MemLearner: Learning to Query Context Memory for Video World Models

Jiwen Yu[1‡], Jianxiong Gao[2‡], Jianhong Bai[3‡], Yiran Qin[1], Kaiyi Huang[1‡], Quande Liu[4], Xintao Wang[4†], Pengfei Wan[4], Kun Gai[4], and Xihui Liu[1†]

[1] The University of Hong Kong
[2] Fudan University
[3] Zhejiang University
[4] Kuaishou Technology

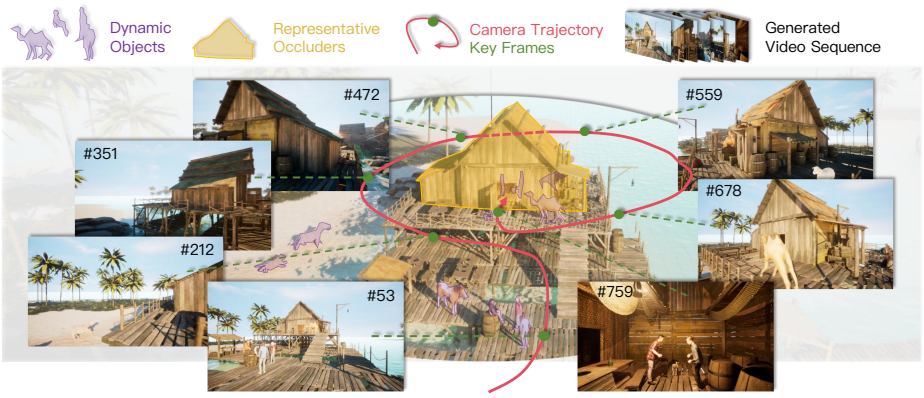

Fig. 1: Teaser Demonstration. We propose a novel memory mechanism for video world models through learning-based adaptive context querying. Compared to prior rule-based context retrieval methods [34, 55, 62], our approach handles scenes with occlusions and dynamic objects. This figure highlights dynamic objects, representative occluders, video generation trajectory, and key frames.

Abstract. Video World Models are interactive video generation models that predict future world states based on user actions and history video frames. A critical challenge in video world models is the lack of memory, causing inconsistent generated scenes over extended durations. Previous methods explored rule-based context frame retrieval as memory, but they fail to generalize in scenarios with scene occlusions and dynamic objects. We propose **MemLearner**, a learning-based adaptive context query method using query tokens to bridge context and predicted tokens. By leveraging the video generation model itself for context querying, MemLearner exploits pre-trained visual priors without training additional modules from scratch, and incorporates efficient strategies for training and inference. We collect a dataset of long videos with scene occlusions and dynamic objects, paired with camera pose annotations, and propose

‡Work done during an internship at Kling Team, Kuaishou Technology.
†Corresponding authors. Project page: `https://yujiwen.github.io/memlearner/`.

a multi-dataset training strategy leveraging both annotated rendered and unannotated real-world videos. Extensive experiments demonstrate that MemLearner significantly outperforms prior video world models in terms of scene consistency and memory, particularly under challenging occlusion and dynamic scenarios.

Keywords: Video World Model · Interactive Video Generation · Video Diffusion Model

1 Introduction

World Models [21] understand and simulate world dynamics by taking historical world states and interactive actions as input to predict new states. World states can be represented as language [8], latent representations [4, 21, 68], 3D/4D [32, 60], or videos [6, 41]. Among these, Video World Models are particularly promising [57,63,64] as videos photorealistically capture real-world dynamics, and abundant internet video data offers significant scaling potential. Despite significant progress made in generating short video clips, Video World Models still face significant challenges of scene consistency over extended durations, caused by insufficient memory mechanisms.

The memory issue in Video World Models arises when later generated scenes become inconsistent with earlier ones due to limited context windows. Existing methods address this challenge by different memory representations: 3D reconstruction [38, 40, 44, 53, 61, 66], compressed feature representations [27, 43, 67, 69], and retrieving key context frames [12, 34, 55, 62]. Context retrieval is particularly promising as it eliminates additional costs and potential errors caused by 3D reconstruction or history compression.

However, existing context retrieval methods are rule-based, relying on FOV overlap [55, 62] or point cloud estimation and surfel matching [34], facing several fundamental limitations in complex scenes with occlusions and dynamic objects. For example, FOV-based context retrieval [62] cannot account for occluding walls between camera views, point-cloud-based retrieval methods cannot accurately reconstruct moving objects. Moreover, these hard-coded retrieval rules fail to take generalized and dynamic environments with state changes into consideration. **These limitations motivate a paradigm shift: rather than using hand-crafted rules to retrieve context, we propose a new memory mechanism that enables the network to learn to adaptively query information from historical frames through end-to-end training.**

We formulate history-conditioned video generation as predicting future frames based on historical context frames. To design a learning-based context query method, we introduce query tokens (**Q** tokens) as an information bridge between context tokens (**C** tokens) and predicted tokens (**P** tokens) (Fig. 2 (a)): **Q** tokens attend to **C** tokens to adaptively extract context information, while **P** tokens attend to **Q** tokens as the generation condition. A key design choice is to leverage the video generation model itself for context querying, rather than introducing a separate module. Specifically, we feed all **C**, **Q**, and **P** tokens together into the

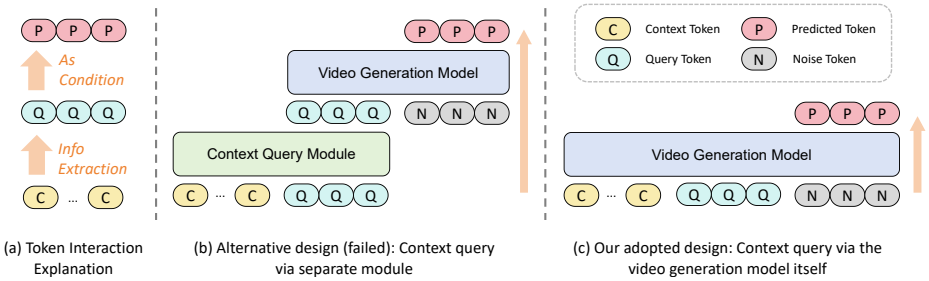

Fig. 2: Architecture clarification. (a) Interaction mechanism among **C**, **Q**, and **P** tokens; (b)&(c) Two designs for context querying, where the alternative design in (b) fails in experiments (Sec. 5.4), and the adopted design in (c) leverages the prior knowledge of the video generation model itself and performs effectively.

video generation model (Fig. 2 (c), detailed in Sec. 3.2), exploiting the model's pre-trained visual priors without additional scratch-trained modules. To further reduce the computational cost of context querying over long video sequences, we propose two efficient strategies for training and inference (Sec. 3.3).

Training learning-based context querying requires a long video dataset with occlusion, dynamic objects, camera pose annotations, and sufficient diversity. However, no existing datasets [37, 50, 62, 70] fully meet these criteria. Real-world videos (e.g., YouTube) provide diverse and dynamic content but lack precise camera pose annotations, while rendered videos (e.g., Unreal Engine) offer accurate poses but suffer from limited visual realism and diversity. To address this, we make two contributions. First, we collect a rendered dataset with customized occlusions and dynamic objects (Sec. 4.1). Second, we propose a multi-dataset training strategy that assigns a dedicated camera encoder to each dataset type, where unannotated data is fed with zero camera parameters. By isolating distinct annotation qualities into separate encoders, the model simultaneously leverages the strengths of diverse data sources without mutual interference (Sec. 4.2).

Our contributions can be summarized as follows:

- We propose MemLearner, a learning-based adaptive context query method that utilizes query tokens to extract valid memory from context tokens, enabling memory-augmented Video World Models with improved consistency and generalization.
- We collect a customized video world model dataset based on Unreal Engine, incorporating occlusion and dynamic objects, and propose a multi-dataset training strategy to leverage the advantages of both rendered and real videos.
- Extensive experiments demonstrate that MemLearner significantly outperforms previous Video World Models in terms of scene consistency and memory, particularly under occlusion and dynamic object scenarios.

2 Related Work

2.1 Video World Models

World Model. The World Model [21] refers to a system that takes historical world state and current action as input to predict future world state. Its primary purpose is to understand and simulate the evolving process of the world. The world state can be represented in various forms, such as language [8], semantic representations [4, 68], 3D/4D [32, 60], and videos [6, 41]. Recently, due to video photorealism, large data volumes, and breakthroughs in video generation models, Video World Models are considered a promising path to World Models [57,64,64].

Video Generation Models. Mainstream video generation models now use the Diffusion Transformer (DiT) [41, 42] architecture, enabling highly realistic video creation [7, 16, 29, 31, 41, 45, 49, 59]. Other architectures include next-token prediction [14, 30, 51, 56] and hybrid approaches [11, 17, 35], but fall short of DiT in generation quality. Interactive control in video generation includes camera pose [5,22,52], trajectory [18,52], and action control [48,65,71], with applications in film production, game video control, and robotic simulation. Streaming video generation conditions on previously generated frames to create new content, with both frame-by-frame [11,46,67] and chunk-by-chunk methods [1,62,65] primarily using DiT. Long video generation suffers from error accumulation, which several methods address [13, 25, 36, 39, 58]. Since our work focuses on learnable memory rather than ultra-long generation, we do not discuss error accumulation.

2.2 Memory Mechanism for Video World Models

Memory capability in video world models refers to generating consistent long videos, especially when revisiting scenes or objects. It is a fundamental prerequisite for planning, interaction and state modeling, as these core capabilities rely on retaining past visual memories. However, many methods struggle with this [15, 26, 46, 48], due to limited context windows that cannot retain sufficient historical information. To address this, additional conditional information is necessary. Existing approaches fall into three paradigms: (1) 3D as memory [38, 40, 44, 53, 61, 66]: reconstructing 3D representations from historical frames and rendering new initial frames as conditions; (2) Feature as memory [27, 69]: extracting semantic features from historical frames or maintaining learnable features injected into the generation model; (3) Context as memory [12, 24, 34, 43, 47, 55, 62, 67]: directly using historical frames as conditions. Among these, context memory is the most straightforward. However, incorporating all historical contexts introduces prohibitive computational overhead.

Some works propose context retrieval to select historical contexts relevant to current generation. Conditioning on only retrieved frames significantly reduces computational overhead. Existing methods typically rely on **rule-based retrieval** [34,55,62], which lack generalization across videos and scenes. We propose a **learning-based context query method** enabling the model to learn to query useful historical information, thereby achieving learnable memory.

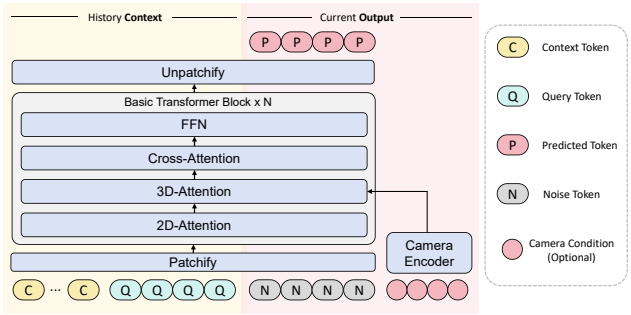

Fig. 3: Model Architecture. Our video generation model adopts a Diffusion Transformer. We concatenate **C**, **Q**, **P** tokens for context-conditioned long video generation. An optional camera encoder supports interactive control.

2.3 Learnable Query Tokens in Various Domains

Learnable query tokens have been widely adopted for compressing visual context across various domains: In multimodal language models, Perceiver Resampler [2] and Q-Former [33] use learnable queries to aggregate visual features for language model consumption; In video understanding, adaptive frame selection methods [3, 9, 54] employ learnable strategies to identify informative frames for recognition and reasoning tasks. Our work addresses a fundamentally different problem: memory issue in video world models, where query tokens serve as an information bridge between context tokens and predicted tokens, adaptively extracting fine-grained, token-level information across multiple historical frames to support consistent video generation under occlusion and dynamic scenarios.

3 MemLearner

3.1 Preliminaries

Our approach is built upon a latent video diffusion model with a causal 3D VAE [28] and Diffusion Transformer (DiT) [42]. As shown in Fig. 3, each DiT block sequentially contains spatial (2D) attention, spatial-temporal (3D) attention, cross-attention, and feed-forward networks (FFN). The 3D VAE encoder compresses a video frame sequence $\mathbf{x}$ temporally and spatially into latent representations $\mathbf{z} = E(\mathbf{x})$, and the DiT is trained to predict noise scaled and added to $\mathbf{z}$ via a standard diffusion objective. We integrate camera control [5, 52] by injecting camera poses $\mathbf{cam} = [R, t] \in \mathbb{R}^{f\times(3\times4)}$ into the model via a one-layer MLP encoder $\mathcal{E}_c(\cdot)$, added to features between the 2D and 3D attention layers. Note that camera control only guides the trajectory of generated videos; our learning-based context query method does not rely on camera poses, as validated in Sec. 5.5. For long video generation, we follow a chunk-by-chunk autoregressive paradigm: historical context frame latents $\mathbf{h}$ are concatenated with the to-be-generated latents $\mathbf{z}$ along the frame dimension and fed jointly into the model,

with the diffusion loss applied only to $\mathbf{z}$. This preserves the model's generative priors without architectural modifications.

3.2 Learning to Query Context

Introducing Adaptive Query Tokens. Our key insight is that different predicted frames require different guidance information from context frames, and even within the generation of a single frame, different denoising stages of the diffusion process need to emphasize different aspects of historical information. We verify this via attention similarity analysis in Supplementary Material Sec. C.2: query tokens exhibit markedly different attention distributions across predicted frames and diffusion timesteps, with early timesteps attending more broadly to context while later timesteps focus on fine-grained local correspondences. Naive history compression or rule-based keyframe retrieval fail to tackle complex scenarios such as occlusion and dynamic objects. To enable adaptive memory, we introduce learnable query tokens $\mathbf{Q}$ that bridge context and generation. These query tokens dynamically extract relevant information from context tokens $\mathbf{C}$ and guide the generation of predicted tokens $\mathbf{P}$. Importantly, this querying mechanism can be learned end-to-end via indirect supervision from the diffusion loss on $\mathbf{P}$ alone, without explicit supervision for a dedicated querying module.

Architecture Design Insights. The intuitive design introduces an additional context query module, as in Fig. 2 (b). However, our experiments show that when jointly training the architecture, the from-scratch context query module fails to learn useful information. Attention similarity analysis (Supplementary Material Sec. C.2) confirms that this module produces near-zero similarity between query and context tokens, indicating a failure to establish meaningful context modeling; this in turn hinders gradient propagation to the video generation model, which consequently ignores the module's output and degrades to a text-to-video model (Sec. 5.4). As in Fig. 2 (c), our design avoids a separate from-scratch module and instead leverages the pre-trained video generation model itself for context querying. This offers clear advantages: (1) it exploits the model's prior knowledge, reducing data and compute requirements; (2) end-to-end training naturally learns context query capability without requiring additional input-output design or supervision losses for a separate module.

Architecture Details. Our work is based on a latent video diffusion model. We denote the video latent tokens as $Z = \{\mathbf{C}, \mathbf{Q}, \mathbf{P}\}$, where $\mathbf{C}$, $\mathbf{Q}$ and $\mathbf{P}$ denote context, query, and predicted tokens, respectively. The diffusion model input is the noisy token at timestep t, written as $Z_t = \{\mathbf{C}, \mathbf{Q}, \mathbf{P}_t\}$, where the context tokens and query tokens remain unperturbed, while the predicted tokens are noised only at the input (and not at any subsequent layer) via randomly sampled Gaussian noise to obtain $\mathbf{P}_t$, with the scaled and added noise being $\epsilon^P \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The training loss for the entire architecture is:

$$\mathcal{L}(\theta) = \mathbb{E}[||\epsilon_\theta(Z_t, \mathbf{cam}, \mathbf{p}, t)|| - \epsilon^P], \tag{1}$$

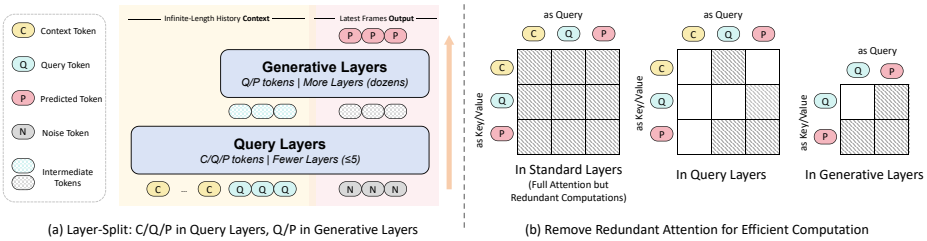

Fig. 4: Efficiency Strategies. (a) Strategy#1: Context querying in shallow Query Layers with **C**, **Q**, **P** tokens; deep Generative Layers use only **Q**, **P** tokens. (b) Strategy#2: Remove unnecessary attention computation for improved efficiency.

where θ denotes all learnable parameters. Note that supervision is applied only to the noise predicted on predicted tokens. We concatenate **C**, **Q** and **P** along the frame dimension and input them into the video DiT, where different tokens interact with each other in the 3D attention. For a 3D attention layer with input F_{in} and output F_{out}, we have $F_{in} = \{\mathbf{C}, \mathbf{Q}, \mathbf{P}\}$ and $F_{out} = \{\mathbf{C}_{out}, \mathbf{Q}_{out}, \mathbf{P}_{out}\}$. For brevity, we use **C**, **Q**, and **P** to denote the features corresponding to different tokens. We define the linear computations in the attention block as $q(\cdot)$, $k(\cdot)$, $v(\cdot)$, and $o(\cdot)$, which correspond to the operations for query, key, value, and output. A standard 3D attention computation is:

$$\begin{aligned} F_{out} &= F_{in} + o(\mathbf{sm}(q(F_{in})k(F_{in})^{\top})v(F_{in})), \\ &= F_{in} + g(F_{in}, F_{in}, F_{in}), \end{aligned} \tag{2}$$

where $\mathbf{sm}(\cdot)$ denotes the softmax operation, and $g(\cdot,\cdot,\cdot)$ is a shorthand notation indicating the inputs for query, key, and value operations, respectively. Considering that the context frame length may be very large, this computation in Eq. 2 is expensive and inefficient.

3.3 Efficient Strategies

We present two simple yet effective strategies to improve the efficiency of Eq. 2 during both training and inference.

Strategy #1: Query Only in Early Layers. Information querying is analogous to encoding, which requires fewer parameters and computations than the generation model (e.g., video VAEs use far fewer parameters than video generation models). Therefore, context querying can be performed only in the early shallow layers, which our experimental results validate. Specifically, assuming the diffusion transformer has $n + m$ layers in total, we divide them into two types as shown in Fig. 4 (a): n shallow Query Layers near the input and m deep Generative Layers near the output, where $n \ll m$. Interactions among **C**, **Q**, and **P** occur only in Query Layers, while Generative Layers process only **Q** and **P**.

Strategy #2: Exclude Unnecessary Computations. We exclude unnecessary attention computations by retaining only three essential patterns as shown in Fig. 4 (b): (1) $\mathbf{Q}$ as queries attending to $\mathbf{P}$ as keys/values, enabling $\mathbf{Q}$ to understand what information to extract from context based on the prediction target; (2) $\mathbf{Q}$ as queries attending to $\mathbf{C}$ as keys/values, enabling $\mathbf{Q}$ to extract information from context; (3) $\mathbf{P}$ as queries attending to $\mathbf{P}$ and $\mathbf{Q}$ as keys/values, enabling $\mathbf{P}$ to extract information from both. All other attention computations are excluded, particularly those where $\mathbf{C}$ serve as queries, significantly reducing computational overhead. We provide experimental analysis of performance under different attention patterns in Sec. 5.5.

Combining the two strategies, we present the revised Eq. 2. For Query Layers:

$$\begin{aligned} \mathbf{C}_{out} &= \mathbf{C}, \\ \mathbf{Q}_{out} &= \mathbf{Q} + g(\mathbf{Q}, \{\mathbf{C}, \mathbf{P}\}, \{\mathbf{C}, \mathbf{P}\}), \\ \mathbf{P}_{out} &= \mathbf{P} + g(\mathbf{P}, \{\mathbf{P}, \mathbf{Q}\}, \{\mathbf{P}, \mathbf{Q}\}). \end{aligned} \tag{3}$$

For Generative Layers, features of $\mathbf{C}$ are removed:

$$\begin{aligned} \mathbf{Q}_{out} &= \mathbf{Q} + g(\mathbf{Q}, \mathbf{P}, \mathbf{P}), \\ \mathbf{P}_{out} &= \mathbf{P} + g(\mathbf{P}, \{\mathbf{P}, \mathbf{Q}\}, \{\mathbf{P}, \mathbf{Q}\}). \end{aligned} \tag{4}$$

4 Dataset

4.1 Dataset Collection

To validate our method, we require a dataset with specific characteristics: long videos, accurate per-frame camera annotations, occlusion relationships, dynamic objects, and sufficient diversity. However, existing long video datasets do not meet these requirements, as shown in Tab. 1. To address this, we collect a rendered dataset based on Unreal Engine with the following two key designs: (1) **Customized scenes and dynamic objects.** We curate a diverse collection of scenes with occlusion relationships, including factories, streets, and natural landscapes, as well as dynamic objects such as humans and animals with various actions. By combining different scenes and dynamic objects, we construct diverse dynamic scenarios. (2) **Automated trajectory generation.** To enable automated trajectory generation with collision avoidance in dynamic scenes, we implement a blueprint script in Unreal Engine. This script can randomly generate camera trajectories of arbitrary length with automatic obstacle avoidance, and the traversal range is customizable. This significantly improves our data collection efficiency. Using this pipeline, we collect 100 long videos across 13 scenes, with each video averaging over 18000 frames, totaling 16.7 hours of video content. Additional details about data collection can be found in the Supplementary Material Sec. A.

Table 1: Comparison of long video datasets for video world models. No existing dataset fully satisfies all four requirements: precise per-frame camera pose annotations, occlusion relationships, dynamic objects, and revisit scenarios. We collect a dataset meeting these requirements. [1]HQ subset only. [2]Available subset only. [3]Insufficient annotation accuracy. [4]Not all frames are annotated. [5]Primarily street walking videos with limited occlusions. [6]Filtered videos with reduced dynamics. [7]Few revisits without specialized design, especially for real-world data.

✔: Fully satisfied ●: Partially satisfied ✗: Not satisfied

Dataset	Source	Style	Duration	Cam. Pose	Occlusion	Dynamic	Revisit
CaM [62]	Simulator	Rendered	7.0 h	✔	✗	✗	✔
SpatialVid [50]	YouTube	Real-World	1146.06 h[1]	✔	●[5]	●[6]	✗[7]
Sekai-real [37]	YouTube	Real-World	304.0 h[1]	●[3]	●[5]	✔	✗[7]
OmniWorld [70]	Simulator	Rendered	11.14 h[2]	✗[4]	✔	✔	●[7]
Ours	Simulator	Rendered	16.7 h	✔	✔	✔	✔

4.2 Multi-Dataset Training Strategy

Video datasets can be categorized into three types based on camera pose annotation accuracy and visual realism: (1) Rendered data [62] with precise pose annotations but non-photorealistic style and limited diversity; (2) Real-world data with estimated poses [50], offering photorealistic style but less precise annotations and limited dynamics due to the filtering required for accurate pose estimation; and (3) Real-world data without accurate pose annotations [37], providing photorealistic style, rich content diversity, and better dynamics, but lacking accurate camera annotations.

To leverage the complementary strengths of these data sources, we propose a multi-dataset training strategy that assigns a dedicated camera encoder to each dataset type. Specifically, rendered data with precise poses and estimated-pose data each use a separate camera encoder to process their respective camera annotations, while data without reliable poses is fed with zero camera parameters (i.e., $\mathbf{R} = \mathbf{0}, \mathbf{t} = \mathbf{0}$) through a third dedicated camera encoder. By isolating different annotation qualities into separate encoders, datasets with varying pose accuracy do not interfere with each other during training, allowing the model to benefit from precise pose supervision of rendered data, the realism of estimated-pose data, and the diversity of unannotated real-world data simultaneously. During inference, we use only the camera encoder trained on precisely annotated data to ensure reliable camera control.

5 Experiments

5.1 Implementation Details.

In our implementation, $\mathbf{P}$ tokens correspond to 77 video frames per generation. The number of $\mathbf{Q}$ tokens equals that of $\mathbf{P}$ tokens. Rather than using learnable parameters, $\mathbf{Q}$ tokens are initialized as randomly sampled noise, which better aligns with the input distribution of diffusion models. During training and inference, $\mathbf{C}$ tokens can be up to 9 times the length of $\mathbf{P}$ tokens, with the last frame of

the **C** token sequence always matching the first frame of the **P** token sequence to ensure continuity. During training, the length of **C** tokens is uniformly sampled from 0 to 9 times the **P** token length, where 0 corresponds to training only the image-to-video capability. We perform full fine-tuning of the video generation model on the collected dataset.

Our primary experiments are built upon an internal 1B-parameter text-to-video diffusion transformer with 28 layers, of which 5 serve as Query Layers. We select this model for its stronger visual quality than open-source models of comparable scale and far lower computational costs than larger alternatives (*e.g.*, 5B, 14B), enabling more comprehensive ablation studies with higher-quality baselines. To verify that MemLearner generalizes across architectures, we additionally apply our method to Wan2.1 (T2V-1.3B) [49], an open-source video DiT. As detailed in Supplementary Material Sec. C.1, MemLearner achieves consistent improvements over the CaM [62] baseline on the open-source model, confirming that our approach is architecture-agnostic and benefits from pre-trained priors regardless of the specific backbone.

Videos are generated at 640×352 resolution, producing 77 frames with a causal 3D VAE that applies a temporal compression ratio of 4, yielding 20-frame video latents. We configure **P** tokens to represent 20 video latent frames, with **Q** tokens matching the **P** token count, while **C** tokens can represent up to 180 video latent frames. Training is conducted for over 20,000 iterations with a batch size of 8 and a learning rate of 5×10^{-5}. We explore two dataset configurations: (1) **Rendered-only**: 50% from our collected dataset and 50% from CaM [62]; (2) **Mixed**: 75% rendered data (combining configuration 1) and 25% real-world data (12.5% Sekai-real [37] and 12.5% SpatialVid [50]). For sampling, we apply Classifier-Free Guidance [23] with 50 steps for text conditioning.

5.2 Evaluation Methods

For evaluation, we reserve a 5% held-out test set via random dataset partitioning with no video overlap between training and test splits. Our evaluation protocol includes: (1) **FID/FVD** to assess video visual quality; (2) **PSNR/LPIPS** to measure memory capability by quantifying pixel-wise differences between frames. Following prior work [62], we adopt two evaluation strategies for memory assessment: (1) **Ground truth comparison (GT Comp.)**: measuring whether predicted frames align with ground truth given context from ground truth frames. With sufficient context, generated results should match the ground truth; (2) **Revisit comparison (Revisit Comp.)**: comparing newly generated revisit frames against previously generated ones in long video sequences. For implementation, we test on simple camera trajectories where the camera rotates n degrees with $n \sim \mathcal{U}(90°, 180°)$ and returns, enabling straightforward identification of corresponding frame pairs. We then compute PSNR/LPIPS between all corresponding frame pairs from the outbound and return paths and average the results, thereby evaluating consistency throughout the entire generation process rather than just between the first and last frames.

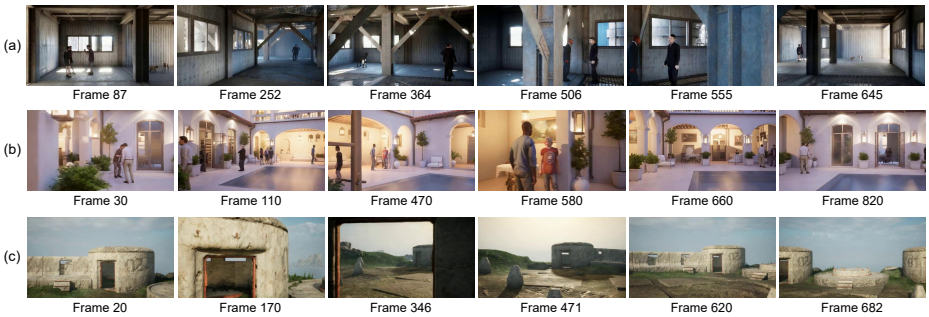

Fig. 5: Qualitative Results. MemLearner effectively handles both indoor (a, b) and outdoor (c, Fig. 1) scenes with occlusions and dynamic objects.

Fig. 6: Real-World Qualitative Results. By incorporating real-world videos into training dataset, MemLearner generalizes to real-world scenes.

To ensure evaluation diversity, we conduct experiments across multiple datasets with distinct characteristics: our collected dataset with occlusions and dynamic objects (Tab. 2, Fig. 7), the CaM dataset [62] without occlusions or dynamics (Tab. 3), and the real-world SpatialVID dataset [50] (Supplementary Material Sec. C.3), where MemLearner consistently outperforms baselines across all three evaluation settings. We further provide a user study in Supplementary Material Sec. C.4, in which MemLearner is preferred over other SOTAs in terms of visual quality and scene consistency.

5.3 Qualitative Results

We present qualitative examples to illustrate MemLearner generation quality. As shown in Fig. 5, our method handles diverse indoor/outdoor scenes with rich occlusions and dynamic objects. We provide video demos in the supplementary HTML for direct viewing. Since our mixed training uses real-world data, MemLearner generalizes to real-world domains without adaptation. Fig. 6 shows the model preserves scene layout and object appearance across revisits, confirming the learned query mechanism generalizes to real environments.

5.4 Comparison Results

In this section, we compare baselines and SOTAs: (1) DFoT [46], a long video generation model without memory design; (2) FramePack [67], which hierarchically compresses history frames as conditions to retain limited memory; (3)

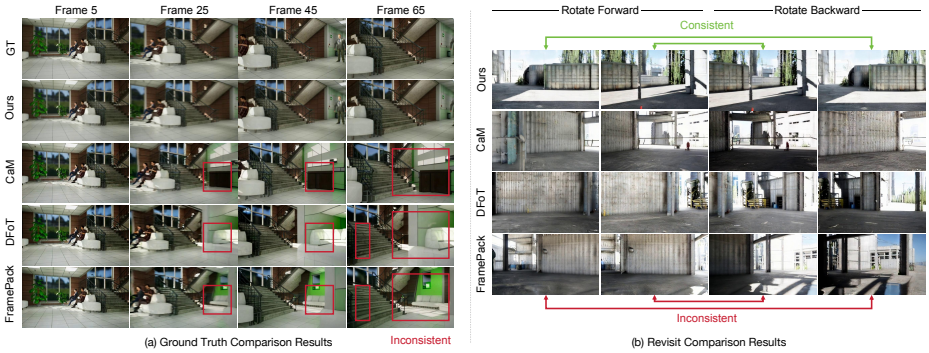

Fig. 7: Qualitative Comparison. MemLearner achieves optimal memory and visual quality, demonstrating learning-based adaptive context query effectiveness. Other methods show inconsistent memory performance.

Table 2: Quantitative Comparison. MemLearner outperforms all metrics. Notably, the Separate Module (Fig. 2 (b)) lacks context learning capability, indicating joint training of a scratch module with pre-trained video DiT is ineffective.

Methods	GT Comp. PSNR↑	LPIPS↓	FID↓	FVD↓	Revisit Comp. PSNR↑	LPIPS↓	FID↓	FVD↓	fps↑
DFoT [46]	16.98	0.4796	147.09	998.43	16.14	0.5481	151.38	1021.46	**1.59**
FP [67]	16.42	0.5104	143.97	967.97	15.86	0.5837	154.11	1037.58	1.40
VMem [34]	19.59	0.3872	129.94	850.17	17.30	0.4187	141.82	968.45	0.73
CaM [62]	19.85	0.3475	125.35	848.61	17.61	0.3934	137.87	948.63	0.97
Fig. 2 (b)	9.16	0.6567	145.63	930.54	-	-	-	-	0.48
Ours	**21.23**	**0.2904**	**112.75**	**835.98**	**18.57**	**0.3230**	**101.57**	**847.52**	0.54

VMem [34], which achieves memory capability via a point-cloud-based retrieval rule for context retrieval. (4) Context-as-Memory [62], which selects relevant conditioning frames via FOV-overlap computation to provide effective memory; (5) Separate Module, i.e., the design in Fig. 2 (b), where an additional five-layer Transformer (with the same layer structure as the video DiT) serves as a context query module. Its output channels are aligned with the video DiT's internal features, eliminating the need for a patchify layer; (6) our proposed MemLearner.

For fair comparison, all methods are implemented using our codebase and dataset. We evaluate both memory-related metrics (PSNR/LPIPS) and visual quality metrics (FID/FVD). Results are summarized in Tab. 2 and Fig. 7. MemLearner achieves the best performance across all metrics and benchmarks. Separate Module achieves poor performance on GT Comp., as it fails to learn conditioning on context information and behaves like a text-to-video model. This demonstrates that jointly training a from-scratch context query module with a pre-trained video generation model cannot effectively learn context query capability; instead, the video generation model ignores the context query module. In contrast, MemLearner, which leverages the video generation model itself for context querying, is significantly more effective.

Table 3: Quantitative comparison on CaM dataset [62] (no occlusions/dynamics). Small performance gaps, widening sharply on our occlusion/dynamic dataset (Tab. 2).

Methods	GT Comp. PSNR↑	GT Comp. LPIPS↓	Revisit Comp. PSNR↑	Revisit Comp. LPIPS↓
VMem [34]	19.84	0.3564	17.98	0.3633
CaM [62]	20.22	0.3003	18.11	0.3414
Ours	**20.35**	**0.2975**	**18.29**	**0.3374**

Table 4: Ablation of Attention Computation: Settings (a)-(c) in Fig. 8. Critical computation is **Q** queries attending to **P** keys/values.

Setting	GT Comp. PSNR↑	GT Comp. LPIPS↓	Revisit Comp. PSNR↑	Revisit Comp. LPIPS↓	fps↑
(a)	**21.23**	**0.2904**	**18.57**	**0.3230**	0.54
(b)	21.19	0.2949	18.53	0.3342	0.51
(c)	17.27	0.4657	16.34	0.5107	**0.56**

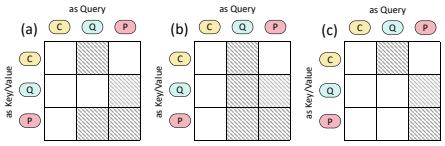

Fig. 8: Attention computation settings for ablation study in Tab. 4. Performance impact of attention scope when **Q** tokens act as queries.

Table 5: Ablation of camera embedding. Even without camera poses for **C** tokens, the model learns to query useful information with no significant performance drop.

Add Camera Emb.	GT Comp. PSNR↑	GT Comp. LPIPS↓	Revisit Comp. PSNR↑	Revisit Comp. LPIPS↓
All tokens	**21.23**	**0.2904**	**18.57**	**0.3230**
Only P tokens	21.17	0.2946	18.38	0.3384

In addition to our collected dataset, we also conduct quantitative evaluations on CaM dataset [62], which lacks occlusion and dynamic object scenes as shown in Tab. 1. Results in Tab. 2 and Tab. 3 show CaM and our method perform comparably on CaM dataset (no occlusions/dynamics), while CaM degrades sharply on our dataset and our method remains robust. This directly validates our approach's effectiveness for handling occlusions and dynamic objects.

5.5 Ablation Study

Ablation of Attention Computation. As shown in Fig. 8, we compare three attention variants: (a) standard setting; (b) augmenting (a) with **Q** tokens as queries attending to **Q** tokens as keys/values; (c) removing from (a) the pattern where **Q** tokens as queries attend to **P** tokens as keys/values. Results in Tab. 4 show that (b) matches (a) in performance, the added computation is non-critical and removable. In contrast, (c) exhibits significant performance degradation, demonstrating that it is essential for **Q** tokens to extract guidance from **P** tokens to determine what information to query from **C** tokens. We further analyze the effect of including **C** tokens as queries in Supplementary Material Sec. C.5, which introduces substantial computational overhead with negligible benefit.

Ablation of Query Layer Number. We ablate Query Layer count impact in Tab. 6. Performance saturates at 5 layers, with computational overhead rising thereafter. 5 layers balance computational cost and memory performance.

Ablation of Camera Embedding. When injecting camera pose embeddings, providing them to **P** tokens is essential for interactive video generation. However, it is unclear whether **C** tokens and **Q** tokens also require camera pose information.

Table 6: Ablation of Query Layer Count: 5 layers balance computational cost and memory performance.

#Layer	GT Comp. PSNR↑	GT Comp. LPIPS↓	Revisit Comp. PSNR↑	Revisit Comp. LPIPS↓	fps↑
1	18.16	0.4056	17.25	0.4144	**0.61**
3	19.43	0.3625	17.83	0.3808	0.58
5	21.23	0.2904	18.57	0.3230	0.54
10	21.36	0.2913	18.50	**0.3214**	0.46
20	**21.37**	**0.2891**	**18.66**	0.3217	0.36

Table 7: Mixed Dataset Training Ablation: Real-world data alone cannot learn memory capability (insufficient revisit).

Datasets	Revisit Comp. PSNR↑	Revisit Comp. LPIPS↓	Real Quality FID↓	Real Quality FVD↓
CaM [62]	18.35	0.3263	167.47	1368.01
Real [37,50]	15.32	0.6019	**104.16**	**828.95**
CaM+Real	18.27	0.3312	111.33	849.98
CaM+Ours	**18.57**	**0.3230**	161.83	1379.42
CaM+Ours+Real	18.49	0.3251	115.47	837.91

We investigate this in Tab. 5, comparing two settings: injecting camera poses only into **P** tokens versus all tokens. For **C** tokens, we inject the corresponding camera pose, while **Q** tokens receive the same as **P** tokens. Surprisingly, the results show that even without providing camera poses to **C** and **Q** tokens, the memory performance does not degrade significantly. This suggests that video generation models have the potential to learn geometric correspondences implicitly without explicit 3D information such as camera poses.

Ablation of Datasets. In Tab. 7, we ablate the impact of training with different dataset mixtures. 'CaM' refers to the dataset proposed in Context-as-Memory [62], 'Real' denotes the mixture of SpatialVID [50] and Sekai-real [37] real-world datasets, and 'Ours' is our proposed dataset. When combining rendered datasets 'CaM' and 'Ours', we use a 1:1 ratio, while rendered and real datasets are mixed at a 3:1 ratio. The 'Real Quality' metric measures the FID/FVD distance between generated videos and real-world videos. We observe that training solely on real-world data fails to learn memory capabilities, as evidenced by poor performance across all metrics, indicating insufficient revisit patterns in these real-world datasets. Models trained purely on rendered data produce videos whose distribution deviates from real videos. Incorporating real data into training effectively improves the realism of generated videos, as further illustrated by qualitative comparisons in Supplementary Material Sec. C.6.

6 Conclusion

In this work, we propose a paradigm shift for context memory in video world models, transitioning from rule-based context retrieval to learning-based context query. To validate our method, we collect rendered data with accurate camera annotations, customized occlusions and dynamic objects, and sufficient revisit patterns. We further propose a multi-dataset training strategy to effectively leverage both rendered and real-world videos.

Limitations and Future Work. While MemLearner advances context memory in video world models, the memory problem remains far from solved: (1) Expanding model capacity and training data scale is imperative. At our current 1B model scale, scenes with many simultaneously interacting characters in large environments remain challenging, as the model lacks sufficient capacity to track

numerous dynamic entities. We observe that generation quality degrades notably when more than five characters interact within the same scene, producing inconsistent appearances or missing objects upon revisit. Scaling to larger models and collecting long video data with richer dynamic interactions are necessary to address this. (2) Current research, including ours, focuses on full context storage and efficient information retrieval/querying [10,19,20,34,43,55,62], yet memory should not scale linearly with generation time. Notably, context compression is orthogonal to our contribution: one can first compress context representations and then apply our learned query mechanism on the compressed tokens. Exploring compression, summarization, updating, editing and selective forgetting are key future directions for practical, intelligent memory systems.

Acknowledgements

This work is supported by the Research Grant Council of Hong Kong through Early Career Scheme under grant 27214925. We thank Yao Teng, Xiaoshi Wu, and Haoran He for constructive discussions and suggestions.

References

1. ai, S., Teng, H., Jia, H., Sun, L., Li, L., Li, M., Tang, M., Han, S., Zhang, T., Zhang, W.Q., Luo, W., Kang, X., Sun, Y., Cao, Y., Huang, Y., Lin, Y., Fang, Y., Tao, Z., Zhang, Z., Wang, Z., Liu, Z., Shi, D., Su, G., Sun, H., Pan, H., Wang, J., Sheng, J., Cui, M., Hu, M., Yan, M., Yin, S., Zhang, S., Liu, T., Yin, X., Yang, X., Song, X., Hu, X., Zhang, Y., Li, Y.: Magi-1: Autoregressive video generation at scale (2025), https://arxiv.org/abs/2505.13211
2. Alayrac, J.B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al.: Flamingo: a visual language model for few-shot learning. Advances in neural information processing systems **35**, 23716–23736 (2022)
3. Arnab, A., Iscen, A., Caron, M., Fathi, A., Schmid, C.: Temporal chain of thought: Long-video understanding by thinking in frames. arXiv preprint arXiv:2507.02001 (2025)
4. Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., Muckley, M., Rizvi, A., Roberts, C., Sinha, K., Zholus, A., et al.: V-jepa 2: Self-supervised video models enable understanding, prediction and planning. arXiv preprint arXiv:2506.09985 (2025)
5. Bai, J., Xia, M., Fu, X., Wang, X., Mu, L., Cao, J., Liu, Z., Hu, H., Bai, X., Wan, P., et al.: Recammaster: Camera-controlled generative rendering from a single video. arXiv preprint arXiv:2503.11647 (2025)
6. Ball, P.J., Bauer, J., Belletti, F., Brownfield, B., Ephrat, A., Fruchter, S., Gupta, A., Holsheimer, K., Holynski, A., Hron, J., Kaplanis, C., Limont, M., McGill, M., Oliveira, Y., Parker-Holder, J., Perbet, F., Scully, G., Shar, J., Spencer, S., Tov, O., Villegas, R., Wang, E., Yung, J., Baetu, C., Berbel, J., Bridson, D., Bruce, J., Buttimore, G., Chakera, S., Chandra, B., Collins, P., Cullum, A., Damoc, B., Dasagi, V., Gazeau, M., Gbadamosi, C., Han, W., Hirst, E., Kachra, A., Kerley, L., Kjems, K., Knoepfel, E., Koriakin, V., Lo, J., Lu, C., Mehring, Z., Moufarek,

A., Nandwani, H., Oliveira, V., Pardo, F., Park, J., Pierson, A., Poole, B., Ran, H., Salimans, T., Sanchez, M., Saprykin, I., Shen, A., Sidhwani, S., Smith, D., Stanton, J., Tomlinson, H., Vijaykumar, D., Wang, L., Wingfield, P., Wong, N., Xu, K., Yew, C., Young, N., Zubov, V., Eck, D., Erhan, D., Kavukcuoglu, K., Hassabis, D., Gharamani, Z., Hadsell, R., van den Oord, A., Mosseri, I., Bolton, A., Singh, S., Rocktäschel, T.: Genie 3: A new frontier for world models (2025)
7. Bao, F., Xiang, C., Yue, G., He, G., Zhu, H., Zheng, K., Zhao, M., Liu, S., Wang, Y., Zhu, J.: Vidu: a highly consistent, dynamic and skilled text-to-video generator with diffusion models. arXiv preprint arXiv:2405.04233 (2024)
8. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)
9. Buch, S., Nagrani, A., Arnab, A., Schmid, C.: Flexible frame selection for efficient video reasoning. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 29071–29082 (2025)
10. Cai, S., Yang, C., Zhang, L., Guo, Y., Xiao, J., Yang, Z., Xu, Y., Yang, Z., Yuille, A., Guibas, L., Agrawala, M., Jiang, L., Wetzstein, G.: Mixture of contexts for long video generation. In: arXiv (2025)
11. Chen, B., Monso, D.M., Du, Y., Simchowitz, M., Tedrake, R., Sitzmann, V.: Diffusion forcing: Next-token prediction meets full-sequence diffusion. arXiv preprint arXiv:2407.01392 (2024)
12. Chen, T., Hu, X., Ding, Z., Jin, C.: Learning world models for interactive video generation. arXiv preprint arXiv:2505.21996 (2025)
13. Cui, J., Wu, J., Li, M., Yang, T., Li, X., Wang, R., Bai, A., Ban, Y., Hsieh, C.J.: Self-forcing++: Towards minute-scale high-quality video generation. arXiv preprint arXiv:2510.02283 (2025)
14. Cui, Y., Chen, H., Deng, H., Huang, X., Li, X., Liu, J., Liu, Y., Luo, Z., Wang, J., Wang, W., Wang, Y., Wang, C., Zhang, F., Zhao, Y., Pan, T., Li, X., Hao, Z., Ma, W., Chen, Z., Ao, Y., Huang, T., Wang, Z., Wang, X.: Emu3.5: Native multimodal models are world learners (2025), https://arxiv.org/abs/2510.26583
15. Decart, E.: Oasis: A universe in a transformer. https://oasis-model.github.io/ (2024)
16. DeepMind, G.: Veo 2: Our state-of-the-art video generation model. https://deepmind.google/technologies/veo/veo-2/ (2024)
17. Deng, H., Pan, T., Diao, H., Luo, Z., Cui, Y., Lu, H., Shan, S., Qi, Y., Wang, X.: Autoregressive video generation without vector quantization. arXiv preprint arXiv:2412.14169 (2024)
18. Fu, X., Liu, X., Wang, X., Peng, S., Xia, M., Shi, X., Yuan, Z., Wan, P., Zhang, D., Lin, D.: 3dtrajmaster: Mastering 3d trajectory for multi-entity motion in video generation. In: ICLR (2025)
19. Gu, Y., Mao, W., Shou, M.Z.: Long-context autoregressive video modeling with next-frame prediction. arXiv preprint arXiv:2503.19325 (2025)
20. Guo, Y., Yang, C., Yang, Z., Ma, Z., Lin, Z., Yang, Z., Lin, D., Jiang, L.: Long context tuning for video generation. arXiv preprint arXiv:2503.10589 (2025)
21. Ha, D., Schmidhuber, J.: Recurrent world models facilitate policy evolution **31** (2018)
22. He, H., Xu, Y., Guo, Y., Wetzstein, G., Dai, B., Li, H., Yang, C.: Cameractrl: Enabling camera control for text-to-video generation. arXiv preprint arXiv:2404.02101 (2024)
23. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)

24. Hong, Y., Mei, Y., Ge, C., Xu, Y., Zhou, Y., Bi, S., Hold-Geoffroy, Y., Roberts, M., Fisher, M., Shechtman, E., et al.: Relic: Interactive video world model with long-horizon memory. arXiv preprint arXiv:2512.04040 (2025)
25. Huang, X., Li, Z., He, G., Zhou, M., Shechtman, E.: Self forcing: Bridging the train-test gap in autoregressive video diffusion. arXiv preprint arXiv:2506.08009 (2025)
26. Kanervisto, A., Bignell, D., Wen, L.Y., Grayson, M., Georgescu, R., Valcarcel Macua, S., Tan, S.Z., Rashid, T., Pearce, T., Cao, Y., et al.: World and human action models towards gameplay ideation. Nature **638**(8051), 656–663 (2025)
27. Kim, S.W., Zhou, Y., Philion, J., Torralba, A., Fidler, S.: Learning to simulate dynamic environments with gamegan. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1231–1240 (2020)
28. Kingma, D.P., Welling, M., et al.: Auto-encoding variational bayes (2013)
29. Kling: Kling ai: Next-generation ai creative studio. `https://app.klingai.com/` (2024)
30. Kondratyuk, D., Yu, L., Gu, X., Lezama, J., Huang, J., Schindler, G., Hornung, R., Birodkar, V., Yan, J., Chiu, M.C., et al.: Videopoet: A large language model for zero-shot video generation. arXiv preprint arXiv:2312.14125 (2023)
31. Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603 (2024)
32. Labs, W.: Generating worlds. `https://www.worldlabs.ai/blog/generating-worlds` (2024)
33. Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: International conference on machine learning. pp. 19730–19742. PMLR (2023)
34. Li, R., Torr, P., Vedaldi, A., Jakab, T.: Vmem: Consistent interactive video scene generation with surfel-indexed view memory. arXiv preprint arXiv:2506.18903 (2025)
35. Li, T., Tian, Y., Li, H., Deng, M., He, K.: Autoregressive image generation without vector quantization. arXiv preprint arXiv:2406.11838 (2024)
36. Li, W., Pan, W., Luan, P.C., Gao, Y., Alahi, A.: Stable video infinity: Infinite-length video generation with error recycling. arXiv preprint arXiv:2510.09212 (2025)
37. Li, Z., Li, C., Mao, X., Lin, S., Li, M., Zhao, S., Xu, Z., Li, X., Feng, Y., Sun, J., Li, Z., Zhang, F., Ai, J., Wang, Z., Wu, Y., He, T., Pang, J., Qiao, Y., Jia, Y., Zhang, K.: Sekai: A video dataset towards world exploration. arXiv preprint arXiv:2506.15675 (2025)
38. Li, Z., Yu, H.X., Liu, W., Yang, Y., Herrmann, C., Wetzstein, G., Wu, J.: Wonderplay: Dynamic 3d scene generation from a single image and actions. In: Proceedings of the IEEE/CVF international conference on computer vision (2025)
39. Liu, K., Hu, W., Xu, J., Shan, Y., Lu, S.: Rolling forcing: Autoregressive long video diffusion in real time. arXiv preprint arXiv:2509.25161 (2025)
40. Ma, B., Gao, H., Deng, H., Luo, Z., Huang, T., Tang, L., Wang, X.: You see it, you got it: Learning 3d creation on pose-free videos at scale. arXiv preprint arXiv:2412.06699 (2024)
41. OpenAI: Creating video from text. `https://openai.com/index/sora/` (2024)
42. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2023)

43. Po, R., Nitzan, Y., Zhang, R., Chen, B., Dao, T., Shechtman, E., Wetzstein, G., Huang, X.: Long-context state-space video world models (2025), https://arxiv.org/abs/2505.20171
44. Ren, X., Shen, T., Huang, J., Ling, H., Lu, Y., Nimier-David, M., Müller, T., Keller, A., Fidler, S., Gao, J.: Gen3c: 3d-informed world-consistent video generation with precise camera control. arXiv preprint arXiv:2503.03751 (2025)
45. Runway: Runway : Tools for human imagination. https://runwayml.com/ (2024)
46. Song, K., Chen, B., Simchowitz, M., Du, Y., Tedrake, R., Sitzmann, V.: History-guided video diffusion. arXiv preprint arXiv:2502.06764 (2025)
47. Sun, W., Zhang, H., Wang, H., Wu, J., Wang, Z., Wang, Z., Wang, Y., Zhang, J., Wang, T., Guo, C.: Worldplay: Towards long-term geometric consistency for real-time interactive world modeling. arXiv preprint arXiv:2512.14614 (2025)
48. Valevski, D., Leviathan, Y., Arar, M., Fruchter, S.: Diffusion models are real-time game engines. arXiv preprint arXiv:2408.14837 (2024)
49. Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., Zeng, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
50. Wang, J., Yuan, Y., Zheng, R., Lin, Y., Gao, J., Chen, L.Z., Bao, Y., Zhang, Y., Zeng, C., Zhou, Y., Long, X., Zhu, H., Zhang, Z., Cao, X., Yao, Y.: Spatialvid: A large-scale video dataset with spatial annotations (2025), https://arxiv.org/abs/2509.09676
51. Wang, X., Zhang, X., Luo, Z., Sun, Q., Cui, Y., Wang, J., Zhang, F., Wang, Y., Li, Z., Yu, Q., et al.: Emu3: Next-token prediction is all you need. arXiv preprint arXiv:2409.18869 (2024)
52. Wang, Z., Yuan, Z., Wang, X., Li, Y., Chen, T., Xia, M., Luo, P., Shan, Y.: Motionctrl: A unified and flexible motion controller for video generation. In: ACM SIGGRAPH 2024 Conference Papers (2024)
53. Wu, T., Yang, S., Po, R., Xu, Y., Liu, Z., Lin, D., Wetzstein, G.: Video world models with long-term spatial memory (2025), https://arxiv.org/abs/2506.05284
54. Wu, Z., Xiong, C., Ma, C.Y., Socher, R., Davis, L.S.: Adaframe: Adaptive frame selection for fast video recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1278–1287 (2019)
55. Xiao, Z., Lan, Y., Zhou, Y., Ouyang, W., Yang, S., Zeng, Y., Pan, X.: Worldmem: Long-term consistent world simulation with memory. arXiv preprint arXiv:2504.12369 (2025)
56. Yan, W., Zhang, Y., Abbeel, P., Srinivas, A.: Videogpt: Video generation using vq-vae and transformers. arXiv preprint arXiv:2104.10157 (2021)
57. Yang, S., Walker, J.C., Parker-Holder, J., Du, Y., Bruce, J., Barreto, A., Abbeel, P., Schuurmans, D.: Position: Video as the new language for real-world decision making. In: Proceedings of the 41st International Conference on Machine Learning (2024)
58. Yang, S., Huang, W., Chu, R., Xiao, Y., Zhao, Y., Wang, X., Li, M., Xie, E., Chen, Y., Lu, Y., Chen, S.H.Y.: Longlive: Real-time interactive long video generation (2025)
59. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al.: Cogvideox: Text-to-video diffusion models with an expert transformer. arXiv preprint arXiv:2408.06072 (2024)
60. Yu, H.X., Duan, H., Herrmann, C., Freeman, W.T., Wu, J.: Wonderworld: Interactive 3d scene generation from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5916–5926 (June 2025)

61. Yu, H.X., Duan, H., Hur, J., Sargent, K., Rubinstein, M., Freeman, W.T., Cole, F., Sun, D., Snavely, N., Wu, J., et al.: Wonderjourney: Going from anywhere to everywhere. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6658–6667 (2024)
62. Yu, J., Bai, J., Qin, Y., Liu, Q., Wang, X., Wan, P., Zhang, D., Liu, X.: Context as memory: Scene-consistent interactive long video generation with memory retrieval. arXiv preprint arXiv:2506.03141 (2025)
63. Yu, J., Qin, Y., Che, H., Liu, Q., Wang, X., Wan, P., Zhang, D., Gai, K., Chen, H., Liu, X.: A survey of interactive generative video. arXiv preprint arXiv:2504.21853 (2025)
64. Yu, J., Qin, Y., Che, H., Liu, Q., Wang, X., Wan, P., Zhang, D., Liu, X.: Position: Interactive generative video as next-generation game engine. arXiv preprint arXiv:2503.17359 (2025)
65. Yu, J., Qin, Y., Wang, X., Wan, P., Zhang, D., Liu, X.: Gamefactory: Creating new games with generative interactive videos (2025)
66. Yu, W., Xing, J., Yuan, L., Hu, W., Li, X., Huang, Z., Gao, X., Wong, T.T., Shan, Y., Tian, Y.: Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. arXiv preprint arXiv:2409.02048 (2024)
67. Zhang, L., Agrawala, M.: Packing input frame context in next-frame prediction models for video generation. arXiv preprint arXiv:2504.12626 (2025)
68. Zhou, G., Pan, H., LeCun, Y., Pinto, L.: Dino-wm: World models on pre-trained visual features enable zero-shot planning. arXiv preprint arXiv:2411.04983 (2024)
69. Zhou, S., Du, Y., Yang, Y., Han, L., Chen, P., Yeung, D.Y., Gan, C.: Learning 3d persistent embodied world models. arXiv preprint arXiv:2505.05495 (2025)
70. Zhou, Y., Wang, Y., Zhou, J., Chang, W., Guo, H., Li, Z., Ma, K., Li, X., Wang, Y., Zhu, H., Liu, M., Liu, D., Yang, J., Fu, Z., Chen, J., Shen, C., Pang, J., Zhang, K., He, T.: Omniworld: A multi-domain and multi-modal dataset for 4d world modeling (2025), https://arxiv.org/abs/2509.12201
71. Zhu, F., Wu, H., Guo, S., Liu, Y., Cheang, C., Kong, T.: Irasim: Learning interactive real-robot action simulators. arXiv preprint arXiv:2406.14540 (2024)