^3$, and $\mathbf{K}_t \in \mathbb{R}^{3\times3}$ represent camera rotation, translation, and intrinsics at time t, respectively. Optionally, we map the trajectory to a low-level *action prompt* (e.g., turn, move) and provide a global text description of the scene. We denote the resulting text/action conditioning by $\mathcal{Y}$. Our goal is to learn the conditional distribution

$$p_\theta(\mathbf{x}_{1:T} \mid \mathbf{x}_{\le 0}, \mathcal{C}_{1:T}, \mathcal{Y}). \tag{1}$$

Standard video generators mainly enforce *local temporal coherence*, which only constrains adjacent frames. For a generic photometric/perceptual distance $d(\cdot,\cdot)$, this is captured by

$$\mathcal{L}_{\text{temp}}(\theta) := \mathbb{E}_{\mathbf{x}\sim p_\theta}\left[\sum_{t=2}^{T} d(\mathbf{x}_t, \mathbf{x}_{t-1})\right]. \tag{2}$$

Such objectives do not prevent *long-horizon geometric drift* under camera motion, because screen-space proximity is not aligned with physical correspondence.

In contrast, a pose-conditioned world model must satisfy a *loop-closure* requirement: if the camera at time t revisits (approximately) a viewpoint observed at some $k \ll t$, then the rendered observations should agree up to projective geometry on their co-visible region. Concretely, define the revisit indicator

$$w_{t,k} := \mathbb{I}(\Delta(\mathcal{C}_t, \mathcal{C}_k) \le \varepsilon), \qquad k < t, \tag{3}$$

where $\Delta(\cdot,\cdot)$ measures pose similarity (e.g., rotation/ translation discrepancy under calibrated intrinsics) and ε is a tolerance threshold. When $w_{t,k} = 1$, the generated frames $\mathbf{x}_t$ and $\mathbf{x}_k$ should be photometrically consistent on their co-visible region $\Omega_{t,k}$. Let $\mathcal{W}_{k\leftarrow t}$ denote the projective warp from t to k (see Appendix B for details). We formalize this via a loop-closure loss:

$$\mathcal{L}_{\text{lc}}(\theta) := \mathbb{E}_{\mathbf{x}\sim p_\theta}\left[\sum_{t=1}^{T}\sum_{k<t} w_{t,k} \cdot \sum_{\mathbf{u}\in\Omega_{t,k}} \rho(\mathbf{x}_t(\mathbf{u}) - \mathbf{x}_k(\mathcal{W}_{k\leftarrow t}(\mathbf{u})))\right], \tag{4}$$

where $\rho(\cdot)$ is a robust penalty. *Note:* Eq. (4) is **not directly optimized** during training; it formalizes the consistency requirement that motivates our architectural design.

The difficulty is that this loop-closure objective couples frames across *arbitrarily long* temporal gaps: achieving loop closure requires retrieving geometrically corresponding content from a long history under causal (streaming) generation. Rather than explicitly optimizing a pixel-level consistency loss, we parameterize p_θ with a Diffusion Transformer (DiT) and inject 3D view geometry into its attention mechanism, so that (i) attention scores become sensitive to *relative viewing rays*, and (ii) the model can efficiently select and attend to the most geometrically relevant historical frames while generating online.

3.2 ViewRope: View-Centric Positional Encoding in Attention

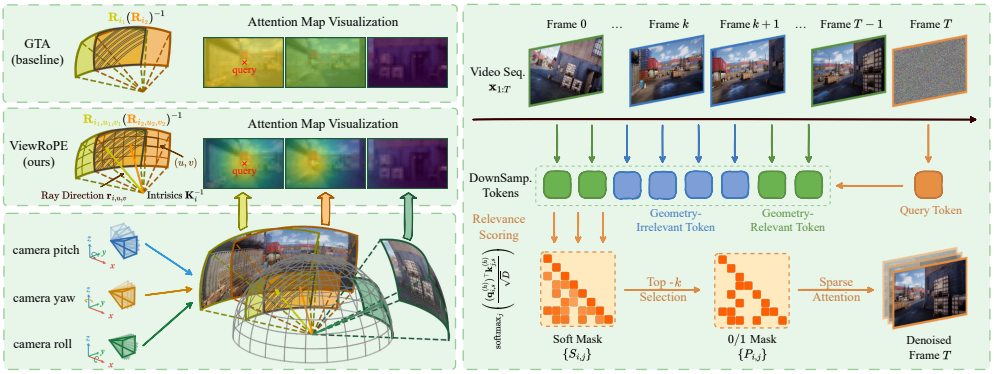
We introduce *View-centric Position Encoding* (**ViewRope**), encoding each token's 3D viewing direction *directly* into the attention mechanism. Unlike 2D/3D positional embeddings or global pose tokens, ViewRope assigns a *per-patch* rotation derived from camera intrinsics/extrinsics, so attention can operate on relative view geometry at patch granularity.

Per-patch ray construction. For a patch centered at pixel coordinates (u, v) in camera/view i, we compute its normalized viewing ray $\mathbf{r}_{i,u,v} \in \mathbb{S}^2$ (the unit sphere) in the camera coordinate system using intrinsics $\mathbf{K}_i$:

$$\mathbf{r}_{i,u,v} = \frac{\mathbf{K}_i^{-1}[u, v, 1]^\top}{\|\mathbf{K}_i^{-1}[u, v, 1]^\top\|_2}. \tag{5}$$

We then build a local rotation $\mathbf{R}^{\text{local}}_{i,u,v} \in SO(3)$ that maps the canonical optical axis $\mathbf{z} = [0, 0, 1]^\top$ to $\mathbf{r}_{i,u,v}$, choosing the minimum-angle rotation (i.e., the rotation whose axis lies in the plane spanned by $\mathbf{z}$ and $\mathbf{r}$, computed via the Rodrigues formula). Combining with the camera extrinsic rotation $\mathbf{R}^{\text{cam}}_i$, we obtain a world-aligned view rotation:

$$\mathbf{R}_{i,u,v} = \mathbf{R}^{\text{cam}}_i \, \mathbf{R}^{\text{local}}_{i,u,v}. \tag{6}$$



(a) View-centric Position Encoding (ViewRope) (b) Geometry-Aware Frame Sparse Attention

Figure 1: **Method overview. (a) ViewRope** computes per-patch viewing rays from intrinsics, constructs local rotations, and rotates query/key feature subvectors in attention. The resulting dot product encodes relative angular relationships between viewing rays. **(b) Geometry-Aware Frame Sparse Attention** estimates block (frame) relevance and selects top-k geometrically relevant historical frames, replacing quadratic dense attention with geometry-driven sparsity.

Rotating query/key feature subvectors. Let $\mathbf{q} \in \mathbb{R}^d$ and $\mathbf{k} \in \mathbb{R}^d$ be query/key vectors for a token. We reserve a subset of channels that can be grouped into m 3D subvectors (so $3m \leq d$), and rotate each subvector by $\mathbf{R}_{i,u,v}$:

$$\mathrm{VR}(\mathbf{q}, \mathbf{R}_{i,u,v}) = \mathbf{q}', \tag{7}$$

where $\mathbf{q}'_{3\ell:3\ell+3} = \mathbf{R}_{i,u,v}\mathbf{q}_{3\ell:3\ell+3}$, $\ell = 0, \ldots, m-1$. We apply the same transformation to keys. Intuitively, this aligns a portion of the feature space with the physical viewing direction of each patch in world coordinates.

Geometry-aware attention scores. Consider a query token from view i at (u_i, v_i) and a key token from view j at (u_j, v_j). Their rotated dot product becomes

$$\left\langle \mathrm{VR}(\mathbf{q}, \mathbf{R}_{i,u_i,v_i}),\ \mathrm{VR}(\mathbf{k}, \mathbf{R}_{j,u_j,v_j}) \right\rangle = \mathbf{q}^\top \mathbf{R}^\top_{i,u_i,v_i} \mathbf{R}_{j,u_j,v_j} \mathbf{k} = \mathbf{q}^\top \left(\mathbf{R}^{-1}_{i,u_i,v_i} \mathbf{R}_{j,u_j,v_j} \right) \mathbf{k}. \tag{8}$$

The relative rotation $\mathbf{R}^{-1}_{i,u_i,v_i}\mathbf{R}_{j,u_j,v_j}$ captures the angular relationship between the two viewing rays. This makes attention naturally sensitive to 3D view similarity, improving long-range recall and loop closure consistency.

3.3 Geometry-Aware Frame Sparse Attention

Long-context generation with dense attention scales quadratically with sequence length. To support streaming world-modeling over many frames, we adopt a *frame-aligned block-sparse* attention scheme inspired by SampleAttention (Zhu et al., 2025a). The key design is to set the block size B to exactly match one latent frame, so blocks correspond to time steps. As ViewRope encodes 3D viewing geometry in the attention space, we can directly estimate frame-level geometric relevance.

Let $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{L \times D}$ be token sequences, partitioned into N blocks of size B ($L = NB$). Denote block i as $\mathbf{Q}_i \in \mathbb{R}^{B \times D}$ and block j as $\mathbf{K}_j, \mathbf{V}_j \in \mathbb{R}^{B \times D}$.

Block relevance estimation (stochastic). Instead of computing full block-to-block attention, we sample a small set of token indices $\mathcal{S} \subset \{1, \ldots, B\}$ of size K_s (e.g., $K_s = 10$) and estimate a head-averaged affinity:

$$\tilde{S}_{ij} = \frac{1}{HK_s} \sum_{h=1}^{H} \sum_{s \in \mathcal{S}} \frac{(\mathbf{q}^{(h)}_{i,s})^\top \mathbf{k}^{(h)}_{j,s}}{\sqrt{D}}. \tag{9}$$

We then apply a causal constraint via a block mask $\mathbf{M}^{\text{causal}}$ (disallowing $j > i$ in the streaming setting) by setting $\tilde{S}_{ij} = -\infty$ when $M^{\text{causal}}_{ij} = 0$.

Top-k block selection. For each query block i, we select the top-k key blocks under $\tilde{S}_{ij}$ among valid past blocks:

$$\mathcal{T}_i = \mathrm{TopK}\big(\{\tilde{S}_{ij}\}_{j:\, M_{ij}^{\text{causal}}=1}\big). \tag{10}$$

We always include $j = i$ to preserve local context. The final sparsity mask is

$$M_{ij} = \begin{cases} 1 & \text{if } (j \in \mathcal{T}_i \text{ or } j = i) \ \wedge \ M_{ij}^{\text{causal}} = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Sparse attention computation. We compute attention for block i only over selected blocks:

$$\mathbf{O}_i = \mathrm{softmax}\left(\frac{\mathbf{Q}_i \left(\mathbf{K}_{\{j \mid M_{ij}=1\}}\right)^\top}{\sqrt{D}}\right) \mathbf{V}_{\{j \mid M_{ij}=1\}}. \tag{12}$$

With fixed k, the attention cost scales linearly with the number of frames, enabling efficient long-horizon generation. We implement the sparse attention kernel with TileLang (Wang et al., 2025), following the optimization principles of FlashAttention (Dao, 2023).

Training vs. inference (streaming with cache). We utilize teacher-forcing (Arriola et al., 2025) to enable AR-generation. During training, historical frames are taken from ground-truth latents, forming a clean KV cache; the current denoising step queries this cache with ViewRope-rotated features and a top-k frame mask. During autoregressive inference, we maintain a KV cache of previously generated latent frames and apply the same relevance estimation and top-k selection at each denoising step, preserving causality while retrieving geometrically relevant history. The detailed training and inference procedures are shown in Algorithm 1.

3.4 Progressive Training Pipeline

To stabilize adaptation to autoregressive streaming generation and long contexts, we employ a progressive schedule:

Stage I: Short-clip teacher forcing. Train on short clips (e.g., $\sim$17 frames) with teacher forcing to align the model with the autoregressive generation interface and caching behavior. **Stage II: Introduce ViewRope.** Enable ViewRope while keeping clips short, allowing the model to learn view-conditioned correspondence without the confound of very long contexts. **Stage III: Enable Frame Sparse Attention.** Activate frame-aligned block sparsity to adapt the model to long-context retrieval efficiently, while keeping sequence lengths moderate. **Stage IV: Scale context length.** Increase training sequence length substantially under sparse attention, endowing the model with long-horizon video generation and improved loop-closure consistency.

4 Experiments

We conduct experiments to validate the effectiveness of ViewRope. All experiments use the same backbone, training budget, and data to ensure fair comparison.

4.1 Experimental Setup

Datasets and Benchmarks. We evaluate on **ViewBench**, a diagnostic benchmark we construct to systematically evaluate view-consistency under camera motion. Existing datasets hold limitations for expected and unified evaluation: Unlike existing datasets—Context-as-Memory (Yu et al., 2025a) (yaw-only) and GF-Minecraft (Yu et al., 2025b) (no geometric overlap annotations)—ViewBench provides: **(i)** complete 3-axis rotation coverage (yaw, pitch, roll) with systematic angle sampling; **(ii)** round-trip loop-closure (Lian et al., 2025) trajectories; **(iii)** 10 photorealistic UE5 environments. The training set contains 1k+ video sequences ($\sim$500k frames), and the evaluation set consists of 600 separately collected samples with non-overlapping trajectories. ViewBench is publicly available[1]. See Appendix C for detailed comparison and dataset specifications.

Metrics. We report metrics across three categories: 1) **Visual Quality**: PSNR, SSIM, LPIPS—standard metrics for frame-level reconstruction quality; 2) **Loop Closure Error (LCE)**: LPIPS

[1] https://github.com/jedward225/viewbench-dataset

Table 1: **Position encoding comparison on ViewBench.** We report visual quality (PSNR↑, SSIM↑, LPIPS↓) and geometric consistency (LCE↓) for 30° and 75° view synthesis. Best in **bold**. Extended comparisons with additional baselines (Plücker, PRoPE) are provided in Appendix D.3.

Method	30 deg				75 deg			
	PSNR	SSIM	LPIPS ↓	LCE ↓	PSNR	SSIM	LPIPS ↓	LCE ↓
3D RoPE	17.09	0.4133	0.4219	0.4929	14.78	0.3634	0.5501	0.4831
GTA	17.33	0.4325	0.4165	0.4707	15.12	0.3784	0.5403	0.4723
ViewRope (Ours)	**17.53**	**0.4378**	**0.4080**	**0.4497**	**15.27**	**0.3916**	**0.5398**	**0.4562**

between the starting frame $\mathbf{x}_0$ and the generated frame $\mathbf{x}_t$ upon returning to the start pose, directly measuring persistent spatial memory. Lower is better.

Baselines. We compare against position encoding methods that differ in how they encode camera geometry: 1) **3D RoPE** (Su et al., 2023): Standard temporal-spatial RoPE without camera geometry, serving as a no-geometry baseline; 2) **Plücker** (Zhang et al., 2024): Additive conditioning with 6D Plücker coordinates per pixel, encoding both ray origin and direction; 3) **GTA** (Miyato et al., 2024): Relative SE(3) transformation applied to attention at per-camera granularity; 4) **PRoPE** (Li et al., 2025b): Relative projective encoding capturing both intrinsics and extrinsics at per-camera granularity. A key distinction is that Plücker, GTA, and PRoPE all operate at *per-camera* level (all patches in a view share the same encoding), while ViewRope operates at *per-patch* level (each patch receives a unique rotation based on its individual ray direction). Due to training budget constraints, the main comparison (Table 1) focuses on 3D RoPE, GTA, and ViewRope; extended comparisons including Plücker and PRoPE are in Appendix D.3.

Furthermore, we investigate the integration of ViewRope with various sparse attention mechanisms, including our proposed Geometry-Aware Sparse Attention and sliding window attention, to validate the effectiveness of our algorithm and demonstrate the efficiency improvements achieved through sparse attention patterns.

Implementation Details. We build upon WAN 2.2 TI2V-5B (Wan et al., 2025), a text-and-image-to-video diffusion transformer, and adapt it for streaming video generation via teacher-forcing training. The training data combines Context-as-Memory (Yu et al., 2025a), GF-Minecraft (Yu et al., 2025b), and ViewBench at a 1:1:1 sampling ratio. All RoPE variants are applied to the same channels for fair comparison. The ViewBench evaluation set is *separately collected* with non-overlapping trajectories. See Appendix A.1 for detailed training configurations.

4.2 View Consistency Comparison

Table 1 presents the quantitative comparison on ViewBench. We make three key observations: **(1) ViewRope achieves the best loop closure performance.** ViewRope reduces LCE by around 4% compared to GTA, the strongest baseline. This demonstrates that per-patch ray encoding provides more precise geometric alignment than per-camera encoding when revisiting previous viewpoints. **(2) Geometry-aware encoding consistently outperforms absolute encoding.** Both GTA and ViewRope outperform 3D RoPE, confirming the findings of prior work (Li et al., 2025b; Miyato et al., 2024) that relative geometric relationships are more effective than absolute coordinates. **(3) ViewRope maintains competitive visual quality.** Despite focusing on geometric consistency, ViewRope achieves comparable or better PSNR/SSIM than baselines, indicating that the geometric inductive bias does not sacrifice generation quality. We further compare with additional baselines (Plücker, PRoPE) under an extended training budget in Appendix D.3, where ViewRope's per-patch encoding consistently outperforms per-camera alternatives.

Comparison with State-of-the-Art Interactive World Models. We further compare ViewRope with two leading interactive world model systems: Matrix-Game-2 (Zhang et al., 2025c) and HY-WorldPlay (Li et al., 2025a). ViewRope consistently outperforms both baselines across all evaluated rotation magnitudes (30°–75°). The improvement is most pronounced in loop-closure error (LCE): ViewRope reduces LCE by 6.5% compared to HY-WorldPlay at 30°, 7.9% at 45°, and 11.4% at 75°, demonstrating that per-patch geometric encoding provides stronger spatial memory than action-conditioned approaches. Notably, the performance gap widens with increasing rotation angle, suggesting that ViewRope's ray-based attention becomes more beneficial for larger camera excursions

Table 2: **Sparse attention comparison on ViewBench.** We report visual quality (PSNR↑, SSIM↑, LPIPS↓) and geometric consistency (LCE↓) for 90° and 180° view synthesis. Best in **bold**.

Method	90 deg				180 deg			
	PSNR	SSIM	LPIPS ↓	LCE ↓	PSNR	SSIM	LPIPS ↓	LCE ↓
Sparse w/o ViewRope	10.97	0.080	0.8887	0.8932	9.937	0.0618	0.9286	0.9243
GTA w/ Sparse	8.603	0.0755	0.8316	0.8020	9.275	0.078	0.8062	0.7924
ViewRope w/ Sliding Window	15.20	0.3701	0.5513	0.6543	14.44	0.3406	0.6139	0.6598
ViewRope w/ Sparse (Ours)	**15.61**	**0.4001**	**0.5382**	**0.5445**	**14.35**	**0.3458**	**0.6043**	**0.5609**

where geometric correspondence is critical. Qualitative comparisons and extended results including large-angle (90°–180°) trajectories are provided in Appendix C.7.

4.3 Efficiency of Geometry-Aware Sparse Attention

Experiments setup. To validate sparse attention, we continue training the models from Section 4.2 on 61-frame sequences with various sparse attention mechanisms (top-k=5) for 6k steps, then on 201-frame sequences for 2k steps to enable longer sequence generation. We evaluate on 90° and 180° scenarios which require longer sequences.

Table 2 shows the results of sparse attention comparison on ViewBench. We make two key observations: **(1) ViewRope w/ Sparse consistently outperforms other methods**. ViewRope w/ Sparse outperforms all baselines, reducing LCE by 16% compared to sliding window attention. This demonstrates that our geometry-aware sparse attention mechanism is more effective for long-sequence view synthesis. **(2) ViewRope stabilizes sparse training**. We found that both naïve sparse attention (without geometric encoding) and GTA w/ Sparse suffer from loss divergence during training, whereas ViewRope w/ Sparse maintains stable convergence throughout. We attribute this stability to ViewRope's ray-based rotations, which impose geometrically meaningful structure on the Q/K dot products used for relevance scoring (Eq. 9). This structure yields more reliable frame selection and, consequently, more stable gradient signals during sparse training.

Counterfactual Validation. To verify causal relevance, we replace the top-k selected frames with alternatives: random selection degrades LCE from 0.5609 to 0.7027 (+25.2%), and explicitly excluding the selected frames causes even worse degradation to 0.7744 (+38.1%), confirming that our geometry-aware selection identifies causally important frames.

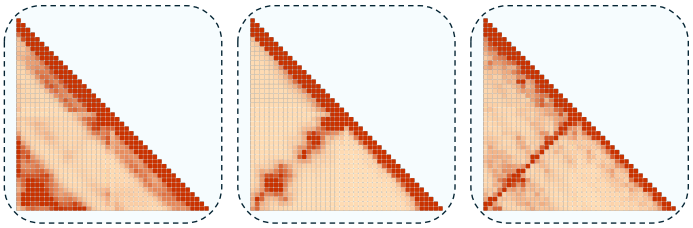


Figure 2: **Visualization of attention specialization.** Left: A standard temporal head focuses on recent or temporally periodic frames. Middle: A geometry-aware head captures long-range spatial overlap (evident in the antidiagonal activation during loop closure). Right: The aggregated attention map illustrates how geometric cues guide sparse block selection.

Visualizing Geometric Relationships. To further substantiate that the model learns meaningful geometric relationships, we visualize the attention maps for a loop closure sequence in Figure 2. We observe that different attention heads specialize in distinct patterns: while common heads (left) focus on temporal locality, specific geometry-sensitive heads (middle) emerge to capture spatial overlap. Notably, these geometry heads exhibit high activation for temporally distant but spatially aligned frames. This geometric signal is successfully integrated into the final estimated attention map (right), thereby guiding the block selection mechanism to retrieve the correct historical context.

Efficiency Comparison. We evaluate computational efficiency. Sparse attention (top-k = 5) reduces training time from 27.66 s/iter to 22.01 s/iter on 201-frame sequences, achieving a ∼25% acceleration compared to dense attention.

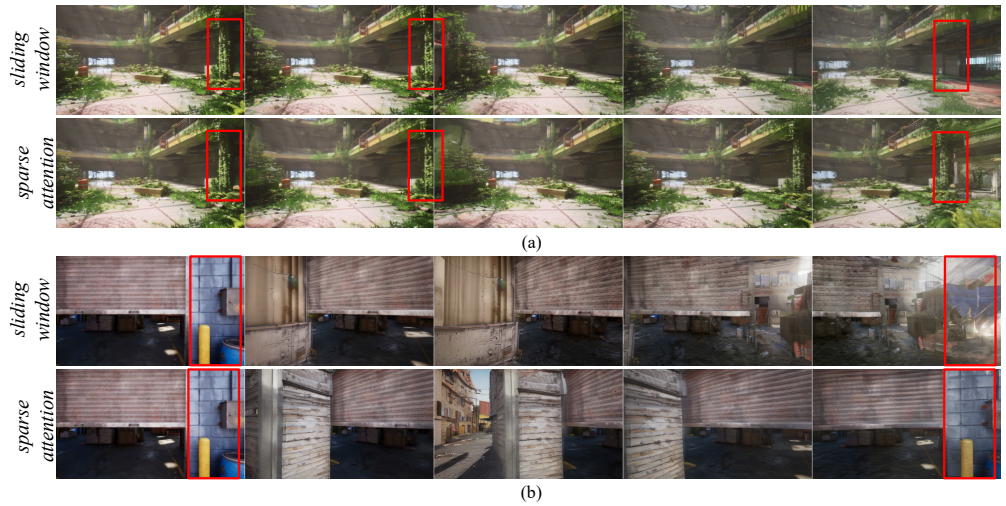


Figure 3: **Case study.** Upper and lower sequences show ViewRope with Sliding Window and Sparse attention, respectively.

Case Study. Figure 3 shows cases where ViewRope w/ Sparse outperforms other methods. In Figure 3(a), the green column on the right side disappears after the turn under the sliding window method, while ViewRope maintains it well. In Figure 3(b), the blue wall on the right side becomes blurry and exhibits noticeable drift and hallucinated details under the sliding window method, whereas ViewRope accurately recovers the original scene structure. This demonstrates that ViewRope's geometry-aware sparse attention mechanism is more effective for long-sequence geometry-aware retrieval.

4.4 Ablation Studies

Channel Allocation for ViewRope. We ablate where to embed ViewRope within the 128-dimensional 3D RoPE structure (T/H/W = 44/42/42 dims). Among four strategies tested (Table 7 in Appendix D.2), embedding in the T-dimension low-frequency bands (ch 32–44) performs best, as it encodes view information without disrupting spatial representations in H/W. Replacing original 3D RoPE or distributing ViewRope across all dimensions degrades performance.

Number of Retrieved Frames. Due to space constraints, we provide the ablation on the number of retrieved frames (k) in Appendix D.1. Increasing k generally improves visual quality (PSNR, SSIM) by accessing more texture details, but geometric consistency (LCE) peaks at $k = 5$, suggesting a trade-off between texture richness and geometric precision.

5 Conclusion and Future Work

We presented **ViewRope**, a geometric positional encoding that embeds per-patch camera-ray directions into video transformer attention, enabling long-term 3D consistency—especially in loop-closure scenarios. Combined with **Geometry-Aware Sparse Attention**, which selectively attends to geometrically relevant history frames, our approach achieves state-of-the-art consistency on the newly proposed **ViewBench** while substantially reducing computational cost. Limitations include degraded performance under drastic scene transitions and large-angle trajectories (90°–180°), where systems leveraging variable-frame-rate training and context-forcing distillation currently excel (see Appendix C.6). Future directions include self-forcing training (Huang et al., 2025b), depth-aware encoding, and extension to more complex geometric environments.

REFERENCES

Marianne Arriola, Aaron Gokaslan, Justin T. Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models, 2025. URL https://arxiv.org/abs/2503.09573.

Fan Bao, Chendong Xiang, Gang Yue, Guande He, Hongzhou Zhu, Kaiwen Zheng, Min Zhao, Shilong Liu, Yaole Wang, and Jun Zhu. Vidu: a highly consistent, dynamic and skilled text-to-video generator with diffusion models, 2024. URL https://arxiv.org/abs/2405.04233.

Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-x: Interactive open-world game video generation, 2024. URL https://arxiv.org/abs/2411.00769.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenhao Xu, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Erhang Li, Fangqi Zhou, Fangyun Lin, Fucong Dai, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoran Wei, Haowei Zhang, Haowen Luo, Haozhe Ji, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, Jialiang Huang, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jingchang Chen, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jinhua Zhu, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexin Huang, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Liang Zhao, Liangsheng Yin, Lihua Guo, Lingxiao Luo, Linwang Ma, Litong Wang, Liyue Zhang, M. S. Di, M. Y Xu, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Panpan Huang, Peixin Cong, Peiyi Wang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, S. H. Liu, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Songyang Zhou, Tao Ni, Tao Yun, Tian Pei, Tian Ye, Tianyuan Yue, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjun Gao, Wentao Zhang, Xi Gao, Xiangwen Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyuan Li, Xu Chen, Xuecheng Su, Xuehai Pan, Xuheng Lin, Xuwei Fu, Y. Q. Wang, Yang Zhang, Yanhong Xu, Yanru Ma, Yao Li, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yiliang Xiong, Ying He, Ying Zhou, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuduan Wang, Yue Gong, Yuhan Wu, Yuheng Zou, Yukun Li, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehua Zhao, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhiyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Zizheng Pan, Zongqing Yao, Bei Feng, Hui Li, J. L. Cai, Jiaqi Ni, Lei Xu, Meng Li, Ning Tian, R. J. Chen, R. L. Jin, S. S. Li, Shuang Zhou, Tianyu Sun, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xinnan Song, Xinyi Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, Dongjie Ji, Jian Liang, Jianzhong Guo, Jin Chen, Leyi Xia, Miaojun Wang, Mingming Li, Peng Zhang, Ruyi Chen, Shangmian Sun, Shaoqing Wu, Shengfeng Ye, T. Wang, W. L. Xiao, Wei An, Xianzu Wang, Xiaowen Sun, Xiaoxiang Wang, Ying Tang, Yukun Zha, Zekai Zhang, Zhe Ju, Zhen Zhang, and Zihua Qu. Deepseek-v3.2: Pushing the frontier of open large language models, 2025. URL https://arxiv.org/abs/2512.02556.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024a.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T. Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models, 2024b. URL https://arxiv.org/abs/2405.10314.

Yizhao Gao, Zhichen Zeng, Dayou Du, Shijie Cao, Peiyuan Zhou, Jiaxing Qi, Junjie Lai, Hayden Kwok-Hay So, Ting Cao, Fan Yang, and Mao Yang. Seerattention: Learning intrinsic sparse attention in your llms, 2025. URL https://arxiv.org/abs/2410.13276.

Google DeepMind. Veo 3 technical report. Technical report, Google DeepMind, 2025. URL https://storage.googleapis.com/deepmind-media/veo/Veo-3-Tech-Report.pdf.

Vitor Guizilini, Igor Vasiljevic, Jiading Fang, Rares Ambrus, Greg Shakhnarovich, Matthew Johnson-Roberson, and Adrien Gaidon. Zero-shot novel view synthesis with large-scale diffusion models, 2024.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3), 2018.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Junchao Huang, Xinting Hu, Boyao Han, Shaoshuai Shi, Zhuotao Tian, Tianyu He, and Li Jiang. Memory forcing: Spatio-temporal memory for consistent scene generation on minecraft, 2025a. URL https://arxiv.org/abs/2510.03198.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion, 2025b. URL https://arxiv.org/abs/2506.08009.

Haomin Jin, Dongheui Lee, Jialin Gu, Jiaxu Zou, Yicheng Feng, Yiming Wu, Weijie Li, Xueqi Jiang, Anpei Chen, Hao Zhang, and Xiaohui Liang. Lvsm: Large video-to-3d synthesis model, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering, 2023. URL https://arxiv.org/abs/2308.04079.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. Hunyuanvideo: A systematic framework for large video generative models, 2025. URL https://arxiv.org/abs/2412.03603.

Xin Kong, Shikun Liu, Xiaoyang Lyu, Marwan Taher, Xiaojuan Qi, and Andrew J. Davison. Eschernet: A generative model for scalable view synthesis, 2024. URL https://arxiv.org/abs/2402.03908.

Xunhao Lai, Jianqiao Lu, Yao Luo, Yiyuan Ma, and Xun Zhou. Flexprefill: A context-aware sparse attention mechanism for efficient long-sequence inference, 2025. URL https://arxiv.org/abs/2502.20766.

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.

Ruilong Li, Brent Yi, Junchen Liu, Hang Gao, Yi Ma, and Angjoo Kanazawa. Cameras as relative positional encoding, 2025b. URL https://arxiv.org/abs/2507.10496.

Kewei Lian, Shaofei Cai, Yilun Du, and Yitao Liang. Toward memory-aided world models: Benchmarking via spatial consistency, 2025. URL https://arxiv.org/abs/2505.22976.

Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines, 2019.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis, 2020.

Takeru Miyato, Bernhard Jaeger, Max Welling, and Andreas Geiger. Gta: A geometry-aware attention mechanism for multi-view transformers, 2024. URL https://arxiv.org/abs/2310.10375.

Yuta Oshima, Yusuke Iwasawa, Masahiro Suzuki, Yutaka Matsuo, and Hiroki Furuta. Worldpack: Compressed memory improves spatial consistency in video world modeling. *arXiv preprint arXiv:2512.02473*, 2025.

Aleksandr Safin, Daniel Cremers, and Laura Leal-Taixé. Repast: Relative pose attention scene representation transformer, 2023.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023. URL https://arxiv.org/abs/2104.09864.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. *arXiv preprint arXiv:2408.14837*, 2024.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models, 2025. URL https://arxiv.org/abs/2503.20314.

Lei Wang, Yu Cheng, Yining Shi, Zhengju Tang, Zhiwen Mo, Wenhao Xie, Lingxiao Ma, Yuqing Xia, Jilong Xue, Fan Yang, and Zhi Yang. Tilelang: A composable tiled programming model for ai systems, 2025. URL https://arxiv.org/abs/2504.17577.

Ethan Weber, Aleksander Holynski, Varun Jampani, Saurabh Gupta, and Noah Snavely. Fillerbuster: A consistent video generation model with explicit background layout control, 2025.

Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video world models with long-term spatial memory. *arXiv preprint arXiv:2506.05284*, 2025.

Yu Wu, Minsik Jeon, Jen-Hao Rick Chang, Oncel Tuzel, and Shubham Tulsiani. Rayrope: Projective ray positional encoding for multi-view attention. *arXiv preprint arXiv:2601.15275*, 2026.

Haocheng Xi, Shuo Yang, Yilong Zhao, Chenfeng Xu, Muyang Li, Xiuyu Li, Yujun Lin, Han Cai, Jintao Zhang, Dacheng Li, Jianfei Chen, Ion Stoica, Kurt Keutzer, and Song Han. Sparse videogen: Accelerating video diffusion transformers with spatial-temporal sparsity, 2025. URL https://arxiv.org/abs/2502.01776.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. Longlive: Real-time interactive long video generation, 2025a. URL https://arxiv.org/abs/2509.22622.

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*, 2025b.

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, 2025. URL https://arxiv.org/abs/2412.07772.

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, 2025a. URL https://arxiv.org/abs/2506.03141.

Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos, 2025b. URL https://arxiv.org/abs/2501.08325.

Jason Zhang, Ruilong Li, Matthew Tancik, Hang Gao, and Angjoo Kanazawa. Cameras as rays: Pose-conditioned transformers for free-view synthesis, 2024.

Jintao Zhang, Chendong Xiang, Haofeng Huang, Jia Wei, Haocheng Xi, Jun Zhu, and Jianfei Chen. Spargeattention: Accurate and training-free sparse attention accelerating any model inference, 2025a. URL https://arxiv.org/abs/2502.18137.

Peiyuan Zhang, Yongqi Chen, Haofeng Huang, Will Lin, Zhengzhong Liu, Ion Stoica, Eric Xing, and Hao Zhang. Vsa: Faster video diffusion with trainable sparse attention, 2025b. URL https://arxiv.org/abs/2505.13389.

Yifan Zhang, Chunli Peng, Boyang Wang, Puyi Wang, Qingcheng Zhu, Fei Kang, Biao Jiang, Zedong Gao, Eric Li, Yang Liu, and Yahui Zhou. Matrix-game: Interactive world foundation model, 2025c. URL https://arxiv.org/abs/2506.18701.

Qianchao Zhu, Jiangfei Duan, Chang Chen, Siran Liu, Guanyu Feng, Xin Lv, Xiao Chuanfu, Dahua Lin, and Chao Yang. Sampleattention: Near-lossless acceleration of long context llm inference with adaptive structured sparse attention, 2025a. URL https://arxiv.org/abs/2406.15486.

Zichen Zhu, Hao Tang, Yansi Li, Dingye Liu, Hongshen Xu, Kunyao Lan, Danyang Zhang, Yixuan Jiang, Hao Zhou, Chenrun Wang, Situo Zhang, Liangtai Sun, Yixiao Wang, Yuheng Sun, Lu Chen, and Kai Yu. Moba: Multifaceted memory-enhanced adaptive planning for efficient mobile task automation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pp. 535–549. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.naacl-demo.43. URL http://dx.doi.org/10.18653/v1/2025.naacl-demo.43.

A APPENDIX

We provide the detailed training and inference pipeline with Frame Sparse Attention under Teacher Forcing as shown in Algorithm 1.

Algorithm 1 Training and Inference with Frame Sparse Attention under Teacher Forcing

Training Procedure:
Input: Clean sequence $\mathbf{Z_0}$, Camera sequence $\mathcal{C}$, Text $\mathcal{Y}$
Noising: Add noise $\mathbf{N}$ to $\mathbf{Z_0}$ to obtain $\mathbf{Z}_t$
$\mathbf{pred} \leftarrow \text{Model}(\mathbf{Z_0}, \mathbf{Z}_t, \mathcal{C}, \mathcal{Y})$
$\mathcal{L} \leftarrow \text{FlowMatching\_loss}(\mathbf{pred}, \mathbf{Z_0}, \mathbf{N})$
Inside Attention:
  Input: Clean $\mathbf{Q}_0$, $\mathbf{K}_0$, $\mathbf{V}_0$, Camera sequence $\mathcal{C}$,
  Input: Noise $\mathbf{Q}_t$, $\mathbf{K}_t$, $\mathbf{V}_t$.
    $\mathbf{Q}_0, \mathbf{Q}_t \leftarrow \text{VR}(\mathbf{Q}_0, \mathcal{C}), \text{VR}(\mathbf{Q}_t, \mathcal{C})$
    $\mathbf{K}_0, \mathbf{K}_t \leftarrow \text{VR}(\mathbf{K}_0, \mathcal{C}), \text{VR}(\mathbf{K}_t, \mathcal{C})$
    $\mathbf{Q}, \mathbf{K}, \mathbf{V} \leftarrow [\mathbf{Q}_0; \mathbf{Q}_t], [\mathbf{K}_0; \mathbf{K}_t], [\mathbf{V}_0; \mathbf{V}_t]$
    Sample indices $\mathcal{I}$ randomly
    $\mathbf{s} \leftarrow \mathbf{q}[\mathcal{I}] \cdot \mathbf{k}[\mathcal{I}]^\top$
    $\mathbf{s} \leftarrow \text{ApplyTeacherForcingMask}(\mathbf{s})$
    $\mathbf{M} \leftarrow \text{TopK}(\text{softmax}(\mathbf{s}))$
    $\mathbf{o} \leftarrow \text{FrameSparseAttention}(\mathbf{q}, \mathbf{k}, \mathbf{v}, \mathbf{M})$

Inference Procedure:
Input: First frame $\mathbf{x}_0$, Camera sequence $\mathcal{C}$, Text $\mathcal{Y}$
for c **in** $\mathcal{C}$ **do**
  Initialize noise $\mathbf{n}_t$
  for $step$ **in** DenoiseSteps **do**
    $\mathbf{pred} \leftarrow \text{Model}(\mathbf{n}_t, c, \mathcal{Y})$
    $\mathbf{n}_t \leftarrow \text{ODE\_Update}(\mathbf{n}_t, \mathbf{pred})$
    Inside Attention:
      Given $\mathbf{q}_t$, $\mathbf{k}_0$, $\mathbf{v}_0$ (Clean KV Cache)
      Sample indices $\mathcal{I}$ randomly
      $\mathbf{q}_c, \mathbf{k}_c \leftarrow \mathbf{q}_t[\mathcal{I}], \mathbf{k}_0[\mathcal{I}]$
      $\mathbf{s} \leftarrow \mathbf{q}_c \cdot \mathbf{k}_c^\top$
      $\mathbf{M} \leftarrow \text{TopK}(\text{softmax}(\mathbf{s}))$
      $\mathbf{o} \leftarrow \text{FrameSparseAttention}(\mathbf{q}_t, \mathbf{k}_0, \mathbf{v}_0, \mathbf{M})$
  end for
  $\mathbf{n}_0 \leftarrow \mathbf{n}_t$
  Model($\mathbf{n}_0, c$, cache=True) {Cache current frame}
end for

A.1 TRAINING CONFIGURATION

We provide detailed training configurations for reproducibility.

Model and Resolution. We build upon WAN 2.2 TI2V-5B (Wan et al., 2025), a text-and-image-to-video diffusion transformer with 5 billion parameters. The training resolution is 480×832 with 61 frames per clip. To convert the model into a streaming video generator, we use teacher-forcing training to align the model with the autoregressive generation interface and KV-caching behavior.

Optimization. We train with a batch size of 64 for 6k steps, using the AdamW optimizer with learning rate 5×10^{-5} and linear warmup for 50 iterations. The training process takes approximately 2 days on 16 NVIDIA A100 GPUs.

Training Data. The training data consists of three sources:

- **Context-as-Memory** (Yu et al., 2025a): $\sim$760k frames of yaw-only camera motion sequences.
- **GF-Minecraft** (Yu et al., 2025b): $\sim$4M frames of gameplay videos with diverse actions.
- **ViewBench**: $\sim$500k frames of synthetic sequences with complete 3-axis rotation coverage.

To obtain a balanced distribution of camera poses and scene geometries, we adjust the sampling rate of each dataset to achieve a 1:1:1 ratio during training.

Fair Comparison. All RoPE variants (3D RoPE, GTA, ViewRope) are applied to the same channels of the latent representation to ensure fair comparison. The channel allocation follows the ablation study in Section 4.4, where ViewRope is embedded in the lowest frequency bands of the temporal dimension (channels 32–44).

B LOOP CLOSURE FORMULATION

This section provides the formal definition of loop-closure consistency referenced in Section 3.1.

Standard video generators enforce *local temporal coherence* via objectives of the form $\mathcal{L}_{\text{temp}}(\theta) := \mathbb{E}_{\mathbf{x}\sim p_\theta}\left[\sum_{t=2}^{T} d(\mathbf{x}_t, \mathbf{x}_{t-1})\right]$, which only constrain adjacent frames and do not prevent long-horizon geometric drift.

A pose-conditioned world model must additionally satisfy loop closure. Define the revisit indicator

$$w_{t,k} := \mathbb{I}(\Delta(\mathcal{C}_t, \mathcal{C}_k) \le \varepsilon), \qquad k < t, \tag{13}$$

where $\Delta(\cdot,\cdot)$ measures pose similarity and ε is a tolerance threshold.

Let $\mathcal{W}_{k\leftarrow t}$ denote the image warp induced by the relative camera motion and scene geometry. For a pixel $\mathbf{u} = (u, v)$ with homogeneous coordinate $\tilde{\mathbf{u}} = [u, v, 1]^\top$, and a depth field D_t at time t:

$$\mathbf{X}_t(\mathbf{u}) := \mathbf{K}_t^{-1}\tilde{\mathbf{u}}\, D_t(\mathbf{u}), \tag{14}$$

$$\mathbf{X}_k(\mathbf{u}) := \mathbf{R}_k\mathbf{R}_t^{-1}\mathbf{X}_t(\mathbf{u}) + \big(\mathbf{P}_k - \mathbf{R}_k\mathbf{R}_t^{-1}\mathbf{P}_t\big), \tag{15}$$

$$\mathcal{W}_{k\leftarrow t}(\mathbf{u}) := \pi\big(\mathbf{K}_k\, \mathbf{X}_k(\mathbf{u})\big), \tag{16}$$

where $\pi(\cdot)$ denotes perspective division. The loop closure loss $\mathcal{L}_{\text{lc}}(\theta)$ enforces $\mathbf{x}_t(\mathbf{u}) \approx \mathbf{x}_k(\mathcal{W}_{k\leftarrow t}(\mathbf{u}))$ for $\mathbf{u} \in \Omega_{t,k}$ (the mutually visible region). We formalize this as:

$$\mathcal{L}_{\text{lc}}(\theta) := \mathbb{E}_{\mathbf{x}\sim p_\theta}\left[\sum_{t=1}^{T}\sum_{k<t} w_{t,k} \sum_{\mathbf{u}\in\Omega_{t,k}} \rho(\mathbf{x}_t(\mathbf{u}) - \mathbf{x}_k(\mathcal{W}_{k\leftarrow t}(\mathbf{u})))\right], \tag{17}$$

where $\rho(\cdot)$ is a robust penalty (e.g., Huber).

C VIEWBENCH BENCHMARK

This appendix provides additional details on ViewBench.

C.1 DATASET COMPARISON

Table 3 compares ViewBench with existing datasets.

Table 3: **Comparison with existing datasets.** ViewBench fills gaps in evaluating view-consistency for interactive world models.

Property	**CaM**	**GF-MC**	**ViewBench**
Yaw	✓	✓	✓
Pitch	×	✓	✓
Roll	×	×	✓
Loop-closure trajectories	×	×	✓
Controlled angle magnitudes	×	×	✓
Per-frame SE(3) c2w	partial	×	✓
Overlap annotations	FOV-based	×	depth-based
Engine	UE5	MC	UE5

C.2 SCENE ENVIRONMENTS

ViewBench comprises 10 photorealistic UE5 environments spanning indoor, outdoor, urban, and natural settings (Table 4). The diversity in geometry, lighting, and texture ensures evaluation results generalize across visual conditions.

C.3 CAMERA ROTATION RANGES

ViewBench trajectories are organized into two parts.

Part 1: Pure rotation. The camera is stationary and performs rotation-only motion covering all 7 axis combinations (3 single-axis + 3 dual-axis + 1 triple-axis). Each trajectory follows a *rotate-away-rotate-back* loop-closure design, with rotation magnitudes sampled from $\{30°, 75°, 90°, 180°\}$. For each of 10 scenes $\times$ 7 axis combinations, we generate 100 samples, yielding $\sim$7,000 clips.

Table 4: **ViewBench scenes.**

Scene	Type	Description
Abandoned_HongKong	Outdoor/Urban	Mid-scale urban ruins
Abandoned_Mall	Indoor	Two-floor shopping mall
DeadCity	Outdoor/Urban	Derelict city, dark lighting
PostApocalypticCity	Outdoor/Urban	Large-scale post-apocalyptic
FPS_Template	Outdoor/Desert	Middle-Eastern battlefield
Container_Yard	Outdoor/Industrial	Container stacking yard
Rome	Outdoor/Historical	Roman-style architecture
SuburbsCityPack	Outdoor/Suburban	Suburban street scene
ChineseAlley	Outdoor/Cultural	Chinese-style alleyway
UrbanDistrict_Gate	Indoor-like	Narrow shantytown alleys

Part 2: Rotation + translation. The camera moves within a compact exploration radius while simultaneously rotating, mimicking interactive navigation. Each sequence is composed of actions randomly drawn from four types: **RotateOnly** (in-place camera rotation), **MoveOnly** (pure WASD translation without rotation), **MoveAndRotate** (simultaneous translation and rotation), and **Orbit** (circular motion around a point of interest). All action types may include roll rotation with type-specific probabilities and ranges.

C.4 Data Format

Each frame is annotated with a 4×4 camera-to-world (c2w) SE(3) matrix, Euler angles (pitch, roll, yaw), position in centimeters, FOV (including the vertical one and the horizontal one), and binary WASD key states. The rotation matrix follows a ZYX convention ($\mathbf{R} = \mathbf{R}_z(\text{yaw}) \cdot \mathbf{R}_y(\text{pitch}) \cdot \mathbf{R}_x(\text{roll})$) in UE5's left-handed coordinate system (X-Forward, Y-Right, and Z-Up). A post-processing pipeline converts ViewBench data into the action formats used by CaM and GF-Minecraft, enabling unified evaluation. Depth-based frame overlap annotations are also provided for attention recall analysis.

C.5 Training and Evaluation Splits

Training. The training set combines Part 1 (pure rotation) and Part 2 (rotation + translation) data, totaling 1,059 video sequences ($\sim$500k frames at 30 fps) across 10 scenes. During training, ViewBench data is mixed with CaM and GF-Minecraft at a 1:1:1 sampling ratio. The evaluation set is *separately collected* from the training set with non-overlapping trajectories, ensuring no data leakage between training and evaluation.

Evaluation. The evaluation set consists of separately collected Part 1 pure-rotation loop-closure trajectories from the same 10 scenes, totaling 600 samples. Each sample is downsampled to 16 fps and contains 61 frames, providing a first frame, the full camera trajectory with per-frame SE(3) poses, and ground-truth UE5-rendered video. We evaluate frame-level PSNR, SSIM, and LPIPS against the ground truth, as well as Loop Closure Error (LCE):

$$\text{LCE} = \text{LPIPS}(\mathbf{x}_0, \hat{\mathbf{x}}_T), \tag{18}$$

where $\mathbf{x}_0$ is the ground-truth first frame and $\hat{\mathbf{x}}_T$ is the generated frame at the return pose. LCE isolates the challenge of remembering previously seen content after an extended camera excursion.

C.6 Complete Baseline Results on ViewBench

Table 5 presents results including large-angle (90°, 180°) trajectories.

Extended analysis of large-angle performance. At rotation magnitudes of 90° and above, HY-WorldPlay outperforms ViewRope in LCE despite using a distilled 4-step model (Table 5). Two system-level factors contribute to this gap. First, fitting large-angle round-trip trajectories into fixed-length evaluation sequences increases the per-frame angular step well beyond what our model encounters during training, causing under-rotation at inference. HY-WorldPlay addresses this through

Table 5: **Complete baseline results on ViewBench** across all rotation magnitudes. Best in **bold**.

Angle	Model	PSNR↑	SSIM↑	LPIPS↓	LCE↓
30°	Matrix-Game-2	14.27	0.2806	0.5723	0.5553
	HY-WorldPlay	17.04	0.4238	0.4697	0.4811
	ViewRope (ours)	**17.53**	**0.4378**	**0.4080**	**0.4497**
45°	Matrix-Game-2	13.55	0.2535	0.6071	0.6175
	HY-WorldPlay	16.38	0.4085	0.5096	0.4944
	ViewRope (ours)	**16.74**	**0.4130**	**0.4527**	**0.4545**
75°	Matrix-Game-2	13.46	0.2625	0.6084	0.6288
	HY-WorldPlay	15.19	0.3847	0.5643	0.5151
	ViewRope (ours)	**15.27**	**0.3916**	**0.5398**	**0.4562**
90°	Matrix-Game-2	12.41	0.1878	0.6684	0.7121
	HY-WorldPlay	**16.56**	**0.4174**	**0.4970**	**0.4169**
	ViewRope (ours)	15.61	0.4001	0.5382	0.5445
180°	Matrix-Game-2	12.74	0.2033	0.6310	0.6732
	HY-WorldPlay	**14.82**	0.3403	**0.5978**	**0.4413**
	ViewRope (ours)	14.35	**0.3458**	0.6043	0.5609

variable-frame-rate training and RL-based action post-training. Second, our teacher-forcing setup leads to compounding errors over long autoregressive rollouts, whereas HY-WorldPlay employs context-forcing distillation with self-correction. Notably, both factors are orthogonal to the positional encoding itself: ViewRope consistently improves LCE over GTA under identical training conditions at every angle. Combining ViewRope with self-forcing (Huang et al., 2025b) or RL-based post-training to close this gap at extreme angles is a natural next step.

C.7 Qualitative Comparison

We present qualitative comparisons between Matrix-Game-2.0 (M-G 2.0), HY-WorldPlay (HY-World), and ViewRope (Ours) on ViewBench loop-closure trajectories. Each case shows the input first frame (left) followed by keyframes sampled from the generated video. Arrow icons indicate the camera rotation direction at each keyframe. The camera first rotates away from the starting viewpoint and then returns, forming a closed loop.

Case 1: Yaw + Pitch in an urban street scene (Figure 4). The camera rotates upward and rightward, then reverses back to the starting view. M-G 2.0 produces severe brightness collapse mid-trajectory, losing nearly all scene content in the dark frames. HY-WorldPlay maintains plausible appearance but exhibits geometric drift—building structures shift position upon return. ViewRope preserves both the scene structure and lighting conditions throughout the trajectory, yielding a return frame closely matching the ground truth.

Case 2: Pure yaw in an Asian street scene (Figure 5). The camera pans left and then reverses rightward to return. M-G 2.0 generates quite big hallucination, introducing new scene elements (e.g., yellow trees) that are absent in the ground truth upon return. In this case, both HY-WorldPlay and ViewRope accurately recover the original storefronts and street layout, demonstrating strong long-term spatial memory.

Case 3: Pure pitch in a Roman architecture scene (Figure 6). The camera tilts downward toward the ground and then pitches back up to the starting view. This tests vertical rotation consistency. M-G 2.0 fails catastrophically on the return—the final frame shows a completely different indoor scene with wooden structures, indicating total loss of scene identity. HY-WorldPlay maintains the general scene category but produces blurry architecture and seemingly fails to return. Comparatively, ViewRope faithfully reproduces the arched stone structures visible in the starting frame.

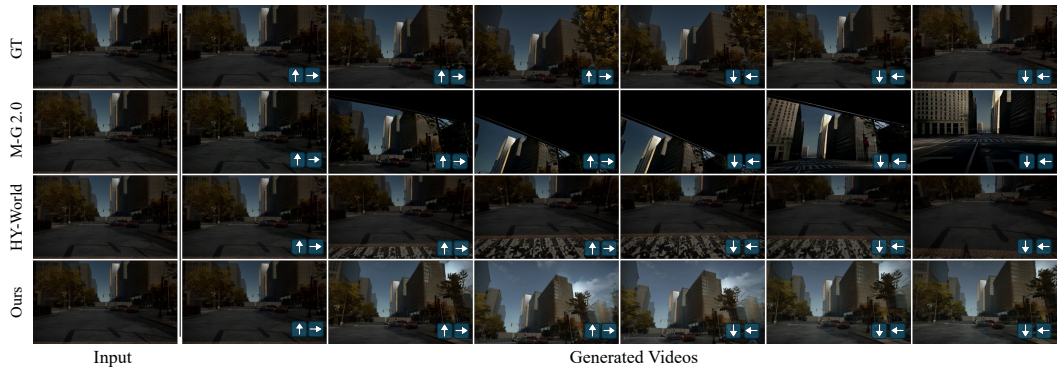


Figure 4: **Case 1: Yaw + Pitch loop closure in an urban street.** M-G 2.0 suffers from brightness collapse. HY-WorldPlay exhibits geometric drift. ViewRope maintains structural and lighting consistency.

Figure 5: **Case 2: Pure yaw loop closure in an Asian street.** M-G 2.0 hallucinates entirely different content on return. HY-WorldPlay introduces nonexistent elements. ViewRope recovers the original scene faithfully.

D Additional Results

D.1 Ablation of Number of Topk Frames

As shown in Table 6, we report the ablation results of the number of topk frames on ViewBench. We use the model trained with topk=5 and adjust the number of retrieved frames at inference time to 1, 3, 10, and 20. Increasing the number of retrieved frames generally improves visual quality metrics (PSNR, SSIM, and LPIPS), suggesting that accessing more reference frames provides richer texture details for generation. However, we observe that the Loop Closure Error (LCE) achieves its optimum at topk=5 and degrades as the number of frames increases further. This indicates that while more context helps visual quality, the model, having been trained with topk=5, may be distracted by the additional retrieved frames or struggle to effectively utilize the expanded context for long-term geometric consistency.

D.2 Ablation of ViewRope Channel Allocation

Efficiently integrating ViewRope into the existing 3D RoPE architecture is a key design challenge. The original model partitions RoPE into Temporal (T), Height (H), and Width (W) components, occupying 44, 42, and 42 dimensions respectively, totaling 128 dimensions. We investigate four strategies for embedding ViewRope: (1) Embedding in the lowest frequency bands of the T dimension (channels 32–44); (2) Embedding in the lowest frequency bands of H and W dimensions (channels 74–86 and 116–128); (3) Embedding in the lowest-frequency bands of H and W, while disabling the

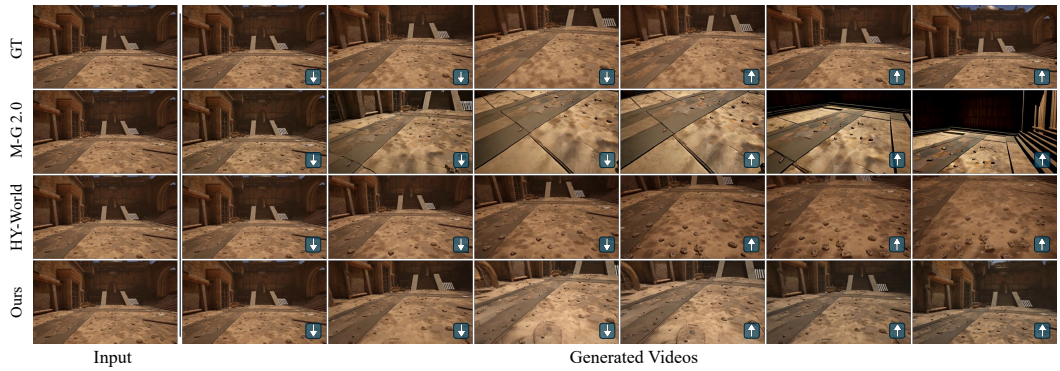


Figure 6: **Case 3: Pure pitch loop closure in Roman architecture.** M-G 2.0 generates a completely different scene upon return. HY-WorldPlay produces blurry, inconsistent structures. ViewRope accurately restores the original arched architecture.

Table 6: **Ablation of number of topk frames on ViewBench.** We report visual quality (PSNR↑, SSIM↑, LPIPS↓) and geometric consistency (LCE↓) for 90° and 180° view synthesis. Best in **bold**.

Top@k	90 deg				180 deg			
	PSNR	SSIM	LPIPS ↓	LCE ↓	PSNR	SSIM	LPIPS ↓	LCE ↓
Top 1	13.01	0.3328	0.8230	0.8416	13.00	0.3337	0.8357	0.8109
Top 3	15.11	0.3792	0.5470	0.5777	14.54	0.3559	0.6096	0.5413
Top 5	15.71	0.3929	0.5267	**0.5591**	14.92	0.3574	0.5991	**0.5385**
Top 10	15.53	0.3951	0.5386	0.5804	**14.99**	0.3548	0.5997	0.5956
Top 20	**16.17**	**0.4087**	**0.5240**	0.5860	14.92	**0.3576**	**0.5976**	0.5976

corresponding 3D RoPE components to prevent interference; (4) Distributing ViewRope across all dimensions (1–128).

Table 7: **Ablation: ViewRope embedding strategies.** Embedding in T-dimension low frequencies yields the best performance.

Embedding Strategy	Training Loss↓
T-dim low-freq (ch 32–44)	**0.0859**
H/W-dim low-freq (ch 74–86, 116–128)	0.0861
H/W-dim low-freq (replace 3D RoPE)	0.0874
All dimensions (ch 1–128)	0.0894

As shown in Table 7, embedding in the lowest frequency bands of the T dimension yields the best performance (0.0859), suggesting that the temporal dimension offers the most suitable capacity for encoding view information without disrupting spatial representations governed by H and W. Replacing the original 3D RoPE with ViewRope leads to degradation (0.0874 vs. 0.0861), indicating that original positional information remains complementary to our geometric encoding. Distributing ViewRope across all dimensions results in the highest loss (0.0894), likely due to excessive interference with the backbone's pre-trained frequency structure.

D.3 Extended Position Encoding Comparison

To further validate ViewRope's advantage, we compare against Plücker (Zhang et al., 2024) and PRoPE (Li et al., 2025b) under an extended training budget (10K base steps + 6K fine-tuning steps), longer than the 6K steps used for the main results in Table 1. All methods in this table are trained from the same checkpoint under identical conditions. We also evaluate a combined configuration (PRoPE + Per-Patch) that applies both per-camera projective encoding and per-patch ray rotation.

Key findings: (1) Per-patch ray rotation is the key differentiator—adding it improves LCE in both families (ViewRope vs. GTA: 8.9%↓ at 75°; PRoPE+Per-Patch vs. PRoPE: 11.2%↓). (2) All RoPE-family methods outperform Plücker-style additive conditioning. (3) ViewRope and PRoPE are com-

Table 8: **Extended position encoding comparison on ViewBench (LCE↓).** Per-Patch and Per-Camera indicate encoding granularity. Best in **bold**.

Method	Granularity		LCE ↓	
	Per-Patch	Per-Camera	30°	75°
Plücker			0.4683	0.4552
GTA		✓	0.4716	0.4462
PRoPE		✓	0.4517	0.4529
ViewRope (Ours)	✓	✓	0.4419	0.4063
PRoPE + Per-Patch	✓	✓	**0.4187**	**0.4022**

plementary: combining them achieves the best LCE, yet ViewRope alone nearly matches this combination (0.4063 vs. 0.4022), showing that per-patch granularity provides the dominant geometric benefit.

D.4 Robustness to Noisy Camera Parameters

To evaluate robustness under imprecise camera estimation (e.g., from SfM), we add Gaussian noise ($\sigma \in \{1°, 3°, 5°\}$) to camera rotation Euler angles at inference time with frozen model weights.

Table 9: **Robustness to noisy camera on ViewBench (synthetic).**

Noise σ	30° PSNR↑	30° LCE↓	75° PSNR↑	75° LCE↓
0° (clean)	18.20	0.4333	15.89	0.4034
1°	18.22	0.4322	15.92	0.4027
3°	18.13	0.4226	15.87	0.4142
5°	18.12	0.4381	15.87	0.4145

ViewRope is remarkably robust: even at $\sigma = 5°$, PSNR drops by only 0.08 dB with LCE virtually unchanged. This robustness stems from two factors: (1) ViewRope encodes *relative* ray geometry ($\mathbf{R}_i^{-1}\mathbf{R}_j$), so per-frame perturbations largely cancel; (2) the diffusion prior provides natural smoothing. State-of-the-art SfM (e.g., COLMAP) typically achieves rotation errors well below 1°, firmly within ViewRope's robust regime.