Video World Models with Long-term Spatial Memory

Tong Wu[*1], **Shuai Yang**[*2,4], **Ryan Po**[1], **Yinghao Xu**[1],
Ziwei Liu[5], **Dahua Lin**[3,4], **Gordon Wetzstein**[1]

[1] Stanford University   [2] Shanghai Jiao Tong University   [3] The Chinese University of Hong Kong
[4] Shanghai Artificial Intelligence Laboratory   [5] S-Lab, Nanyang Technological University
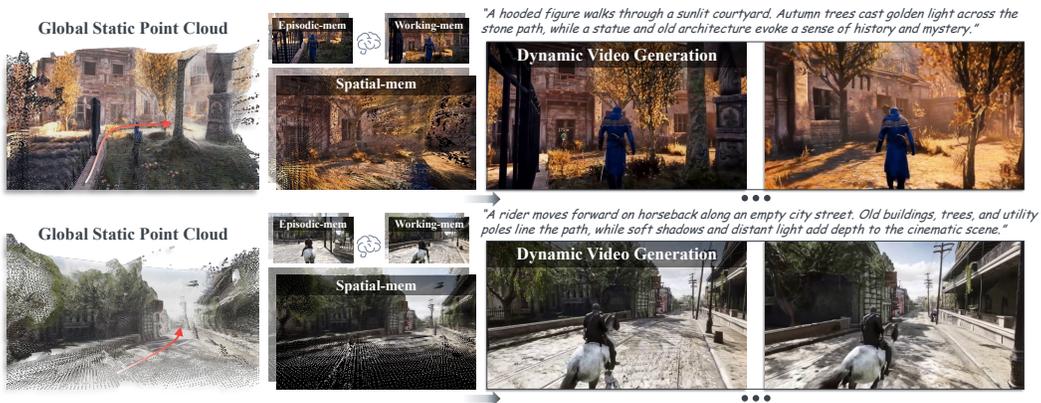
`https://spmem.github.io/`

Figure 1: We augment video world models with memory. In this context, we consider the conventional approach of conditioning autoregressively generated frames with a few recent context frames as a short-term working memory. We explore two additional mechanisms modeling different types of long-term memory: spatial and episodic memory. The former is represented as a point map that is autoregressively generated along with the video frames and fused into the spatial memory by extracting only its static scene parts. To remember visual detail and identities for long time horizons, we also store a sparse set of historical reference frames as an episodic memory. Together, our memory mechanisms significantly improve the long-term consistency of emerging video world models.

Abstract

Emerging world models autoregressively generate video frames in response to actions, such as camera movements and text prompts, among other control signals. Due to limited temporal context window sizes, these models often struggle to maintain scene consistency during revisits, leading to severe forgetting of previously generated environments. Inspired by the mechanisms of human memory, we introduce a novel framework to enhancing long-term consistency of video world models through a geometry-grounded long-term spatial memory. Our framework includes mechanisms to store and retrieve information from the long-term spatial memory and we curate custom datasets to train and evaluate world models with explicitly stored 3D memory mechanisms. Our evaluations show improved quality, consistency, and context length compared to relevant baselines, paving the way towards long-term consistent world generation.

1   Introduction

World models are generative systems that learn to predict an environment in response to actions, making them well suited for simulating complex, interactive settings [28, 2, 30, 74, 90]. Video diffusion models [11, 37, 44, 79, 55] have emerged as a powerful approach to architecting world models, especially when used with autoregressive next-frame prediction [1, 12, 18, 22, 41, 53, 60, 65, 73, 81, 35]. Existing video generation models, however, often struggle with long-horizon consistency due to limited temporal context windows, frequently forgetting previously seen scenes during revisits. This is due to the relatively small number of previously generated context frames that the model can consider when generating new frames—a problem primarily caused by the quadratic growth of computational complexity in the attention module of the underlying diffusion transformers.

To address this challenge, current world models simply keep the number of context frames low to maintain computational feasibility. Several very recent approaches explore progressive downsampling of temporally more distant frames to increase the temporal context window size [25, 86]. Yet, all these approaches rely on image-based representations of the past and lack a persistent 3D understanding of the world, limiting spatial consistency.

Inspired by the mechanisms of human memory [3], we propose a new framework to enhance the long-term consistency of video world models through long-term spatial memory grounded in geometry. Drawing from cognitive theories, our approach incorporates three distinct forms of memory—***spatial, working, and episodic***—each modeled through a dedicated representation. Similar to existing models, our framework relies on a set of recently generated context frames. We consider this a *short-term working memory* mechanism, making both static and dynamic aspects of the most recent past accessible in the form of pixel data. To help remember long-term spatial relationships, we introduce an additional *long-term spatial memory*. This mechanism primarily relies on an explicit 3D representation of the generated world, augmented by a *sparse episodic memory* in the form of a set of keyframes from the past. We implement the spatial memory using a geometry-grounded point cloud representation. Before storing newly generated information into this memory mechanism, we filter out dynamic parts of the world to primarily remember the static parts of the work in this memory.

The primary contribution of our work is the design of our memory mechanism, combining short-term working memory with long-term spatial and sparse episodic memory, as illustrated in Figure 1. We develop approaches to store newly generated information into the spatial and episodic memory bank as well as retrieve information from it to effectively condition the generation of new video frames. Moreover, we curate a custom dataset to train and evaluate a proof-of-principle implementation of the proposed mechanisms. Our evaluations show that the quality and 3D consistency of our approach surpasses that of relevant baselines. With this work, we hope to contribute to the growing community effort of unlocking infinite-length, consistent world generation capabilities for computer graphics, robotics, and other interactive applications.

2   Related work

We consider a world model to be a generative system that autoregressively generates image or video frames, conditioned on actions or camera pose. This work builds on several important concepts in the generative AI literature, which we briefly review in the following.

Image and video generation.   In recent years, diffusion models have emerged as the state-of-the-art paradigm for image generation [31], surpassing the performance of GAN-based methods in both fidelity and diversity by modeling complex distributions through iterative denoising [45, 51, 64, 67, 20]. These successes have naturally extended to the task of video generation, as diffusion-based architectures have been adapted to include the temporal domain, enabling the generation of high-quality video clips spanning the order of tens to hundreds of frames [89, 48, 78, 56, 21, 14, 8]. Additional advancements in efficiency and controllability have also followed, through techniques such as improved tokenization [63, 75] and flow-matching objectives [49, 40].

Autoregressive video generation.   To support generation of longer videos with online inference capabilities, autoregressive (AR) approaches and architectures have been introduced in the diffusion framework [62, 43, 76, 71]. The autoregressive regime is also particularly intuitive, given the natural temporal ordering of video data. Taking inspiration from LLMs, state-of-the-art methods model

video data by processing frames into spatio-temporal tokens [48, 89, 78, 63, 21, 56, 14], learning to autoregressively generate new tokens until a full video is formed. This is in stark contrast with conventional diffusion-based frameworks, which choose to iteratively denoise an entire image/video at once [9, 46, 37, 58, 79, 44, 29, 52]. Recent approaches include training conditional diffusion models that generate new frames given a set of previous clean frames, allowing AR generation simply by passing newly generated frames as context for future frames [37, 34, 16, 41, 88, 24, 23]. Others modify the diffusion objective by assigning independent noise levels for each frame during training, which also supports AR inference [13, 80, 60]. In practice, these methods can be used for infinite-length generations by utilizing a sliding window context, but suffers from limited memory and drift.

Controlled video generation. Controlled or conditional video generation is a core component of world simulators. A series of works have explored explicit camera control, including [72, 32, 33, 33, 5, 4], which enable novel-view or multi-view generation by disentangling camera trajectories from dynamic content. Most of these approaches take either a single image or a text prompt as input, and often struggle to maintain long-term consistency, especially in dynamic scenes. More recent efforts [85, 7, 6] focus on re-filming or generating synchronized views in dynamic settings, further advancing controllability. Beyond direct camera injection, structural conditioning via point clouds, tracking, or 3D-aware priors has proven effective for improving spatial consistency and trajectory alignment [83, 57, 26, 82, 15]. In addition to spatial and motion-level control, some models support action-based or scene-level conditioning [54, 19], where either structured action vocabularies or segmented text prompts serve as high-level drivers of video progression.

Long-context video generation. While advances in autoregressive methods have allowed for generation of longer, and even potentially inifinite-length videos [14], the reliance on sliding context windows limit their effective memory. The straightforward approach of increasing memory by training with longer context lengths is effective but computationally demanding. Recent works have explored efficient architectural alternatives to attention-based transformers, such as state-space models and linear attention [69, 50]. While efficient, such methods often perform worse than conventional diffusion transformer architectures. Other concurrent works have proposed compressing prior frames to lower context length [86, 25], subsequently lowering computational demand at the cost of information lost in context frames. A separate branch of works ground previously generated frames in a 3D representation (e.g., point clouds [83, 26, 82, 57]) and query this representation based on the camera position of future frames. These methods allow for precise camera control, but struggle to handle dynamic scenes with complex motions. SlowFast-VGen [38] introduces a dual-branch slow–fast learning paradigm to explicitly disentangle long-term temporal dependencies from short-term motion dynamics, enabling more coherent and action-aware long video generation. Complementarily, [17] leverages continual adaptation at inference to maintain temporal consistency over extremely long sequences, demonstrating that lightweight test-time optimization can substantially extend generation duration without retraining.

In this work, we adopt the conventional approach of concatenating recent context frames as working memory in diffusion-based generation to autoregressively generate multiple future frames. Meanwhile, we iteratively predict and filter the static point maps of newly generated frames to update a global spatial memory, which serves as geometric guidance for long-term generation. We further incorporate a sparse set of historical reference frames as an episodic memory.

3 Method

Our framework includes mechanisms for storing new observations into memory and retrieving information from it. Drawing inspiration from three distinct forms of human memory—***spatial, working, and episodic***—we introduce a memory storage mechanism that models each type using a dedicated representation. In the following section, we describe how each memory type is constructed and maintained (Sec. 3.2), how it is integrated into the model as a conditional signal (Sec. 3.3), and how we curate the appropriate data to facilitate efficient learning of these mechanisms (Sec. 3.4).
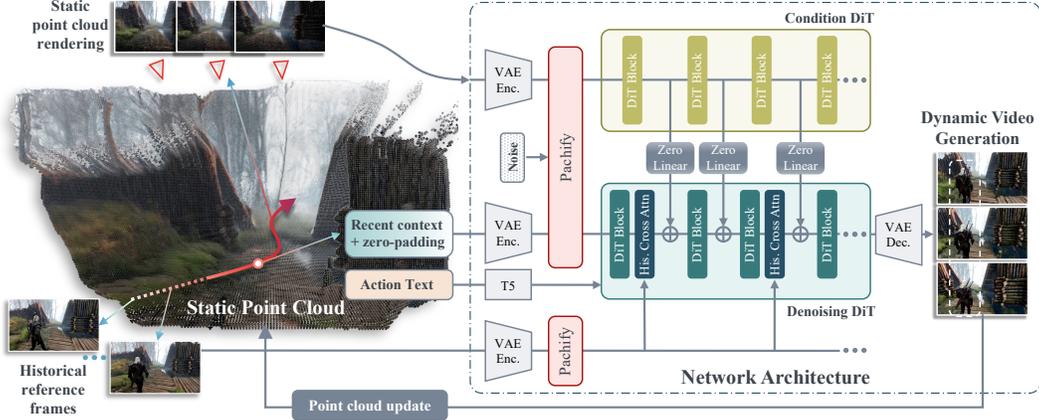
Figure 2: **Overview of our system.** A latent video generation model, implemented by a diffusion transformer (DiT), is conditioned on three different memory mechanisms when autoregressively generating new frames. First, recent context frames model a *short-term working memory*. Second, a point cloud representation (left) is autoregressively generated along with the video frames. This *long-term spatial memory* contains the static parts of the world. Third, a set of historical reference frames (lower left) is stored as a *sparse, long-term episodic memory*. Together, these memory mechanisms enable consistent long-term video generation.

3.1 Preliminaries

Diffusion models [59, 36, 61] learn to model a data distribution by reversing a forward diffusion process denoted by $x_t = \alpha_t x_0 + \sigma_t \epsilon$, where the positive scalar parameters α_t and σ_t determine the signal-to-noise ratio based on a predefined noise schedule, and ϵ is drawn from a standard Gaussian distribution. The diffusion model then learns to predict the noise added through the following denoising objective:

$$\mathcal{L}(\theta) = \mathbb{E}_{t,x_0,\epsilon} \left\| \epsilon_\theta(x_t, t) - \epsilon \right\|_2^2. \tag{1}$$

Video diffusion models commonly operate in a latent space, following a two-stage process: first, input videos are encoded using a 3D variational autoencoder (VAE) [66, 27, 10], and then a diffusion model is trained to model the resulting latent representations. Generation proceeds by sampling within the latent space and decoding the results back to pixel space. Our prototype implementation builds on CogVideoX [79], which adopts this two-stage framework. Specifically, CogVideoX employs a diffusion transformer with 3D attention blocks to capture the distribution of the latent space. Our model improves upon this architecture with additional control signals for enabling long-term memory.

3.2 Spatial Memory Storage

Coherent Static Structure for Spatial Guidance. Human *spatial memory* refers to our ability to encode, store, and retrieve information relating to the physical layout and structure of our environment. To this end, we construct a persistent spatial memory in the form of a long-term static point map that captures high-confidence, temporally consistent 3D structures. To further separate static elements like buildings from dynamic elements like characters and animals, we adopt truncated signed distance function (TSDF) fusion [84]. We denote the TSDF value and associated weight of a voxel v as $D(v)$ and $W(v)$, respectively. Given a new observation from frame i, the voxel update rule follows the standard weighted averaging:

$$D'(v) = \frac{W(v) \cdot D(v) + w_i \cdot d_i(v)}{W(v) + w_i}, \quad W'(v) = W(v) + w_i, \tag{2}$$

where $d_i(v)$ is the truncated signed distance between voxel v and the observed surface in frame i, and w_i is a frame-dependent confidence weight (typically set to 1).

This fusion process inherently filters out dynamic elements in the scene: due to inconsistent depth observations across frames, such voxels accumulate low-confidence, noisy TSDF values and are naturally suppressed in the final fused volume.

4

During the autoregressive generation process, spatial memory is incrementally updated with newly observed static maps, which are reconstructed in an online recurrent manner by CUT3R [70] and filtered by TSDF-Fusion to eliminate the dynamic parts.

Recent Frames for Dynamic Context Guidance.   In humans, ***working memory*** is in charge of temporarily holding information needed for performing reasoning and comprehension tasks. Similarly, our model requires knowledge of nearby previous frames to generate temporally coherent future frames. Drawing on this concept, we incorporate a short-term memory stream based on recent frames, which provides motion continuity atop the static scene structure. We adopt a simple autoregressive generation strategy, where the model generates $N - k$ future frames by conditioning each step on the most recent $k + 1$ latent frames. This procedure can be iteratively applied, enabling open-ended video generation with consistent temporal dynamics.

Representative Historical Slots to Enhance Details.   Human *episodic memory* is a type of explicit memory that stores specific important events from the past, allowing us to "recall" the relevant experiences when needed. While the fused static point cloud captures stable scene geometry, the fused static point cloud is often too sparse to preserve detailed visual cues from the past. To compensate for this, we maintain a set of representative historical frames as auxiliary references. Specifically, during generation, we monitor the size of newly revealed unknown regions via mask-based visibility checks. When the revealed area exceeds a predefined threshold, the corresponding frame is selected and added to the memory set in an incremental fashion.

3.3   Memory-guided Video Generation

Different from those video generation models that focus on camera control under the same temporal sequences [82, 7, 6, 83, 26], we mainly focus on the temporal progression. For example, for a car driving on the road, when the camera moves to the left, the car should complete reasonable dynamic motion in accordance with the hints of the camera language and the input prompt, while maintaining the consistency of the static part to achieve the simulation of the interactive world model. Based on this requirement, we carefully design several key modules to achieve dynamic and static decoupling.

As illustrated in Figure 2, first, we introduce static point cloud rendering as an additional conditioning input to our video diffusion model. The condition video is rendered from the current static spatial memory along the input trajectory, with background regions lacking point clouds set to black. We then utilize the pre-trained 3DVAE [79] to encode the static point cloud rendering into condition latents. We follow a similar design as the ControlNET [87] to add the static point clouds rendering to guide the camera movement and keep the static areas consistency. We copy the first 18 pre-trained DiT blocks from CogVideoX as the condition DiT to process the condition latents. In the condition DiT, we process the output feature from each main DiT block through a zero-initialized linear layer before adding it to the corresponding feature map in the main DiT. Second, to support the generation of new dynamic elements and the temporal extension of existing ones, we propose to concatenate the last five frames of source video tokens with the target video tokens along the frame dimension for dynamic context guidance. In addition, the target condition tokens are also combined with recent context tokens as mentioned above to ensure frame-level correspondence. Third, for modeling information exchange between memory frames and the frames currently being generated, we select the representative historical slots frames as auxiliary reference frames. This reference frames are also encoded by 3DVAE and patchify them as reference tokens. we add a historical cross attention to guide information exchange between the frames currently being generated and memory frames. Specifically, the video tokens act as queries and the reference tokens serve as keys and values.

3.4   Geometry-grounded Video Dataset Creation

To train and evaluate our geometry-grounded video generation model, we require a custom dataset as shown in Figure 3 which is described in the following.

Dataset Construction.   We build our dataset from raw videos collected from MiraData [42], segmenting each video into multiple 97-frame clips. For each clip, the first 49 frames serve as the source sequence and the remaining 48 as the target, with a shared transition frame between the source and target sequence to preserve temporal continuity. To recover scene geometry, we perform
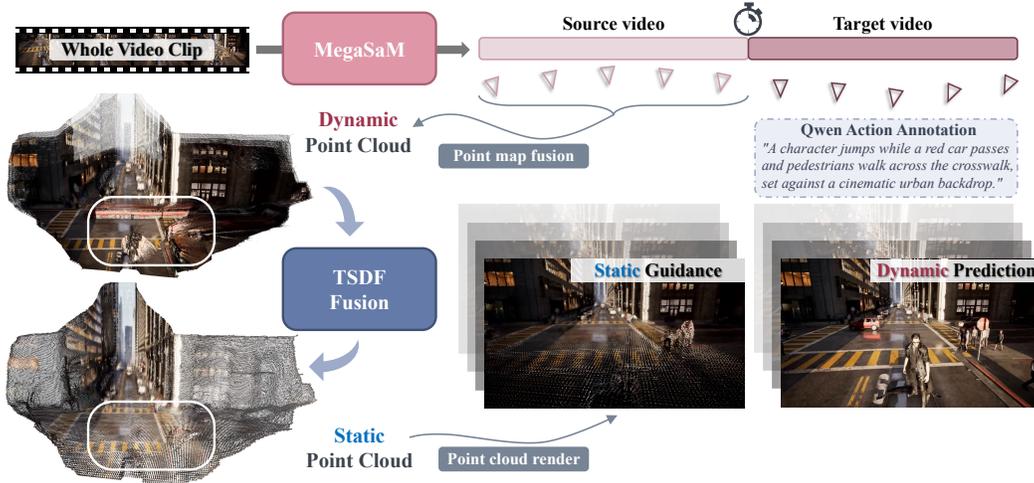
Figure 3: **Dataset construction pipeline.** We use Mega-SaM [47] to extract camera poses and dynamic point maps from the full video clip. For the source part, *dynamic regions are erased* via TSDF-Fusion, and the point cloud is rendered along the target trajectory to to serve as static geometry guidance for the target part. Qwen [77] generates annotations for actions in future target frames.

Table 1: **Quantitative evaluation.** We evaluate our method and baselines using FVD, view recall, and a user study. For view recall, we use standard metrics to compare the consistency of revisiting parts of a scene. The user study provides relative average human ranking scores for three different criteria of all baselines. Our method outperforms these baselines in all cases.

Method	FVD	View Recall Consistency			User Study		
		PSNR ↑	SSIM ↑	LPIPS ↓	Cam-Acc ↑	Stat-Cons ↑	Dyn-Plaus ↑
TrajectoryCrafter	355.23	11.71	0.4380	0.5996	1.6320	1.7802	1.6255
DaS	363.36	12.01	0.4512	0.5874	2.5660	2.4396	2.7033
Wan2.1-Inpainting	280.06	12.16	0.4506	0.5875	2.1760	2.3956	2.2701
Ours	**157.11**	**19.10**	**0.6471**	**0.3069**	**3.6260**	**3.3846**	**3.4011**

4D reconstruction using Mega-SaM [47], extracting camera intrinsics, extrinsics, and per-frame depth maps. We apply TSDF-Fusion to the source frames, integrating RGB-D observations into a volumetric grid. This process suppresses inconsistent depth caused by dynamic objects, yielding a clean reconstruction of the static scene.

Paired Training Data. Given the fused geometry, we project the target camera poses to render visibility masks and static-region reconstructions via point-based rendering. The full RGB frames of the target sequence are retained as future supervision, containing dynamic elements beyond the static scene memory. Our final dataset comprises 90K structured video samples, each paired with explicit 3D spatial memory and future observations.

Our dataset can be downloaded here [1]. Additional details on memory storage and retrieval mechanisms as well as dataset creation are found in the supplement.

4 Experiments

Implementation Details. We implement our conditional video diffusion model based on CogVideoX-5B-I2V [79] architecture, pretrained from DaS [26]. During training, we set the video length to 49 frames with a resolution of 480×720. We trained for 6,000 iterations with a learning rate of 2×10^{-5}, using a mini-batch size of 8 and are conducted on eight NVIDIA-A100 GPUs. At inference time, we adopt the latest 5 historical frames from the recent sequence to enable smooth

---

[1] https://huggingface.co/datasets/ysmikey/spmem_megadata

Table 2: **Quantitative evaluation on VBench**. Our model achieves top overall performance among relevant baselines, as measured by VBench metrics.

Method	Aesthetic Quality ↑	Imaging Quality ↑	Temporal Flickering ↑	Motion Smoothness ↑	Subject Consistency ↑	Background Consistency ↑
TrajectoryCrafter	0.5255	0.6428	0.6160	0.9843	0.8830	0.9227
DaS	0.5635	0.6617	_0.7520_	0.9856	0.9325	0.9494
GEN3C	0.5203	0.5654	0.7179	_0.9882_	0.9178	0.9433
Wan2.1-Inpainting	_0.5661_	**0.6788**	0.6433	0.9868	_0.9357_	**0.9513**
Ours	**0.5835**	_0.6701_	**0.7580**	**0.9886**	**0.9359**	_0.9506_

motion prediction. At each auto-regressive iteration, the point map of the newly generated frames are predicted, aligned, and fused into the historical global point clouds, where we create new point rendering sequence given the aimed camera trajectory.

Metrics and Baselines. Our evaluations primarily focus on baseline methods that use point-map-based conditioning. This includes TrajctoryCrafter [82], DiffusionAsShader (DaS) [26], GEN3C [57], and also the state-of-the-art video generative model Wan2.1 [68]. Our test set includes 500 randomly selected video sequences from MiraData, which are not seen during training. We evaluate each method on FVD, view recall consistency, and general video quality on multiple dimensions. Specifically, 1) For static spatial information, the view recall consistency is evaluated using image reconstruction metrics (i.e., PSNR, SSIM, and LPIPS) for paired frames at the same camera location within a video sequence generated with forward and reversed camera trajectory. 2) For general video quality evaluation, we use the standard FVD and six metrics from Vbench [39] for the visual quality, motion smoothness, and consistency. Moreover, we conduct a user study to further validate the criteria above.

4.1 Quantitative Evaluation

Evaluating View Recall Consistency and Camera Accuracy. To verify that our method benefits from the spatial memory mechanism by maintaining high consistency and accurate camera control when revisiting previously generated parts of the world, we conduct experiments on reversed trajectories, where the same camera pose is visited twice, enabling the construction of paired data for reconstruction metrics. As shown in Table 1, our method achieves significantly improved scores in terms of FVD, PSNR, SSIM, and LPIPS compared to all baselines, owing to our memory mechanisms. It is worth noting, however, that even the PSNR of our method is far from perfect, indicating that remembering each and every visual detail of a complex scene is a very challenging task.

Evaluation on Video Quality. We further utilize VBench to evaluate general video quality across multiple dimensions, as shown in Table 2. Compared with baseline methods, our approach demonstrates better performance in aesthetic quality, reduced temporal flickering, smoother motion, and improved subject consistency. Methods primarily designed for 3D static NVS or 4D NVS within the same temporal sequence perform less effectively on our benchmark, likely due to the sparsity of the retained static point cloud and the presence of large spatial holes. While Wan2.1 surpasses our method in imaging quality due to its advanced backbone, and also achieves a higher score in background consistency, we note that the Wan2.1 inpainting model often fails to follow geometric guidance and tends to generate relatively static scenes, which makes it easier to maintain high background consistency scores.

4.2 Qualitative Evaluation

We conduct qualitative comparisons with other geometry-grounded methods, focusing on three key criteria, as illustrated in Figure 4. First, our method demonstrates superior performance in accurately following camera trajectories, guided by point map rendering. In contrast, the Wan2.1 inpainting model and DaS model often fail, especially under significant camera motion. Second, for view recall consistency, we present pairwise comparisons between two frames generated at the same camera pose but at different points in the sequence. Compared with the baselines, our results exhibit significantly higher consistency in static regions during scene revisits, thanks to the static memory mechanism. Finally, we evaluate the ability to generate new actions based on instructions while incorporating static memory. We focus on the harmonious integration of static and dynamic elements, as well as
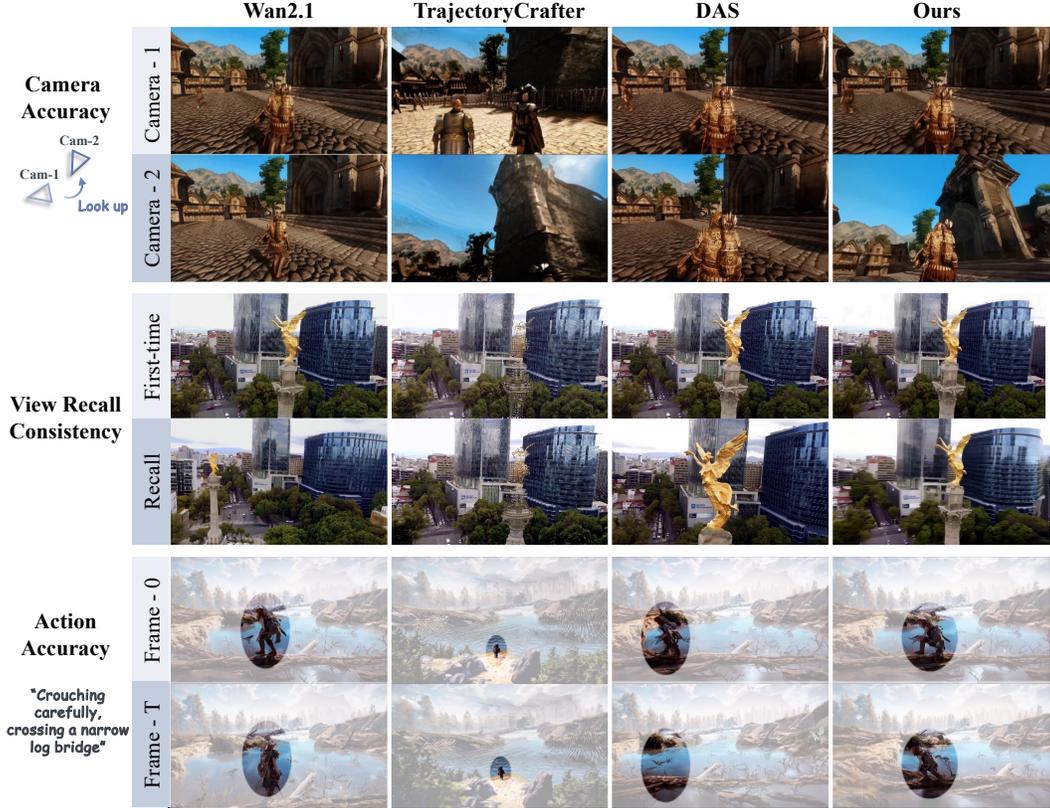
Figure 4: **Qualitative evaluation.** We compare our approach to relevant baselines in several conditions. Baselines cannot accurately generate significant camera pose changes while maintaining a consistent scene (top). When revisiting a previously seen camera pose, baselines fail to complete sparse point clouds or forget details, resulting in inconsistency (center). The accuracy of prompted actions is often low, and sometimes the character disappears during the generation for the baselines (bottom). Our approach successfully handles these challenging scenarios.

how closely the dynamic components follow the instructions. The comparison shows that our method performs well in action prediction, whereas the others either fail to accurately follow the instructions or suffer from action drifting, severe deformation, or even character disappearance.

4.3 User Study

We selected 14 representative use cases, including novel-view synthesis of static scenes, novel-view synthesis of dynamic scenes with temporal progression of dynamic subjects in first-person and third-person perspectives, and scene styles covering realistic and game style. We conduct a user study, evaluating baselines and our methods from three perspectives: camera accuracy(Cam-Acc), static consistency(Stat-Cons), and dynamic plausibility (Dyn-Plaus). We invited 20 subjects, each with at least one year of experience in video/3D/4D generation, to rank the results generated by the four methods (TrajectoryCrafter, DaS, Wan2.1-Inpainting, and ours). Following ControlNet, we evaluate the results using the Average Human Ranking (AHR) metric, where participants rated each output on a scale of 1 to 4 (with lower scores indicating poorer quality). The average rankings in Table 1 (right) show that our method achieves clear and consistent improvements over the baselines across all metrics.

4.4 Ablation Study

To verify the effectiveness of each memory component in our video generation framework, we conduct comprehensive ablation studies, as shown in Table 3. Experimental results on VBench metrics indicate that each component consistently contributes to performance improvements. Unsurprisingly,

8

Figure 5: **Ablation of different memory mechanisms.** We evaluate several variants of our model: **w/o short-term working memory**: the full model without recent context frames; **w/o long-term episodic memory**: the full model without sparse historical keyframes; **full model** including short-term working and long-term spatial and episodic memory. Unsurprisingly, the working memory is required for smooth and plausible motions of dynamic objects. The episodic memory is crucial in helping remember visual details from the past, including previously seen characters or objects.

Table 3: **Ablation of memory mechanisms.** Using all three types, i.e., short-term working and long-term spatial and episodic memory, leads to the best results measured by VBench metrics.

Method	Aesthetic Quality ↑	Imaging Quality ↑	Temporal Flickering ↑	Motion Smoothness ↑	Subject Consistency ↑	Background Consistency ↑
w/o episodic mem	0.5603	0.6485	0.7260	0.9870	0.9326	0.9489
w/o working mem	0.5551	0.6384	0.6740	0.9862	0.9331	0.9453
Full model	**0.5835**	**0.6701**	**0.7580**	**0.9886**	**0.9359**	**0.9506**

the context frames play a crucial role in enhancing short-term motion coherence. During the autoregressive generation process, the context frames enable smooth transitions in dynamic regions and help produce more plausible motions consistent with the preceding frames. The sparse set of historical reference frames enable the model to better retain and utilize temporally distant details. This episodic memory improves long-term consistency for static regions and subjects, and further enhances the plausibility and continuity of motions involving moving entities. The best results are achieved with both of these mechanisms as well as our long-term spatial memory. This is evidenced by both Table 3 and Figure 5, showing that each of our memory mechanisms significantly improves the model's quality in terms of motion smoothness, static consistency, and overall visual quality.

5 Discussion

Inspired by the mechanisms of human memory, we introduce a geometry-grounded long-term spatial memory mechanism for video world models. This mechanism improves quality, spatial consistency, and context length compared to relevant baselines.

Limitations and Future Work. The TSDF-Fusion algorithm we use for storing newly generated information into the spatial memory is far from perfect. Specifically, artifacts are introduced when looking at previously generated content from camera poses that are very different from those of the previous observations, as illustrated in Figure 6. Our memory mechanism is primarily designed to enable *spatial consistency* whereas frame packing strategies for extending the temporal context window size [86] primarily focus on *character consistency*. Future work may combine these mechanisms to achieve both types of consistency. The forgetting problem we tackle is just one of several challenges of video world models. Drift, or image quality degradation due to error accumulation over time, is another challenge that we do not address.

Societal Impacts. Video generation models provide significant benefits for content creation but could be adapted for DeepFake generation. Such applications pose significant societal risks and we strongly oppose the use of our work to create deceptive content intended to mislead or spread misinformation.
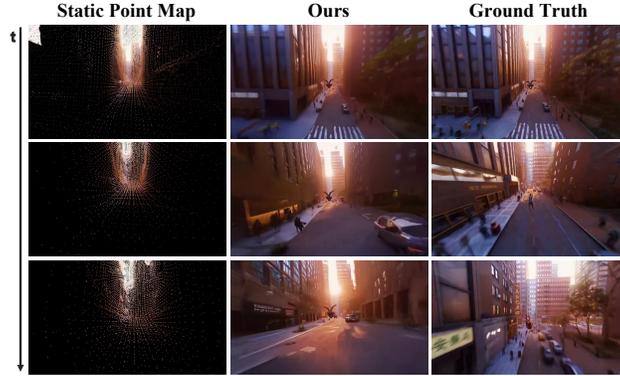
Figure 6: **Failure case.** When the distance between consecutive camera poses is too large and the trajectory exhibits overly abrupt angles, the 4D reconstruction may fail, resulting in significant ghosting artifacts between frames. Consequently, TSDF-Fusion will filter out a large portion of point clouds that should belong to static regions, ultimately leading to an extremely sparse spatial memory and loss of critical information. For example, Spiderman rapidly traversing between skyscrapers illustrates how such a challenging camera trajectory can cause omissions in spatial memory storage, resulting in imprecise camera control and inconsistencies.

Conclusion. Video world models play a crucial role for content creation or creating training data for agents or robots. Enabling long-term consistency through memory mechanisms like ours makes these models more effective.

6 Acknowledgment

References

[1] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos J Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024.

[2] Genesis Authors. Genesis: A universal and generative physics engine for robotics and beyond. *URL https://github. com/Genesis-Embodied-AI/Genesis*, 2024.

[3] Alan Baddeley. *Essentials of human memory*. Psychology Press, 2013.

[4] Sherwin Bahmani, Ivan Skorokhodov, Guocheng Qian, Aliaksandr Siarohin, Willi Menapace, Andrea Tagliasacchi, David B. Lindell, and Sergey Tulyakov. Ac3d: Analyzing and improving 3d camera control in video diffusion transformers. *Proc. CVPR*, 2025.

[5] Sherwin Bahmani, Ivan Skorokhodov, Aliaksandr Siarohin, Willi Menapace, Guocheng Qian, Michael Vasilkovsky, Hsin-Ying Lee, Chaoyang Wang, Jiaxu Zou, Andrea Tagliasacchi, David B. Lindell, and Sergey Tulyakov. VD3d: Taming large video diffusion transformers for 3d camera control. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=On4bSOR5MM.

[6] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, and Di Zhang. Recammaster: Camera-controlled generative rendering from a single video, 2025. URL https://arxiv.org/abs/2503.11647.

[7] Jianhong Bai, Menghan Xia, Xintao Wang, Ziyang Yuan, Zuozhu Liu, Haoji Hu, Pengfei Wan, and Di ZHANG. Syncammaster: Synchronizing multi-camera video generation from diverse viewpoints. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=m8Rk3HLGFx.

[8] A. Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, and Dominik Lorenz. Stable video diffusion: Scaling latent video diffusion models to large datasets. *ArXiv*, abs/2311.15127, 2023. URL `https://api.semanticscholar.org/CorpusID:265312551`.

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

[10] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. URL `https://openai.com/research/video-generation-models-as-world-simulators`.

[11] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1:8, 2024.

[12] Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-x: Interactive open-world game video generation. *arXiv preprint arXiv:2411.00769*, 2024.

[13] Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. In *NeurIPS*, 2024.

[14] Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Juncheng Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengchen Ma, Weiming Xiong, Wei Wang, Nuo Pang, Kang Kang, Zhi-Xin Xu, Yuzhe Jin, Yupeng Liang, Yu-Ning Song, Peng Zhao, Bo Xu, Di Qiu, Debang Li, Zhengcong Fei, Yang Li, and Yahui Zhou. Skyreels-v2: Infinite-length film generative model. 2025. URL `https://api.semanticscholar.org/CorpusID:277856899`.

[15] Luxi Chen, Zihan Zhou, Min Zhao, Yikai Wang, Ge Zhang, Wenhao Huang, Hao Sun, Ji-Rong Wen, and Chongxuan Li. Flexworld: Progressively expanding 3d scenes for flexiable-view synthesis. *arXiv preprint arXiv:2503.13265*, 2025.

[16] Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Seine: Short-to-long video diffusion model for generative transition and prediction. In *ICLR*, 2023.

[17] Karan Dalal, Daniel Koceja, Gashon Hussein, Jiarui Xu, Yue Zhao, Youjin Song, Shihao Han, Ka Chun Cheung, Jan Kautz, Carlos Guestrin, Tatsunori Hashimoto, Sanmi Koyejo, Yejin Choi, Yu Sun, and Xiaolong Wang. One-minute video generation with test-time training, 2025. URL `https://arxiv.org/abs/2504.05298`.

[18] Etched Decart, Q McIntyre, S Campbell, Xinlei Chen, and R Wachen. Oasis: A universe in a transformer. *URL: https://oasis-model. github. io*, 2024.

[19] Julian Decart, Quinn Quevedo, Spruce McIntyre, Xinlei Campbell, Robert Chen, and Wachen. Oasis: A universe in a transformer. 2024. URL `https://oasis-model.github.io/`.

[20] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *ArXiv*, abs/2105.05233, 2021. URL `https://api.semanticscholar.org/CorpusID:234357997`.

[21] Zhengcong Fei, Debang Li, Di Qiu, Jiahua Wang, Yikun Dou, Rui Wang, Jingtao Xu, Mingyuan Fan, Guibin Chen, Yang Li, and Yahui Zhou. Skyreels-a2: Compose anything in video diffusion transformers. 2025. URL `https://api.semanticscholar.org/CorpusID:277509893`.

[22] Ruili Feng, Han Zhang, Zhantao Yang, Jie Xiao, Zhilei Shu, Zhiheng Liu, Andy Zheng, Yukun Huang, Yu Liu, and Hongyang Zhang. The matrix: Infinite-horizon world generation with real-time moving control. *arXiv preprint arXiv:2412.03568*, 2024.

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

[24] Kaifeng Gao, Jiaxin Shi, Hanwang Zhang, Chunping Wang, Jun Xiao, and Long Chen. Ca2-vdm: Efficient autoregressive video diffusion model with causal generation and cache sharing. *arXiv preprint arXiv:2411.16375*, 2024.

[25] Yuchao Gu, Weijia Mao, and Mike Zheng Shou. Long-context autoregressive video modeling with next-frame prediction. *ArXiv*, abs/2503.19325, 2025. URL `https://api.semanticscholar.org/CorpusID:277313237`.

[26] Zekai Gu, Rui Yan, Jiahao Lu, Peng Li, Zhiyang Dou, Chenyang Si, Zhen Dong, Qifeng Liu, Cheng Lin, Ziwei Liu, Wenping Wang, and Yuan Liu. Diffusion as shader: 3d-aware video diffusion for versatile video generation control. *arXiv preprint arXiv:2501.03847*, 2025.

[27] Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Fei-Fei Li, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. In *ECCV*, 2024.

[28] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems 31*, pages 2451–2463. Curran Associates, Inc., 2018. URL `https://papers.nips.cc/paper/7512-recurrent-world-models-facilitate-policy-evolution`. `https://worldmodels.github.io`.

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

[30] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[31] Boyu Han, Qianqian Xu, Shilong Bao, Zhiyong Yang, Kangli Zi, and Qingming Huang. Lightfair: Towards an efficient alternative for fair t2i diffusion via debiasing pre-trained text encoders. In *Advances in Neural Information Processing Systems*, 2025.

[32] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for video diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=Z4evOUYrk7`.

[33] Hao He, Ceyuan Yang, Shanchuan Lin, Yinghao Xu, Meng Wei, Liangke Gui, Qi Zhao, Gordon Wetzstein, Lu Jiang, and Hongsheng Li. Cameractrl ii: Dynamic scene exploration via camera-controlled video diffusion models. *arXiv preprint arXiv:2503.10592*, 2025.

[34] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022.

[35] Roberto Henschel, Levon Khachatryan, Daniil Hayrapetyan, Hayk Poghosyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*, 2024.

[36] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[37] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, 2022.

[38] Yining Hong, Beide Liu, Maxine Wu, Yuanhao Zhai, Kai-Wei Chang, Linjie Li, Kevin Lin, Chung-Ching Lin, Jianfeng Wang, Zhengyuan Yang, Yingnian Wu, and Lijuan Wang. Slowfast-vgen: Slow-fast learning for action-driven long video generation, 2024. URL `https://arxiv.org/abs/2410.23277`.

[39] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024.

[40] Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Zhuang Nan, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. *ArXiv*, abs/2410.05954, 2024. URL `https://api.semanticscholar.org/CorpusID:273228937`.

[41] Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. In *ICLR*, 2025.

[42] Xuan Ju, Yiming Gao, Zhaoyang Zhang, Ziyang Yuan, Xintao Wang, Ailing Zeng, Yu Xiong, Qiang Xu, and Ying Shan. Miradata: A large-scale video dataset with long durations and structured captions, 2024. URL `https://arxiv.org/abs/2407.06358`.

[43] Dan Kondratyuk, Lijun Yu, Xiuye Gu, Jose Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, et al. Videopoet: A large language model for zero-shot video generation. In *ICML*, 2024.

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

[45] Alex X. Lee, Richard Zhang, Frederik Ebert, P. Abbeel, Chelsea Finn, and Sergey Levine. Stochastic adversarial video prediction. *ArXiv*, abs/1804.01523, 2018. URL `https://api.semanticscholar.org/CorpusID:4591836`.

[46] Muyang Li, Ji Lin, Chenlin Meng, Stefano Ermon, Song Han, and Jun-Yan Zhu. Efficient spatially sparse inference for conditional gans and diffusion models. *Advances in neural information processing systems*, 35:28858–28873, 2022.

[47] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[48] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, Tanghui Jia, Junwu Zhang, Zhenyu Tang, Yatian Pang, Bin She, Cen Yan, Zhiheng Hu, Xiao wen Dong, Lin Chen, Zhang Pan, Xing Zhou, Shaoling Dong, Yonghong Tian, and Li Yuan. Open-sora plan: Open-source large video generation model. *ArXiv*, abs/2412.00131, 2024. URL `https://api.semanticscholar.org/CorpusID:274436278`.

[49] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *ArXiv*, abs/2210.02747, 2022. URL `https://api.semanticscholar.org/CorpusID:252734897`.

[50] Songhua Liu, Weihao Yu, Zhenxiong Tan, and Xinchao Wang. Linfusion: 1 gpu, 1 minute, 16k image. *ArXiv*, abs/2409.02097, 2024. URL `https://api.semanticscholar.org/CorpusID:272366893`.

[51] Michaël Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *CoRR*, abs/1511.05440, 2015. URL `https://api.semanticscholar.org/CorpusID:205514`.

[52] Yuta Oshima, Shohei Taniguchi, Masahiro Suzuki, and Yutaka Matsuo. Ssm meets video diffusion models: Efficient long-term video generation with structured state spaces. *arXiv preprint arXiv:2403.07711*, 2024.

[53] J Parker-Holder, P Ball, J Bruce, V Dasagi, K Holsheimer, C Kaplanis, A Moufarek, G Scully, J Shar, J Shi, et al. Genie 2: A large-scale foundation world model. *URL: https://deepmind. google/discover/blog/genie-2-a-large-scale-foundation-world-model*, 2024.

[54] Jack Parker-Holder, Philip Ball, Jake Bruce, Vibhavari Dasagi, Kristian Holsheimer, Christos Kaplanis, Alexandre Moufarek, Guy Scully, Jeremy Shar, Jimmy Shi, Stephen Spencer, Jessica Yung, Michael Dennis, Sultan Kenjeyev, Shangbang Long, Vlad Mnih, Harris Chan, Maxime Gazeau, Bonnie Li, Fabio Pardo, Luyu Wang, Lei Zhang, Frederic Besse, Tim Harley, Anna Mitenkova, Jane Wang, Jeff Clune, Demis Hassabis, Raia Hadsell, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 2: A large-scale foundation world model. 2024. URL `https://deepmind.google/discover/blog/genie-2-a-large-scale-foundation-world-model/`.

[55] Ryan Po, Wang Yifan, Vladislav Golyanik, Kfir Aberman, Jonathan T Barron, Amit Bermano, Eric Chan, Tali Dekel, Aleksander Holynski, Angjoo Kanazawa, et al. State of the art on diffusion models for visual computing. *Computer Graphics Forum*, 43(2):e15063, 2024.

[56] Di Qiu, Zhengcong Fei, Rui Wang, Jialin Bai, Changqian Yu, Mingyuan Fan, Guibin Chen, and Xiang Wen. Skyreels-a1: Expressive portrait animation in video diffusion transformers. abs/2502.10841, 2025. URL `https://api.semanticscholar.org/CorpusID:276409428`.

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

[58] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. In *ICLR*, 2023.

[59] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. 2015.

[60] Kiwhan Song, Boyuan Chen, Max Simchowitz, Yilun Du, Russ Tedrake, and Vincent Sitzmann. History-guided video diffusion. *arXiv preprint arXiv:2502.06764*, 2025.

[61] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021.

[62] Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

[63] Anni Tang, Tianyu He, Junliang Guo, Xinle Cheng, Li Song, and Jiang Bian. Vidtok: A versatile and open-source video tokenizer. *ArXiv*, abs/2412.13061, 2024. URL `https://api.semanticscholar.org/CorpusID:274788955`.

[64] S. Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1526–1535, 2017. URL `https://api.semanticscholar.org/CorpusID:4475365`.

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

[66] R Villegas, H Moraldo, S Castro, M Babaeizadeh, H Zhang, J Kunze, PJ Kindermans, MT Saffar, and D Erhan. Phenaki: Variable length video generation from open domain textual descriptions. In *ICLR*, 2023.

[67] Carl Vondrick and Antonio Torralba. Generating the future with adversarial transformers. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2992–3000, 2017. URL `https://api.semanticscholar.org/CorpusID:8234308`.

[68] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[69] Hongjie Wang, Chih-Yao Ma, Yen-Cheng Liu, Ji Hou, Tao Xu, Jialiang Wang, Felix Juefei-Xu, Yaqiao Luo, Peizhao Zhang, Tingbo Hou, Peter Vajda, Niraj Kumar Jha, and Xiaoliang Dai. Lingen: Towards high-resolution minute-length text-to-video generation with linear computational complexity. *ArXiv*, abs/2412.09856, 2024. URL `https://api.semanticscholar.org/CorpusID:274763003`.

[70] Qianqian Wang*, Yifei Zhang*, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[71] Yuqing Wang, Tianwei Xiong, Daquan Zhou, Zhijie Lin, Yang Zhao, Bingyi Kang, Jiashi Feng, and Xihui Liu. Loong: Generating minute-level long videos with autoregressive language models. *arXiv preprint arXiv:2410.02757*, 2024.

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

[73] Wenming Weng, Ruoyu Feng, Yanhui Wang, Qi Dai, Chunyu Wang, Dacheng Yin, Zhiyuan Zhao, Kai Qiu, Jianmin Bao, Yuhui Yuan, et al. Art-v: Auto-regressive text-to-video generation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7395–7405, 2024.

[74] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *Conference on robot learning*, pages 2226–2240. PMLR, 2023.

[75] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. *ArXiv*, abs/2410.10629, 2024. URL `https://api.semanticscholar.org/CorpusID:273346094`.

[76] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

[77] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[78] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Xiaotao Gu, Yuxuan Zhang, Weihan Wang, Yean Cheng, Ting Liu, Bin Xu, Yuxiao Dong, and Jie Tang. Cogvideox: Text-to-video diffusion models with an expert transformer. *ArXiv*, abs/2408.06072, 2024. URL `https://api.semanticscholar.org/CorpusID:271855655`.

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

[80] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *CVPR*, 2025.

[81] Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos. *arXiv preprint arXiv:2501.08325*, 2025.

[82] Mark YU, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. *arXiv preprint arXiv:2503.05638*, 2025.

[83] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024.

[84] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *CVPR*, 2017.

[85] David Junhao Zhang, Roni Paiss, Shiran Zada, Nikhil Karnad, David E Jacobs, Yael Pritch, Inbar Mosseri, Mike Zheng Shou, Neal Wadhwa, and Nataniel Ruiz. Recapture: Generative video camera controls for user-provided videos using masked video fine-tuning. *arXiv preprint arXiv:2411.05003*, 2024.

[86] Lvmin Zhang and Maneesh Agrawala. Packing input frame context in next-frame prediction models for video generation. *arXiv preprint arXiv:2504.12626*, 2025.

[87] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[88] Zhicheng Zhang, Junyao Hu, Wentao Cheng, Danda Paudel, and Jufeng Yang. Extdm: Distribution extrapolation diffusion model for video prediction. In *CVPR*, 2024.

[89] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *ArXiv*, abs/2412.20404, 2024. URL `https://api.semanticscholar.org/CorpusID:275133398`.

[90] Zheng Zhu, Xiaofeng Wang, Wangbo Zhao, Chen Min, Nianchen Deng, Min Dou, Yuqi Wang, Botian Shi, Kai Wang, Chi Zhang, et al. Is sora a world simulator? a comprehensive survey on general world models and beyond. *arXiv preprint arXiv:2405.03520*, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We claim the introduction of a long-term memory mechanism, including a point-cloud-based spatial memory and an episodic memory comprised of a sparse set of keyframes along with memory storage and retrieval techniques. Our evaluations show that this method outperforms the baselines.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 5, paragraph on "Limitations and Future Work".

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The assumptions on our different memory mechanisms are: (1) short-term working memory: models static and dynamic parts of the scene, but is limited to short context window size due to quadratic growth of computational of attention; (2) long-term spatial memory: models the static parts of the scene, but not dynamic aspects and limited in the visual detail a point cloud can store via TSDF fusion; (3) long-term episodic memory: sparse image "snapshots" of past events.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provided implementation details as best as possible within the page limit of the paper, and additional details in the supplement. Code and trained model will be made public.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will make code and the pretained model available.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: These details are clarified in the paper and supplement and will also be easily reproducible from the provided code.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA]

Justification: Our metrics use standard benchmarks for the tasks we evaluate. These metrics do not include statistical significance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See text in paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We reviewed the Code of Ethics and confirm that this research conforms to it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Sec. 5.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: Our training and evaluation data is publicly available. Safeguards of the base model, i.e., CogVideoX, are inherited.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: See text.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: All code and data we will release will contain the required information.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: See text and supplement.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: We followed the standards of the country where user studies were performed and personal data analyzed.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs were exclusively used for writing, editing, or formatting purposes.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.