SCENEQUERY: TARGET-QUERIED SCENE MEMORY FOR VISUOSPATIAL VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Visuospatial video generation emerges as a new task to synthesize videos that satisfy target-object or camera-trajectory instructions while remaining consistent with the already observed environment. Although existing methods condition generation on scene observations, how to retrieve task-relevant scene evidence for individual output positions remains under-explored. We introduce SceneQuery, a task-conditioned scene retrieval framework in which task instructions guide not only what to generate, but also what scene evidence to retrieve. SceneQuery constructs global and spatial memory from RGB observations and derives a query for each output token using target-object semantics and token coordinates for grounding, or the corresponding target-camera ray for navigation. Unlike queries derived from evolving target features, these task-derived queries remain fixed throughout denoising, providing a stable, output-aligned retrieval signal. The retrieved evidence conditions video generation, while the memory and retrieval modules are jointly optimized through the generation objective without additional memory supervision or explicit 3D reconstruction. Controlled query-source comparisons support task-derived retrieval across both tasks. On ScanNet++, selected configurations improve target-view reconstruction for navigation and reduce geometric error for grounding relative to a matched context-guided baseline, while the primary joint grounding-success rate remains unchanged.

1 INTRODUCTION

Video generative models have made substantial progress in synthesizing realistic visual sequences (Wan Team et al., 2025). Recent work has extended these models to visuospatial tasks: given RGB observations of an environment, a model generates a video that approaches a specified object or follows a requested camera trajectory (Xiao et al., 2026), as illustrated in Figure 1. Such generation requires following the instruction while preserving the objects and layout of the observed environment. To do so, the model must use the observations to determine where the requested object is and what scene content should be visible as the camera moves.

Existing approaches make scene observations available to video generators through context conditioning or explicit memory retrieval. History conditioning (Song et al., 2025) and context compression (Zhang et al., 2025) enable generation from past observations, while Video4Spatial (Xiao et al., 2026) jointly conditions video generation on sparse scene observations and task instructions for object grounding and scene navigation. More recent methods introduce explicit memory mechanisms to organize and retrieve scene information, including frame selection based on camera field-of-view overlap (Yu et al., 2025a), surfel-indexed spatial retrieval (Li et al., 2025b), and learnable queries for extracting information from context memory (Yu et al., 2026). However, in instruction-driven visuospatial generation, the relevance of scene information depends not only on the observed context but also on the generation objective and the corresponding output position. Object grounding requires evidence about a specified target, whereas scene navigation requires information relevant to the desired camera viewpoint. Even within a single task, different output positions may depend on different parts of the observed scene. This raises a fundamental question: how can task-specific generation requirements be translated into output-aligned memory queries that selectively retrieve relevant scene evidence?

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.

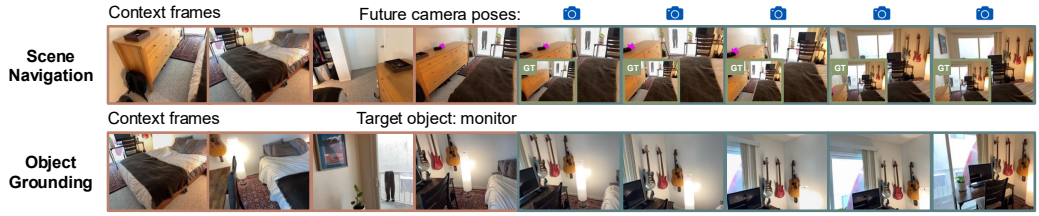


Figure 1: **Video-based spatial tasks.** Given context frames, the model generates a video following target camera poses (top) or approaching a specified object (bottom). Blue borders mark generated frames; insets show ground truth.

We argue that task instructions should guide not only what to generate, but also what scene evidence to retrieve. Motivated by this insight, we propose SceneQuery, a task-conditioned scene retrieval framework that explicitly constructs memory queries from the generation objective and output positions. Rather than deriving retrieval queries from the evolving target features, SceneQuery determines them directly from task-specific conditions, keeping them fixed throughout the denoising process. This decouples memory retrieval from the changing generation state and provides a stable, task-aligned conditioning signal. Specifically, for object grounding, each query combines the semantic description of the target object with the spatiotemporal position of the corresponding output token. For scene navigation, each query encodes the target-camera ray associated with that position. Despite their different conditioning modalities, both tasks employ a shared query-memory interface to retrieve scene evidence aligned with individual output tokens.

To realize this formulation, SceneQuery organizes the observed scene into a queryable memory representation. Specifically, learned global memory slots aggregate information across observations, while spatial memory retains localized visual features from individual views. An additional initial-view reference preserves fine-grained details around the starting viewpoint. Given the task-derived queries, SceneQuery retrieves evidence from both the scene memory and the initial-view reference through cross-attention, and fuses the resulting features to provide output-aligned conditioning. The retrieved evidence is then incorporated into the video diffusion Transformer through a learned residual pathway, complemented by task-specific feature modulation. The memory construction, query encoding, and retrieval modules are jointly optimized using the video generation objective, without additional memory supervision, depth information, or explicit 3D reconstruction.

We evaluate SceneQuery on object grounding and scene navigation. Controlled query-source comparisons show that task-derived queries outperform queries formed from evolving target features under the same memory organization and with the task adapter disabled, supporting the proposed retrieval design across both tasks. Against a matched context-guided Video4Spatial backbone, selected SceneQuery configurations improve target-view reconstruction for navigation and reduce geometric error for grounding, while leaving the primary joint grounding-success rate unchanged. Further ablations examine how memory organization and conditioning strength affect these outcomes.

Our main contributions are as follows:

- We introduce task-conditioned scene retrieval for visuospatial video generation, constructing queries from task instructions and output positions independently of evolving target features.
- We develop a unified scene-memory framework for object grounding and scene navigation, combining global and spatial memory with task-specific queries to retrieve output-aligned scene evidence, without additional memory supervision or explicit 3D reconstruction.
- We evaluate the proposed retrieval design through controlled query-source comparisons across both tasks and characterize its benefits and limitations through backbone comparisons and ablation studies.

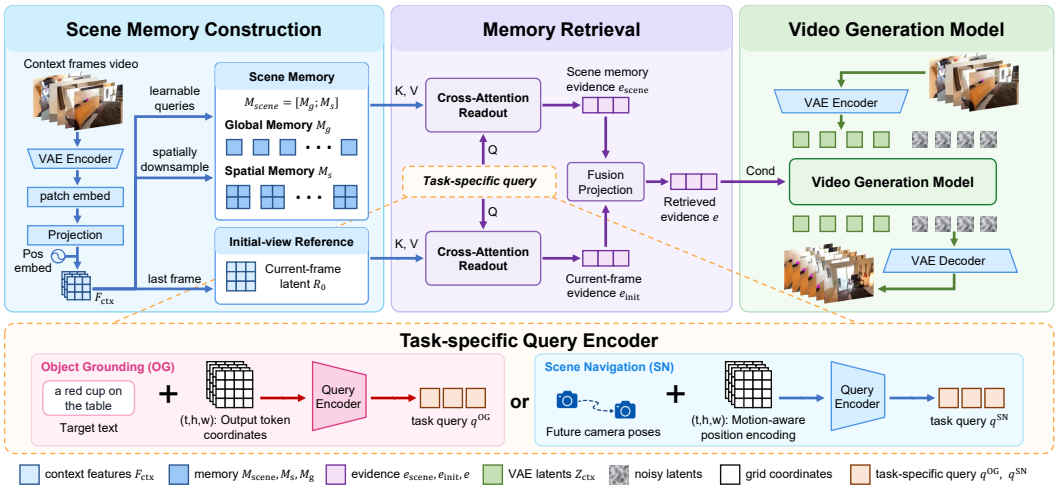


Figure 2: **Overview of SceneQuery.** Scene memory construction encodes context observations into global and spatial memory and retains an initial-view reference. Task-specific encoders construct stable queries aligned with output tokens: grounding queries combine target text with output positions, while navigation queries encode target-camera rays. The queries retrieve scene and initial-view evidence, which is fused to condition video generation.

2 Related Work

Spatial Video Generation. Navigation World Models predict observations under navigation controls (Bar et al., 2025), while History-Guided Video Diffusion supports flexible history conditioning (Song et al., 2025). FramePack compresses frame history within a bounded context budget for next-frame video prediction (Zhang et al., 2025). Video4Spatial jointly processes context and target tokens for object grounding and scene navigation from sparse RGB observations (Xiao et al., 2026). Unlike localization in reconstructed scenes (Li et al., 2025a), these tasks require generating scene-consistent videos that reach an object or follow a trajectory. SceneQuery adds explicit task-conditioned scene retrieval to this setting.

Scene Memory. Memory-based video generators retrieve observations to maintain scene consistency. Context-as-Memory and WorldMem select history using camera states (Yu et al., 2025a; Xiao et al., 2025), while VMem indexes views with 3D surfels (Li et al., 2025b). MemCam combines context compression with co-visibility-based retrieval (Gao et al., 2026); Memorize When Needed couples camera-based retrieval with a separate memory branch and gated injection (Guo et al., 2026). SceneQuery queries RGB feature memory with fixed, task-derived queries per output token, without context-camera poses, 3D indexing, or dependence on noisy target features.

Scene Representations. SRT renders novel views through ray queries over latent scene representations (Sajjadi et al., 2022). AnySplat predicts 3D Gaussians from unposed images (Jiang et al., 2025), while VolSplat predicts voxel-aligned Gaussians (Wang et al., 2025). MotionCtrl and CameraCtrl encode camera trajectories as video generation conditions (Wang et al., 2024; He et al., 2024). GEN3C conditions generation on rendered 3D caches (Ren et al., 2025); TrajectoryCrafter combines point-cloud renders and source videos to redirect camera trajectories (Yu et al., 2025b). CineScene instead injects pretrained 3D-aware features (Huang et al., 2026). SceneQuery uses ray and semantic queries to condition video denoising through a shared feature memory, without 3D reconstruction or a pretrained geometry encoder.

3 Method

SceneQuery queries a scene representation with task instructions to guide video generation at each output position. Figure 2 shows three stages: constructing scene memory from context observations, reading it with task queries, and using the retrieved evidence to condition generation.

3.1 Problem Formulation and Overview

Let $X_{\text{obs}} = \{x_j^{\text{obs}}\}_{j=1}^{N_c}$ denote ordered RGB observations sparsely sampled from one scene. They need not be contiguous, and their camera poses are not provided. Following Video4Spatial Xiao et al. (2026), an initial target frame x_0 is appended to form $X_{\text{ctx}} = [X_{\text{obs}}; x_0]$. The task instruction $\mathcal{I}_\tau$ specifies a target object in text for object grounding (OG), or a camera trajectory and intrinsics for scene navigation (SN). Starting from x_0, the generated video should approach the specified object for OG or follow the requested camera trajectory for SN. The conditional generation distribution is

$$X_{\text{out}} \sim p_{\theta_\tau}(\cdot \mid X_{\text{ctx}}, \mathcal{I}_\tau), \qquad \tau \in \{\text{OG}, \text{SN}\}. \tag{1}$$

A video encoder $\mathcal{E}$ maps the context and target videos to latents, $Z_{\text{ctx}} = \mathcal{E}(X_{\text{ctx}})$ and $Z_{\text{out}} = \mathcal{E}(X_{\text{out}})$. Noise is added to the target latents, while the context latents remain clean. A video diffusion Transformer predicts the flow velocity conditioned on the context and task instruction:

$$\begin{aligned} Z_{\text{out},t} &= (1-t)Z_{\text{out}} + t\epsilon, & \epsilon &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \\ \widehat{v} &= v_{\theta_\tau}(Z_{\text{out},t}, t, Z_{\text{ctx}}, \mathcal{I}_\tau), & t &\in [0,1]. \end{aligned} \tag{2}$$

Inference integrates the predicted velocity from Gaussian noise at $s = 1$ to $s = 0$, then decodes the resulting latents. The memory pathway supplies evidence at each target position:

$$M_{\text{scene}} = \text{Write}(X_{\text{ctx}}), \quad q_i^\tau = \text{Query}_\tau(\mathcal{I}_\tau, u_i), \quad e^\tau_{\text{scene},i} = \text{Read}(M_{\text{scene}}, q_i^\tau), \tag{3}$$

here u_i is the spatiotemporal position of target token i.

3.2 Scene Memory Construction

Global memory summarizes the scene, while spatial memory retains local features. Both use tokens obtained by jointly encoding the observations and appended initial target frame x_0 with the video VAE. The backbone's patch embedding maps the resulting latents Z_{ctx} to an $T_c \times H_c \times W_c$ token grid, where T_c counts temporal positions and H_c, W_c are its spatial dimensions. A memory projection maps these tokens to d_m dimensions. Before attention aggregation or spatial pooling, we add an MLP embedding of each token's context-grid coordinates (t, h, w), with each axis normalized to $[-1, 1]$. These coordinates encode temporal order and spatial location within the encoded context grid, rather than 3D scene coordinates. The resulting context features are $F_{\text{ctx}} \in \mathbb{R}^{T_c \times H_c \times W_c \times d_m}$.

The final temporal position supplies the initial-view reference, while the preceding positions $F_{\text{obs}} = \{F_{\text{ctx},j}\}_{j=1}^{T_c-1}$ supply global and spatial memory.

Global memory. Global memory uses N_g learned slot queries $S \in \mathbb{R}^{N_g \times d_m}$ to aggregate information across the observed views. Each slot attends to all scene tokens, with S as queries and F_{obs} as both keys and values:

$$\begin{aligned} \widetilde{M}_{\text{g}} &= \text{LN}\left(S + \text{Attn}(S, F_{\text{obs}}, F_{\text{obs}})\right), \\ M_{\text{g}} &= \widetilde{M}_{\text{g}} + \text{FFN}(\widetilde{M}_{\text{g}}). \end{aligned} \tag{4}$$

The result is N_g memory tokens of dimension d_m. These tokens summarize information across context frames for later retrieval by task queries.

Spatial memory. Spatial memory retains features from local regions of the observations. At each non-reference temporal position, average pooling reduces the context tokens to an $H_s \times W_s$ grid; concatenating these grids forms spatial memory:

$$M_{\text{s}} = \text{Concat}\left[\text{Pool}(F_{\text{ctx},1}), \cdots, \text{Pool}(F_{\text{ctx},T_c-1})\right]. \tag{5}$$

This gives $(T_c-1)H_sW_s$ memory tokens, each summarizing a spatial region at one context temporal position. The grid indexes regions in the encoded views rather than locations in a reconstructed 3D scene. Task queries can therefore retrieve local visual features alongside the scene summary provided by global memory.

Global and spatial tokens receive distinct learned type embeddings and token-wise normalization before forming $M_{\text{scene}} = [M_{\text{g}}; M_{\text{s}}]$.

The unpooled final temporal grid forms $R_0 = F_{ctx,T_c} \in \mathbb{R}^{H_c \times W_c \times d_m}$ after reference-type embedding and token-wise normalization. Joint temporal encoding retains initial-view detail and preceding context.

3.3 TASK-SPECIFIC QUERY CONSTRUCTION

Both tasks assign one query per target token on the $T_q \times H_q \times W_q$ grid after VAE encoding and patchification, yielding $Q^\tau \in \mathbb{R}^{N_q \times d_m}$ with $N_q = T_q H_q W_q$. These queries combine task instructions with output positions to request evidence specific to each token.

Object grounding queries. For object grounding, each query combines the requested object's semantics with its target token's position. The pretrained text encoder and backbone projection encode the description; averaging valid text tokens gives the shared feature b_{obj}. Target-grid coordinates form a sequence $U = [u_1, \cdots, u_{N_q}]^\top \in \mathbb{R}^{N_q \times 3}$, with one triplet $u_i = (t_i, h_i, w_i)$ per output token and each axis normalized to $[-1, 1]$. Separate MLPs map the shared text feature and each coordinate triplet to d_m dimensions. The target-position MLP has independent parameters from the context-position MLP used in memory construction. The query is

$$q_i^{\text{OG}} = \text{LN}\left(\lambda_{\text{text}} P_{\text{text}}(b_{\text{obj}}) + \lambda_{\text{pos}} P_{\text{pos}}(u_i)\right), \tag{6}$$

The weights λ_{text} and λ_{pos} balance object semantics and output position, so each query requests object-relevant evidence for its corresponding token.

Scene navigation queries. For scene navigation, each query encodes the viewing geometry associated with its target token. Target camera poses and intrinsics define a ray at each video pixel: pixels in a frame share a camera origin o but have distinct viewing directions d. Each ray is encoded as $[o \times d; d]$ using Plücker coordinates Sitzmann et al. (2021). Temporal packing and a spatial adapter align this ray field with the target token grid, giving a feature $\mathbf{a}_i$ at each position. Projection to d_m dimensions and normalization form the query:

$$q_i^{\text{SN}} = \text{LN}(P_{\text{ray}}(a_i)), \tag{7}$$

These queries request scene evidence along the target viewing rays, without requiring known 3D surface points. Following Video4Spatial Xiao et al. (2026), navigation training expresses poses relative to a randomly selected zero-pose reference frame; this leaves the initial-view memory R_0 unchanged.

Both query sequences follow the target token order and share the retrieval mechanism below. Derived from instructions and positions, they remain independent of noisy target features and fixed throughout denoising.

3.4 QUERY-BASED SCENE RETRIEVAL

Each task query reads from scene memory M_{scene} and the initial-view reference R_0. The two sources provide information across the observations and finer detail around the starting view, respectively. Separate cross-attention modules return one feature from each source:

$$e^\tau_{\text{scene},i} = \text{Attn}(q_i^\tau, M_{\text{scene}}, M_{\text{scene}}), \tag{8}$$

$$e^\tau_{\text{init},i} = \text{Attn}(q_i^\tau, R_0, R_0), \tag{9}$$

The two readouts are fused with a decreasing reference weight, giving the initial view more influence near the start of the generated video:

$$w_i = w_{\text{end}} + (1 - w_{\text{end}})(1 - \rho_i), \qquad \rho_i \in [0, 1], \tag{10}$$

where ρ_i is the normalized target temporal position and w_{end} is the final reference weight. A projection maps the weighted sum to the retrieved evidence:

$$e_i^\tau = P_{\text{fuse}}\left(\lambda_{\text{scene}} e^\tau_{\text{scene},i} + w_i e^\tau_{\text{init},i}\right), \tag{11}$$

Here λ_{scene} controls the contribution of scene memory relative to the temporally weighted initial-view reference. The resulting e_i^τ is the scene evidence supplied to output token i during generation.

3.5 MEMORY-CONDITIONED VIDEO GENERATION

The retrieved evidence e_i^τ guides generation at its corresponding output position. Residual injection adds this evidence to the target features, while feature modulation adapts them to the requested object or view. Both updates are applied before each Transformer block.

Memory evidence integration. Memory residual injection projects e_i^τ into the backbone feature space. Its contribution to target token i at block ℓ is

$$\Delta\phi_{R,i}^{\ell} = r_\tau g_R^{\ell} P_{\text{out}}(e_i^\tau), \tag{12}$$

where P_{out} is shared across blocks and the learned gate g_R^ℓ controls the evidence contribution at each block.

The task adapter modulates the current target feature ϕ_i^ℓ according to its output condition c_i^τ. This condition fuses the evidence e_i^τ and query q_i^τ for grounding, or uses the aligned ray feature a_i for navigation. A gated FiLM module Perez et al. (2018) predicts the channel-wise scale, shift, and gate from this condition to produce

$$\Delta\phi_{F,i}^{\ell} = f_\tau g_F^{\ell}\, \text{FiLM}_\tau\left(\text{LN}(\phi_i^\ell), c_i^\tau\right), \tag{13}$$

where g_F^ℓ is a learned block gate. FiLM predicts its modulation values for each position, using $\tanh$ for scale and shift and a sigmoid for the channel gate.

The fixed scales (r_τ, f_τ) set the strengths of residual injection and feature modulation, with the same values used during training and inference. The two updates are added to the target feature before it enters block ℓ:

$$\widetilde{\phi}_i^{\ell} = \phi_i^\ell + \Delta\phi_{R,i}^{\ell} + \Delta\phi_{F,i}^{\ell}, \tag{14}$$

Here h_i^ℓ includes the backbone's camera-control update when used. The Transformer processes the updated target tokens with the context, and its final output head predicts the denoising velocity $\widehat{v}$.

Training objective. Scene memory and task-conditioned retrieval are jointly optimized under the video generation objective. For each task τ, the backbone's flow-matching loss supervises the velocity predicted from the noisy target latents and conditioning inputs:

$$\mathcal{L}_\tau = \mathbb{E}_{X_{\text{out}}, X_{\text{ctx}}, \mathcal{I}_\tau, t, \epsilon}\left[\left\|v - v_{\theta_\tau}(Z_{\text{out},t}, t, Z_{\text{ctx}}, \mathcal{I}_\tau)\right\|_2^2\right]. \tag{15}$$

where $v = \epsilon - Z_{\text{out}}$ is the target velocity for the interpolation in Equation 2. This loss jointly optimizes the memory writer, query encoder, memory reader, and task adapter, together with LoRA Hu et al. (2022) adaptation of the backbone. Because the retrieved evidence conditions the velocity prediction, the generation loss supervises both what the memory retains and how task queries read it, without additional memory supervision or a 3D reconstruction objective.

The memory-injection and task-adapter branches have zero-initialized output layers and small nonzero initial block gates, so both branches initially contribute zero updates.

4 Experiments

4.1 Experimental Setup

Implementation Details. We fine-tune Wan2.2-TI2V-5B (Wan Team et al., 2025) with rank-32 LoRA on attention and feed-forward layers using a learning rate of 10^{-4}. Training uses 168 scene observations plus one initial view as context; generation outputs 161 frames at 416×256 resolution. We train separate models for object grounding (OG) and scene navigation (SN) tasks. All experiments use flow-matching objectives and fixed training budgets across runs.

Dataset and Evaluation. We evaluate on scene-disjoint ScanNet++ (Yeshwanth et al., 2023) splits. For OG, we generate text instructions describing target objects and their spatial locations within scenes. For SN, we directly use ScanNet++ with 1K videos and camera pose annotations, randomly sampling short clips for training. For both tasks, we evaluate on 20 randomly selected scenes from ScanNet++ that are excluded from the training set, with 128 validation samples per task configuration and fixed generation seeds for reproducibility.

Metrics. For OG, we report Spatial Distance (SD) for geometric consistency, Instruction Following (IF), and the joint success rate IF_δ, which requires both IF success and SD below threshold δ ($\delta = 0.2$ as primary, $\delta = 0.1$ as stricter). IF uses a vision-language evaluator to check whether the target object's bounding-box center is near the frame center. For SN, we report PSNR, SSIM, and

LPIPS for target-view reconstruction. We additionally report VBench Imaging Quality (IQ) (Huang et al., 2024), Dynamic Degree (DD), and temporal RGB-change discrepancy (T-gap) as perceptual and motion diagnostics. Tables report aggregate means; OG coverage remains unresolved, so rate comparisons are descriptive.

4.2 Main Results

We compare OG with Wan2.2-5B (Wan Team et al., 2025), Veo 3.1 (Google DeepMind, 2025), and FramePack (Zhang et al., 2025), and SN with VolSplat (Wang et al., 2025), AnySplat (Jiang et al., 2025), TrajectoryCrafter (Yu et al., 2025b), and Gen3C (Ren et al., 2025); both use Video4Spatial (Xiao et al., 2026) as the matched baseline.

Table 1(a) separates geometric consistency from instruction response. Relative to the matched baseline, the selected model reduces SD by 21.2%, while the primary joint success rate $IF_{0.2}$ is unchanged. The gain therefore concerns geometric consistency rather than a uniform improvement in object-grounding success. The nonzero GT SD and imperfect GT IF also expose reconstruction and evaluation limitations.

Table 1(b) shows improved target-view reconstruction over the matched baseline, with higher PSNR and lower LPIPS. These measurements do not independently establish camera-trajectory accuracy or an explicit 3D scene model.

Table 1: **Main results.** Object grounding and scene navigation on ScanNet++. GT is an evaluator reference. Res. = Resolution, Ctx. = Context frames, Gen. = Generation frames. 3D info. denotes context geometry beyond the target trajectory; P+D denotes pose and depth.

Method	Res.	Ctx.	Gen.	SD ↓	IF ↑	$IF_{0.2}$ ↑	IQ ↑	DD
GT	416×256	-	161	0.11	100.00	100.00	65.79	87.50
Wan2.2-5B	1280×704	1	121	0.52	75.78	10.94	50.72	96.88
Veo 3	1280×720	1	192	0.41	79.69	28.91	62.49	97.66
FramePack	704×544	104+1	180	0.90	16.41	3.13	49.51	100.00
Video4Spatial	416×256	168+1	161	<u>0.22</u>	**48.62**	**36.00**	<u>55.15</u>	93.75
SceneQuery	416×256	168+1	161	**0.18**	<u>43.12</u>	**36.00**	**55.23**	92.97

Method	3D	PSNR ↑	SSIM ↑	LPIPS ↓	IQ ↑
VolSplat	Pose	9.88	0.32	0.77	33.00
AnySplat	Pose	9.77	0.40	0.76	48.61
TrajectoryCrafter	P+D	10.14	0.38	0.76	43.89
Gen3C	P+D	9.68	0.37	0.77	43.77
Video4Spatial	None	<u>11.08</u>	<u>0.4049</u>	<u>0.6149</u>	<u>54.31</u>
SceneQuery	None	**12.63**	**0.4420**	**0.5213**	**54.39**

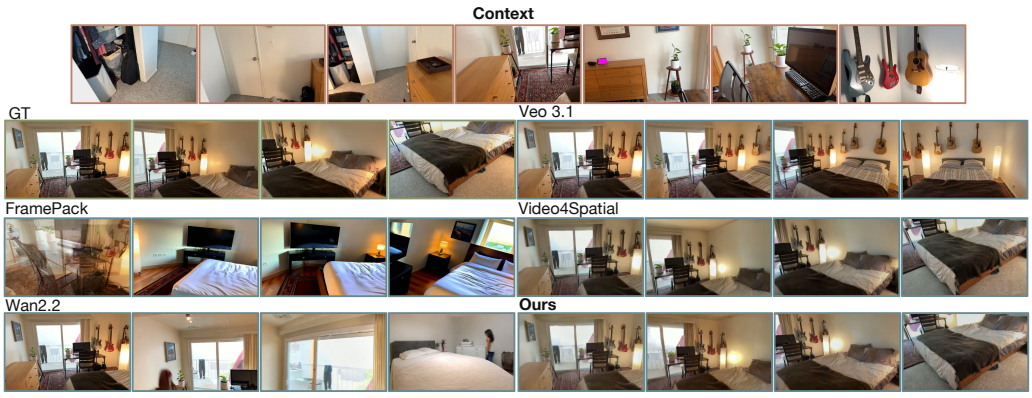


Figure 3: **Qualitative object-grounding comparison.** The top row shows the observed context, and each subsequent row shows the ground-truth or generated trajectory for the same instruction. SceneQuery retains the room layout from the context and approaches the requested object, whereas the competing methods may drift to inconsistent views or hallucinate scene content. For the best experience, see the supplementary videos.

4.3 Ablation Studies

Following Video4Spatial (Xiao et al., 2026), we fix the training context at $\mathrm{ctx} = 169$ and vary inference context without retraining. Table 2 reports the values plotted for the selected full checkpoints.

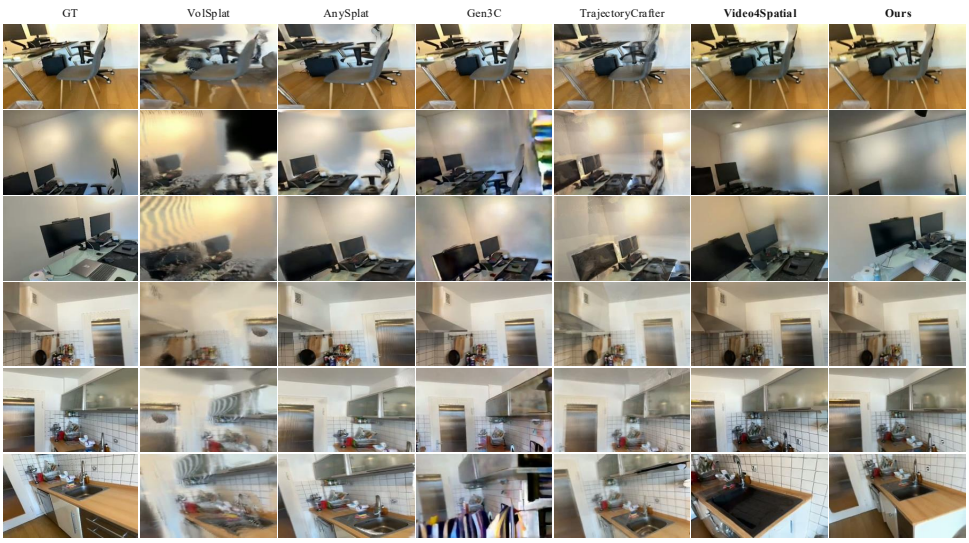


Figure 4: **Qualitative scene-navigation comparison.** Each column shows a trajectory-aligned frame for the ground-truth video, competing methods, Video4Spatial, and SceneQuery. SceneQuery preserves the observed scene content while following the requested camera motion, with fewer spatial distortions than the competing video-generation and 3D-reconstruction baselines.

Scene memory construction. Table 3(a) compares memory organization strategies. Combining global and spatial memory gives the strongest OG geometric consistency and is competitive for SN reconstruction, but it does not dominate every metric. Memory organization therefore affects task objectives differently, rather than providing a uniformly best representation.

Task-specific query construction. Table 3(b) isolates the query design by comparing task-derived queries with queries formed from the current noisy target hidden state, while keeping memory organization fixed and disabling the task adapter. Task queries improve both tasks, reducing OG SD and SN LPIPS. This controlled comparison supports constructing queries from task instructions and output positions rather than from evolving noisy features.

Initial-view reference integration. Table 3(c) examines how the initial-view reference is integrated on top of global–spatial memory. The decaying reference improves the main OG geometry and SN reconstruction measures, but the constant reference remains better on some perceptual scores. The strategy therefore introduces a useful temporal bias without improving every metric.

Memory-conditioned video generation. Table 3(d) examines the contributions of the memory pathway and task adapter. The memory pathway includes scene memory construction, task-query-based retrieval, and residual injection; the task adapter performs feature modulation conditioned on retrieved evidence (OG) or aligned ray features (SN). All configurations retain the backbone's original context and instruction inputs.

For SN, the full model improves reconstruction over the memory-only configuration, supporting the usefulness of feature modulation for navigation.

For OG, memory-only gives stronger instruction following, whereas adding the task adapter improves geometric alignment at the cost of semantic responsiveness. The two branches therefore expose a task-dependent trade-off rather than a uniform component gain.

Table 2: **Inference context length.** Ctx counts RGB frames including the initial view; $169^{\dagger}$ is the training context.

Ctx.	OG SD ↓	SN LPIPS ↓
45	0.2660	0.6083
85	0.2481	0.5693
$169^{\dagger}$	0.2421	0.5213
337	0.2555	0.4670
673	0.2957	0.4784

Table 3: **Ablation study.** (a) Memory organization. (b) Query construction, without task adaptation. (c) Initial-view reference integration. (d) Added generation branches. Enabled branches use scale 1; full rows use the default configuration. Bold/underline indicate best/second-best within each panel.

Configuration	Object grounding				Scene navigation				
	SD ↓	IF ↑	IQ ↑	DD	PSNR ↑	SSIM ↑	LPIPS ↓	T-gap ↓	IQ ↑
(a) Scene memory construction									
Uniform pooling	0.2214	51.65	52.39	94.53	10.4079	0.3740	0.6513	4.4928	**56.50**
Global only	0.2080	**61.11**	53.80	89.84	11.8073	0.4181	0.5740	3.4807	55.31
Spatial only	0.1855	61.82	54.74	82.03	11.5677	0.4146	0.5932	3.7052	55.12
Global + spatial	**0.1888**	58.11	53.77	92.19	11.5637	0.4063	0.5768	3.4043	55.93
(b) Task-specific query construction									
Noisy-state query	0.2024	53.79	**53.04**	95.31	11.8759	0.4206	0.5717	3.2945	52.36
Task query	**0.1889**	**58.71**	52.11	93.75	**12.3382**	**0.4267**	**0.5278**	**3.1048**	**56.81**
(c) Initial-view reference integration									
Constant reference	0.1888	48.88	**57.91**	86.72	11.5637	0.4063	0.5768	3.4043	**55.93**
Decaying reference	**0.1764**	**58.11**	53.77	**92.19**	**11.8454**	**0.4255**	**0.5731**	**3.2714**	51.29
(d) Memory-conditioned video generation									
w/o memory	0.2238	41.73	55.15	93.75	11.08	0.4049	0.6149	3.9562	54.31
w/o task adaptation	**0.1889**	**58.71**	52.11	**93.75**	12.3382	0.4267	0.5278	3.1048	56.81
Full model (selected)	0.1764	43.12	**55.23**	92.97	**12.6297**	**0.4420**	**0.5213**	**3.1386**	**54.39**

These results indicate that the two tasks benefit from different module configurations. Scene navigation benefits from combining memory retrieval with feature modulation, while object grounding benefits primarily from memory-based evidence and uses the adapter for geometric refinement with a semantic trade-off.

Choosing integration strength. Table 4 varies the memory-injection and task-adapter scales (r, f) using the same values during training and inference. The selected scales differ by task, showing that the balance between evidence injection and feature modulation affects generation quality; this sweep does not isolate the task adapter's incremental effect.

Random camera reference. Navigation training follows Video4Spatial by randomly selecting the zero-pose reference frame; Table 5 reports the resulting change.

Table 4: **Integration scales.** Full-model sweep over memory-injection and task-adapter scales.

	Object grounding					Scene navigation				
(r, f)	SD ↓	IF ↑	$\text{IF}_{0.2}$ ↑	IQ ↑	DD	PSNR ↑	SSIM ↑	LPIPS ↓	T-gap ↓	IQ ↑
$(1, 1)$	0.2520	32.37	26.20	52.37	93.75	11.8454	0.4255	0.5731	3.2714	51.29
$(0.25, 1)$	0.2720	36.70	26.00	54.87	92.19	**12.6297**	**0.4420**	**0.5213**	**3.1386**	54.39
$(0.5, 1)$	0.2185	42.54	32.26	55.02	96.09	11.3658	0.4060	0.5933	3.8687	50.29
$(1, 0.25)$	**0.1764**	**43.12**	**36.00**	**55.23**	92.97	10.4586	0.3676	0.6661	4.4107	**56.21**
$(1, 0.5)$	0.2292	39.15	26.42	53.60	96.09	11.3381	0.3975	0.6010	3.7476	53.25

Table 5: **Camera-reference randomization for SN.**

Method	PSNR ↑	SSIM ↑	LPIPS ↓	T-gap ↓	IQ ↑
Fixed reference	11.4814	0.4126	0.5927	3.5661	**52.31**
Random reference	**11.8454**	**0.4255**	**0.5731**	**3.2714**	51.29

5 Conclusion

SceneQuery uses task conditions as explicit addresses for retrieving scene evidence. Task-derived queries outperform noisy-state queries in both tasks. Selected configurations improve navigation reconstruction and grounding geometry relative to the matched baseline, but not the primary joint grounding-success rate. Memory organization and integration strength remain task- and metric-dependent, supporting task-conditioned scene access without implying uniform component gains. The context-length results likewise show that additional observations help navigation reconstruction but do not uniformly improve grounding.

AI USE STATEMENT

In this work, we used generative AI tools for assisting with translation. We have not used generative AI tools for generating synthetic data sets, helping develop theoretical models or conceptual frameworks, designing or providing feedback on research methodology or experiments, cleaning and reformating dataset, supporting qualitative and thematic data analysis or interpreting results. Formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses and implementing methods are not applicable to this work. Additionally, we used generative AI tools for sourcing/searching for information. We have reviewed all AI-assisted work.

REFERENCES

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15791–15801, 2025.

Xinhang Gao, Junlin Guan, Shuhan Luo, Wenzhuo Li, Guanghuan Tan, and Jiacheng Wang. MemCam: Memory-augmented camera control for consistent video generation. *arXiv preprint arXiv:2603.26193*, 2026.

Google DeepMind. Veo 3.1, 2025. URL `https://deepmind.google/models/veo/`.

Yanjun Guo, Zhengqiang Zhang, Pengfei Wang, Xinyue Liang, Zhiyuan Ma, and Lei Zhang. Memorize when needed: Decoupled memory control for spatially consistent long-horizon video generation. *arXiv preprint arXiv:2604.18215*, 2026.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. CameraCtrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *International Conference on Learning Representations (ICLR)*, 2022.

Kaiyi Huang, Yukun Huang, Yu Li, Jianhong Bai, Xintao Wang, Zinan Lin, Xuefei Ning, Jiwen Yu, Yu Wang, and Xihui Liu. CineScene: Implicit 3d as effective scene representation for cinematic video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25381–25392, 2026.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. VBench: Comprehensive benchmark suite for video generative models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. AnySplat: Feed-forward 3d gaussian splatting from unconstrained views. In *ACM SIGGRAPH Asia Conference Proceedings*, 2025.

Rong Li, Shijie Li, Lingdong Kong, Xulei Yang, and Junwei Liang. SeeGround: See and ground for zero-shot open-vocabulary 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3707–3717, 2025a.

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

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3942–3951, 2018.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. GEN3C: 3d-informed world-consistent video generation with precise camera control. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Mehdi S. M. Sajjadi, Henning Meyer, Etienne Pot, Urs Bergmann, Klaus Greff, Noha Radwan, Suhani Vora, Mario Lucic, Daniel Duckworth, Alexey Dosovitskiy, Jakob Uszkoreit, Thomas Funkhouser, and Andrea Tagliasacchi. Scene representation transformer: Geometry-free novel view synthesis through set-latent scene representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://arxiv.org/abs/2111.13152.

Vincent Sitzmann, Semon Rezchikov, William T. Freeman, Joshua B. Tenenbaum, and Frédo Durand. Light field networks: Neural scene representations with single-evaluation rendering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Kiwhan Song, Boyuan Chen, Max Simchowitz, Yilun Du, Russ Tedrake, and Vincent Sitzmann. History-guided video diffusion. In *International Conference on Machine Learning*, 2025.

Wan Team, 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.

Weijie Wang, Yeqing Chen, Zeyu Zhang, Hengyu Liu, Haoxiao Wang, Zhiyuan Feng, Wenkang Qin, Zheng Zhu, Donny Y. Chen, and Bohan Zhuang. VolSplat: Rethinking feed-forward 3d gaussian splatting with voxel-aligned prediction. *arXiv preprint arXiv:2509.19297*, 2025. URL https://arxiv.org/abs/2509.19297.

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 Conference Proceedings*, 2024.

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

Zeqi Xiao, Yiwei Zhao, Lingxiao Li, Yushi Lan, Ning Yu, Rahul Garg, Roshni Cooper, Mohammad H Taghavi, and Xingang Pan. Video4Spatial: Towards visuospatial intelligence with context-guided video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3933–3944, 2026.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A high-fidelity dataset of 3d indoor scenes. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Jiwen Yu, Jianhong Bai, Yiran Qin, Quande Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Context as memory: Scene-consistent interactive long video generation with memory retrieval. In *SIGGRAPH Asia 2025 Conference Papers*, 2025a. doi: 10.1145/3757377.3763833.

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

Mark Yu, Wenbo Hu, Jinbo Xing, and Ying Shan. TrajectoryCrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *IEEE/CVF International Conference on Computer Vision*, 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. In *Advances in Neural Information Processing Systems*, 2025.