

Spatia: Video Generation with Updatable Spatial Memory

Jinjing Zhao^{1*} Fangyun Wei^{2*} Zhening Liu³ Hongyang Zhang⁴ Chang Xu¹ Yan Lu²

¹The University of Sydney ²Microsoft Research Asia

³Hong Kong University of Science and Technology ⁴University of Waterloo

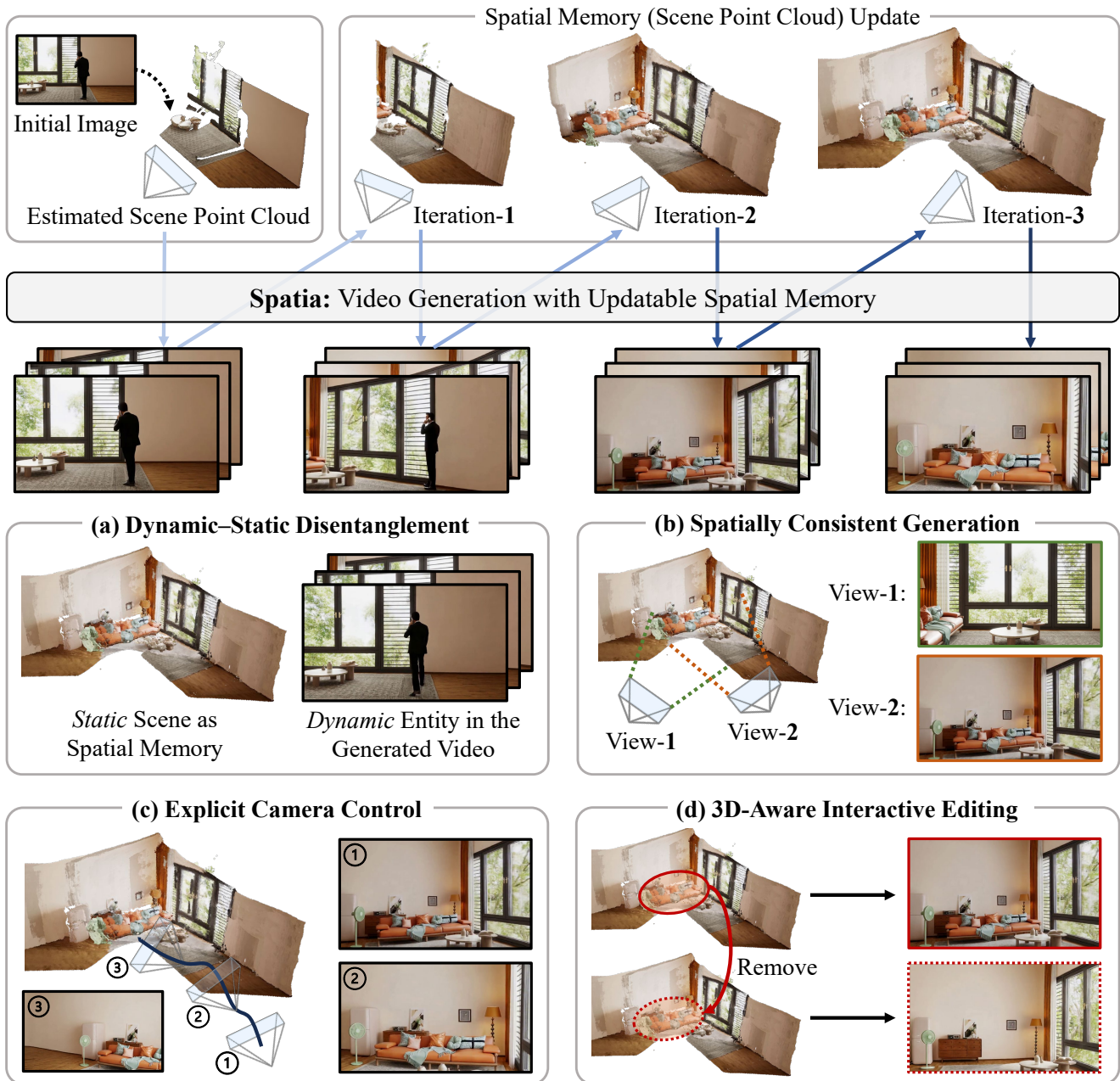


Figure 1. **Spatia** maintains a scene point cloud as its spatial memory and conditions on it throughout the iterative video generation process. It enables: (a) dynamic–static disentanglement by treating the static scene as spatial memory while generating videos that include dynamic entities; (b) spatially consistent generation across multiple views; (c) explicit camera control via 3D-aware trajectory rendering; and (d) 3D-aware interactive editing, allowing users to modify or remove scene elements prior to video generation.

*Equal contribution.

Abstract

Existing video generation models struggle to maintain long-term spatial and temporal consistency due to the dense, high-dimensional nature of video signals. To overcome this limitation, we propose Spatia, a spatial memory-aware video generation framework that explicitly preserves a 3D scene point cloud as persistent spatial memory. Spatia iteratively generates video clips conditioned on this spatial memory and continuously updates it through visual SLAM. This dynamic-static disentanglement design enhances spatial consistency throughout the generation process while preserving the model’s ability to produce realistic dynamic entities. Furthermore, Spatia enables applications such as explicit camera control and 3D-aware interactive editing, providing a geometrically grounded framework for scalable, memory-driven video generation.

1. Introduction

Video generation has emerged as a foundational technique powering a wide spectrum of tasks. On one hand, recent advances in video generation foundation models [30, 31, 35, 46, 47, 65, 66, 76, 84, 87, 106] have significantly improved the quality and controllability of short-duration video synthesis. On the other hand, there is a growing need to extend these models toward long-horizon video generation, enabling applications that require *temporal consistency* and *persistent memory*, such as world models [2, 17, 24, 68, 86, 96, 108, 109, 116], AI-driven game generation [10, 12, 18, 27, 81, 103, 111, 117], and embodied AI [8, 36, 45, 50, 52, 78, 85, 94, 119].

Unlike LLMs [1, 5, 57, 64, 67, 83, 100], video generation models encounter intrinsic difficulties in encoding long-term historical information, primarily due to the dense and high-dimensional nature of video signals. For instance, a short 5-second 480P (640×480) video at 24 FPS—consisting of 120 frames—already corresponds to $40 \times 30 \times 30 = 36,000$ spatio-temporal tokens when using a video encoder [46, 87] with a spatial downsampling factor of 16 and a temporal downsampling factor of 4. Including even one additional 5-second video clip as context would dramatically increase the computational and memory demands, rendering it impractical to directly model minute- or hour-scale temporal contexts, which could easily span millions of tokens.

By comparison, 36,000 tokens can represent around 27,000 words¹. In other words, with the same number of tokens, a video generation model can capture only about 5 seconds of visual history, whereas an LLM can encompass a context equivalent to 27,000 words. Therefore, unlike LLMs that can directly attend to all historical text tokens,

¹A word is represented by an average of 1.3 tokens using GPT-3’s text tokenizer.

video generation models must rely on alternative mechanisms to preserve memory and encode contextual dependencies without simply modeling the entire sequence of historical spatio-temporal tokens.

In this work, we introduce an explicit memory mechanism designed to achieve consistent and long-horizon video generation, particularly in scenarios where the same location reappears multiple times during the generation process. As illustrated in Figure 1, we take the image-to-video task as an illustrative example. The process begins by estimating an initial 3D scene point cloud from the conditional input image, which serves as the spatial memory of the scene. We then iteratively perform two key steps:

1. Generate a new video clip conditioned on both the current 3D scene point cloud and the previously generated video clip, ensuring temporal and spatial consistency across iterations.
2. Update the scene point cloud using visual SLAM algorithms based on both newly generated and previously generated frames, thereby incorporating new content while preserving existing scene information.

This iterative update enables the system to maintain scene consistency and geometric coherence over long sequences, allowing the model to effectively “remember” previously visited locations. As a result, it can generate videos with realistic long-term structural continuity, yielding a persistent spatial memory of the scene. We name our approach as **Spatia**, short for spatial memory-aware video generation. Spatia enjoys the following key characteristics, which arise from the integration of the spatial memory mechanism:

- *Dynamic-Static Disentanglement* (Figure 1(a)). Spatia preserves a scene point cloud as spatial memory while simultaneously generating dynamic entities that interact coherently with the scene. This contrasts with previous methods [40, 51, 112, 113] addressing the video generation memory problem, which are typically limited to producing videos with static scenes only.
- *Spatially Consistent Generation* (Figure 1(b)). By retrieving spatial memory, Spatia can generate diverse video sequences depicting the same location from different viewpoints while preserving a consistent spatial structure.
- *Explicit Camera Control* (Figure 1(c)). Unlike previous approaches [6, 28, 34, 37, 38, 51, 104] that encode camera trajectories into latent features and inject them into video generation models—an indirect strategy that may result in inaccurate or unstable control—Spatia achieves camera control in a more explicit and geometrically grounded manner. Mirroring the rendering process of 3DGS [43], it directly applies the desired camera path to the 3D scene point cloud and renders a corresponding 2D point cloud sequence, which serves as a conditioning signal to guide video generation along the specified camera trajectory.
- *3D-Aware Interactive Editing* (Figure 1(d)). Since Spa-

tia conditions video generation on the 3D scene point cloud, users can interactively edit the scene before generation—for example, by removing or modifying specific objects. Such edits are directly reflected in the generated videos, enabling intuitive and fine-grained control over scene composition and content.

We experimentally demonstrate that Spatia, equipped with the proposed spatial memory mechanism, significantly enhances spatial consistency throughout the generation process, without compromising its ability to produce dynamic entities or the visual quality of generated videos. Additional benefits include enabling long-horizon generation and supporting applications such as spatial editing.

2. Related Works

Video Generation Models. The field of video generation has evolved rapidly, progressing from early UNet-based latent diffusion models [7, 15, 16, 114] to large-scale Diffusion Transformers [25, 69]. This architectural transition has given rise to a new generation of powerful foundation models, including open-source systems [46, 63, 84, 87, 106] and high-performance proprietary counterparts [31, 47, 66]. While bidirectional models employing global spatio-temporal attention achieve impressive fidelity [46, 55, 63, 65, 70, 76, 87, 120], their quadratic computational complexity fundamentally limits them to short-clip generation. To generate arbitrarily long sequences, autoregressive frameworks [2, 29, 39, 41, 44, 86, 95] have been proposed, which iteratively synthesize new content conditioned on previously generated frames. Subsequent studies [13, 14, 32, 59, 77, 80, 97, 107] further address the issue of error accumulation in long-horizon generation. While these methods achieve strong temporal coherence, they still lack an explicit spatial memory mechanism.

Camera Control in Video Generation. Precise camera control has become a key goal in video synthesis. One line of work conditions generation on explicit camera parameters—for example, AnimateDiff [34] employs motion LoRAs to learn specific camera trajectories. Other methods incorporate various camera representations, such as point trajectories or Plücker embeddings [79], directly into the generator [28, 37, 38, 51, 104, 121]. For finer-grained control, geometry-aware approaches use 3D information—such as rendered point clouds—to provide dense spatial guidance for camera path generation [33, 75, 105, 112, 113]. Meanwhile, video editing-based methods achieve controllability by re-targeting existing footage to new viewpoints [6] or by transferring camera motion from reference videos [61].

Long-term Memory Modeling. A central strategy for improving the long-term memory capacity of LLMs lies in expanding their native context window, which has grown dramatically—from the limited spans of early models [9, 20, 71–73] to the million- or even ten-million-token ranges

achieved by modern architectures [1, 3, 4, 19, 64, 67, 82]—enabled by techniques like KV-cache compression [21, 48, 53, 58, 98, 118]. In video generation, the bidirectional spatiotemporal attention used in most diffusion models prevents standard KV caching, thereby severely limiting the context window and restricting access to previously generated content. To preserve long-term spatial consistency, recent works have introduced memory-based architectures. For maintaining spatial coherence in explorable 3D scenes, methods such as [17, 40, 60, 62, 113] leverage progressive expansion or warping pipelines to refine global scene geometry. Context-as-Memory [110] retrieves previous frames based on camera FOV overlap, while VMem [51] introduces a surfel-indexed view memory for efficient geometric indexing and retrieval of past views.

Scene Point Cloud Estimation. Recent progress in visual geometry estimation is led by Dust3R [92], which unifies pairwise pose and geometry estimation but encounters a costly $\mathcal{O}(N^2)$ global alignment bottleneck when reconstructing an N -view input sequence. This limitation motivates follow-up works [11, 49, 91, 102] to develop more scalable solutions by introducing efficient sequential or parallel architectures. In parallel, universal end-to-end models [42, 88, 89, 93] eliminate the pairwise dependency, employing large Transformers to infer globally consistent 3D geometry and camera parameters for all views in a single forward pass.

3. Method

Problem Formulation. The objective of Spatia is to endow a video generation model with persistent spatial memory, enabling it to produce videos that are both spatially and temporally consistent. To achieve this, Spatia maintains and iteratively updates a static scene point cloud throughout the generation process. This point cloud serves as an explicit geometric memory that anchors all generated content within a coherent spatial layout. Spatia formulates the entire framework as a multi-modal conditional generation problem, where generation is conditioned on textual instructions, spatial memory, and temporal context. Specifically, the framework operates in two stages:

1. Generating a video clip conditioned on multi-modal inputs—including text instructions (for instruction following), geographically retrieved information from the spatial memory (for spatial consistency), and either an initial image or previously generated clips (for temporal continuity).
2. Updating the spatial memory to incorporate newly generated content, ensuring that subsequent generations remain geometrically consistent with the evolving scene.

Note that the above stages can be performed iteratively to enable long-horizon generation. Sections 3.1 and 3.2 detail the training and inference of Spatia, respectively.

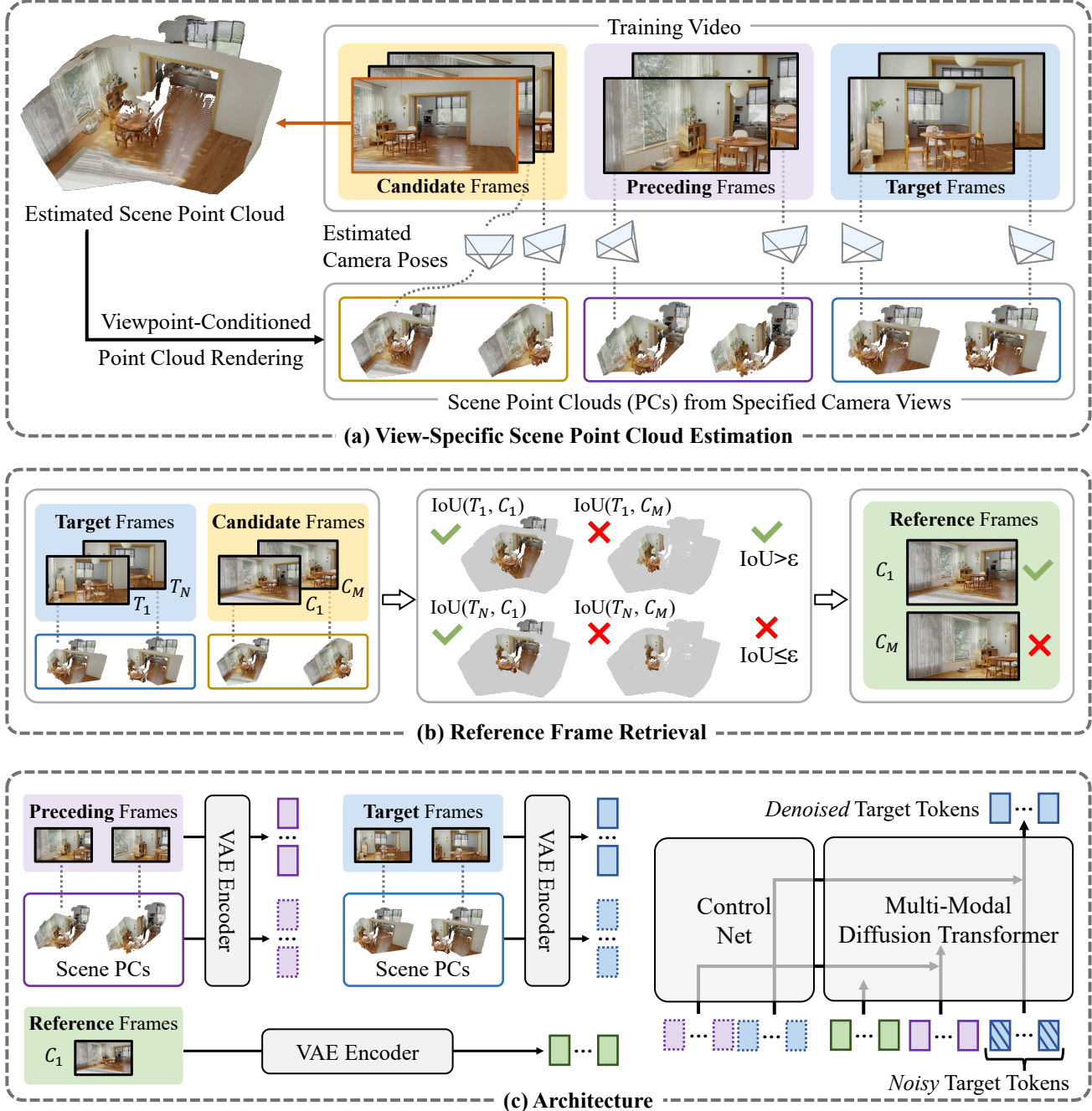


Figure 2. Overview of the training stage of Spatia. Each training video is divided into a target clip, a preceding clip, and a candidate-frame set. Text tokens are omitted for simplicity. (a) A frame is randomly selected from the candidate-frame set to estimate a 3D scene point cloud \mathcal{S} . Using the estimated camera poses together with \mathcal{S} , we then generate view-specific scene point cloud sequences for both the target and preceding clips. (b) The most spatially relevant frames are then retrieved from the candidate-frame set as reference frames. (c) The spatial conditions obtained from (a) and (b) guide the video generation process. The detailed network architecture is provided in Figure 3.

3.1. Training

Training Data. We address the text-and-image-to-video generation problem, where each training sample consists of a video \mathcal{V} paired with a textual description \mathcal{T} that narrates its content. For a given training video \mathcal{V} , we decompose it into three parts, $\mathcal{V} = \{\mathcal{T}\}^N \cup \{\mathcal{P}\}^M \cup \{\mathcal{C}\}^O$, where $\{\mathcal{T}\}^N$,

$\{\mathcal{P}\}^M$, and $\{\mathcal{C}\}^O$ denote the target-frame set, preceding-frame set, and candidate-frame set, containing N , M and O frames, respectively. Specifically, we randomly select one clip from \mathcal{V} as the target clip $\{\mathcal{T}\}^N$, in which each \mathcal{T} represents a target frame to be generated by the model. The clip immediately preceding $\{\mathcal{T}\}^N$ is defined as the preced-

ing clip $\{P\}^M$, with P referring to a single frame providing temporal context. The remaining frames within \mathcal{V} , excluding those in $\{T\}^N$ and $\{P\}^M$, are treated as candidate frames $\{C\}^O$, which serve as potential references for spatial and geometric consistency.

Overview. Figure 2 illustrates the overall training pipeline, which can be divided into two main parts: data pre-processing—including *View-Specific Scene Point Cloud Estimation* (Section 3.1.1) and *Reference Frame Selection* (Section 3.1.2)—and the model *Architecture* (Section 3.1.3), formulated as a multi-modal conditional generation framework.

3.1.1. View-Specific Scene Point Cloud Estimation

Scene Point Cloud Estimation. As shown in Figure 2(a), we first randomly sample a frame from the candidate-frame set $\{C\}^O$ and employ MapAnything [42] to estimate a scene point cloud \mathcal{S} . Note that if the training video \mathcal{V} contains dynamic entities, we perform a segmentation process to remove these entities before point cloud estimation. Specifically, we first utilize Keye-VL-1.5 [101] to identify dynamic entities and generate corresponding text prompts for each detected entity. Then, we apply ReferDINO [54] to segment out these dynamic entities, ensuring that the resulting point cloud \mathcal{S} represents only the static components of the scene.

Per-Frame Camera Pose Estimation. Next, we estimate the camera pose for each frame in $\{T\}^N \cup \{P\}^M \cup \{C\}^O$ using MapAnything [42]. The corresponding per-frame camera poses are denoted as $\{\theta_T\}^N$, $\{\theta_P\}^M$ and $\{\theta_C\}^O$.

View-Specific Scene Point Clouds. Given the estimated scene point cloud \mathcal{S} and the per-frame camera poses $\{\theta_T\}^N$, $\{\theta_P\}^M$ and $\{\theta_C\}^O$, we apply each camera pose to \mathcal{S} to render the scene from the corresponding viewpoint, as illustrated in Figure 2(a). The resulting view-specific scene point clouds are denoted as $\{\mathcal{S}_T\}^N$, $\{\mathcal{S}_P\}^M$ and $\{\mathcal{S}_C\}^O$, respectively.

3.1.2. Reference Frame Retrieval

The objective of this stage is to select up to K of the most spatially relevant frames from the candidate-frame set $\{C\}^O$ as reference frames for the target clip $\{T\}^N$. Reference frames are defined as those that exhibit spatial overlap with $\{T\}^N$, depicting similar regions or viewpoints within the scene. These frames provide additional spatial cues that enhance geometric consistency during video generation.

To identify these reference frames, we compute spatial correspondence between $\{T\}^N$ and $\{C\}^O$ using their associated scene point clouds (i.e., $\{\mathcal{S}_T\}^N$ and $\{\mathcal{S}_C\}^O$). The detailed retrieval process is illustrated in Figure 2(b) and presented in Algorithm 1 in the appendix. The retrieved reference-frame set is denoted as $\{R\}^K \subset \{C\}^O$, where K represents the maximum number of reference frames.

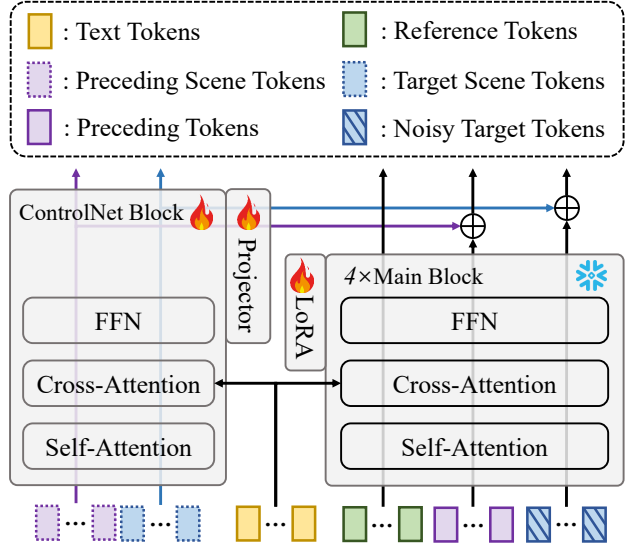


Figure 3. Illustration of a single network block composed of one ControlNet [115] block operating in parallel with four main blocks. Detailed definitions of all token types are provided in Figure 2.

3.1.3. Architecture

Figure 2(c) illustrates the architecture of Spatia, which adopts a multi-modal conditional generation framework. The objective is to generate the target video clip $\{T\}^N$ conditioned on the preceding video clip $\{P\}^M$, their corresponding scene point clouds $\{\mathcal{S}_T\}^N$ and $\{\mathcal{S}_P\}^M$, the retrieved reference frames $\{R\}^K$ and the text instruction \mathcal{T} . **Token Extraction.** For the video-modality inputs, $\{T\}^N$ and $\{P\}^M$, we employ the Wan2.2 [87] video encoder to convert them into spatio-temporal tokens, denoted as X_T and X_P , respectively. Since this video encoder can also process single-frame images, we use it to encode the image-modality inputs, i.e., the reference frames $\{R\}^K$. The resulting token sequence, denoted as X_R , is obtained by independently encoding each reference frame and concatenating the resulting tokens along the sequence dimension.

As described in Section 3.1.1, $\{\mathcal{S}_T\}^N$ and $\{\mathcal{S}_P\}^M$ represent the 3D scene point cloud sequences associated with the target video clip $\{T\}^N$ and the preceding video clip $\{P\}^M$, respectively. To encode $\{\mathcal{S}_T\}^N$ and $\{\mathcal{S}_P\}^M$, each sequence is projected onto the 2D image plane, resulting in a pair of scene projection videos. Missing pixel values in these projection videos are filled with zeros. Both are then processed by the same video encoder, yielding latent representations denoted as X_{S_T} and X_{S_P} , respectively.

At last, to encode the text instruction \mathcal{T} , we follow Wan2.2 [87] and employ its text encoder to obtain the corresponding text tokens, denoted as $X_{\mathcal{T}}$.

Network Structure. Figure 3 presents the detailed architecture of our network, serving as a complementary illustration to Figure 2(c). We adopt Flow Matching [56] for model

training under multiple conditioning signals—including text tokens \mathbf{X}_T , reference tokens \mathbf{X}_R , preceding video tokens \mathbf{X}_P , preceding scene video tokens \mathbf{X}_{S_P} , and target scene video tokens \mathbf{X}_{S_T} —to guide the generation process from pure noise toward the target video tokens \mathbf{X}_T .

Concretely, given target video tokens \mathbf{X}_T , we first sample $t \in [0, 1]$ from a logit-normal distribution and initialize the noise $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ following a Gaussian distribution. The intermediate sample $\mathbf{x}_t = (1 - t)\mathbf{x}_0 + t\mathbf{X}_T$ is then obtained via linear interpolation. The model is trained to predict the velocity $\mathbf{u}_t = d\mathbf{x}_t/dt$ by minimizing the mean squared error between the predicted velocity \mathbf{v}_t and the ground-truth velocity \mathbf{u}_t :

$$\mathcal{L} = \mathbb{E}_{t, \mathbf{x}_0, \mathbf{X}_T} \|\mathbf{v}_t - \mathbf{u}_t\|^2. \quad (1)$$

Spatia includes 8 network blocks, each containing one ControlNet [115] block operating in parallel with four main blocks, as illustrated in Figure 3. Each main block follows the design of Wan2.2 [87], consisting of a self-attention layer, a cross-attention layer, and an FFN. Each ControlNet block adopts the same architecture but appends a projector—implemented as a simple MLP layer—after the FFN.

The first ControlNet block processes the concatenation of \mathbf{X}_{S_P} and \mathbf{X}_{S_T} . The resulting outputs are passed to the subsequent ControlNet block and, after projection through a simple MLP layer, the projected features, denoted as \mathbf{X}'_{S_P} and \mathbf{X}'_{S_T} , are also fed into the corresponding main block. Meanwhile, the text tokens \mathbf{X}_T are incorporated via the cross-attention layer.

The first main block takes the concatenation of \mathbf{X}_R , \mathbf{X}_P , and \mathbf{x}_t as input. The concatenated tokens are sequentially processed through a stack of layers, including self-attention, cross-attention, and FFN. In the cross-attention layer, the text tokens serve as keys and values, allowing semantic conditioning on textual instructions. This process yields updated features denoted as \mathbf{X}'_R , \mathbf{X}'_P , and \mathbf{x}'_t . To integrate scene-level spatial context, the outputs from the associated ControlNet block, \mathbf{X}'_{S_P} and \mathbf{X}'_{S_T} , are fused into the corresponding features via simple addition, resulting in $\mathbf{X}'_P + \mathbf{X}'_{S_P}$ and $\mathbf{x}'_t + \mathbf{X}'_{S_T}$. Together with \mathbf{X}'_R , these form the outputs of the block.

3.2. Inference

Spatia enables iterative user interaction. At each iteration, the user specifies a text instruction and a camera trajectory based on the current 3D scene point cloud to generate a new video clip. The newly generated content, together with previously produced clips, is then used to update the spatial memory (scene point cloud). This iterative process continues, as illustrated in Figure 4.

4. Experiment

Implementation Details. Our network backbone is initialized from Wan2.2 [87], containing 5B parameters.

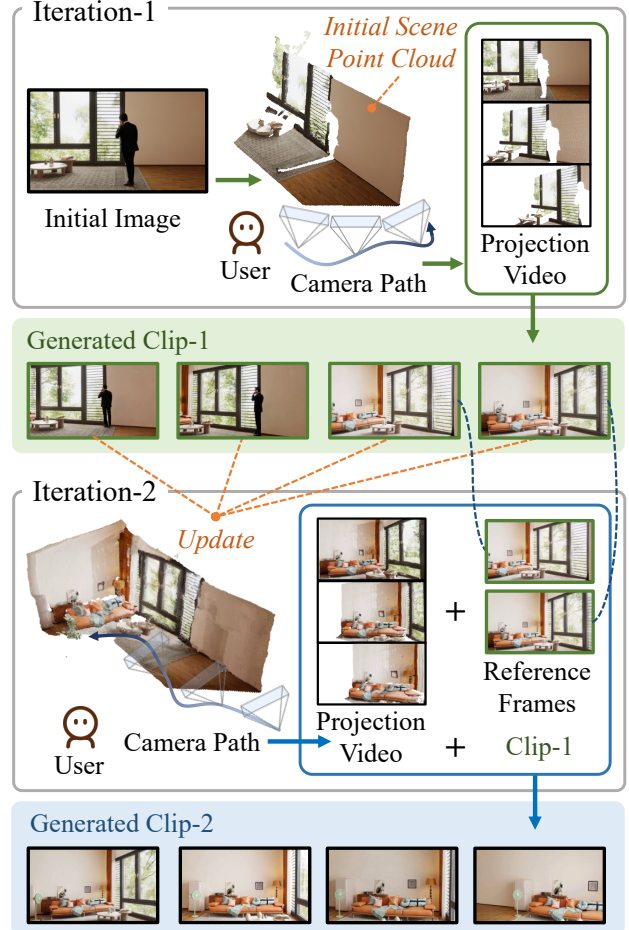


Figure 4. Illustration of the Spatia inference process. At the first iteration, the user provides an initial image, from which Spatia estimates the initial 3D scene point cloud. The user then specifies a text instruction and a camera path based on the estimated scene, producing a projection video along the desired trajectory that conditions the generation of clip-1. In subsequent iterations, two steps are performed: (1) Spatia updates the spatial memory (3D scene point cloud) using all previously generated frames via MapAnything [42]; and (2) the user specifies a new text instruction and camera path based on the updated scene. Spatia then takes the reference frames (generated as described in Section 3.1.2), the previously generated clip, and the new projection video as input to produce the next video clip. *Text instructions are omitted.*

Each ControlNet block is initialized from its corresponding main block. The training set consists of two sources: RealEstate [122] (40K training videos) and SpatialVID [90] (HD subset, 10K videos), both at 720P resolution. We first train the ControlNet blocks for 8,000 iterations while freezing the main network. Next, we freeze the ControlNet blocks and fine-tune the main blocks using LoRA (rank = 64) for 5,000 iterations. Both stages adopt the AdamW optimizer, with learning rates of 1e-5 and 1e-4, respectively, and a batch size of 64 on 64× AMD MI250 GPUs. By default, the model generates 81 frames for the first (image-

Table 1. Visual quality comparison on the *WorldScore* benchmark. The final **Static** and **Dynamic** world scores are computed by aggregating all relevant metrics. The **Average** score represents the mean of the static and dynamic world scores. *Static scene generation models* cannot handle dynamic entities, while *foundation video generation models* typically lack persistent memory mechanisms.

Method	Average Score	Static Score	Dynamic Score	Camera Ctrl	Object Ctrl	Content Align	3D Const	Photo Const	Style Const	Subject Quality	Motion Acc	Motion Mag	Motion Smooth
<i>Static scene generation models</i>													
WonderJourney [109]	54.19	63.75	44.63	84.60	37.10	35.54	80.60	79.03	62.82	<u>66.56</u>	-	-	-
InvisibleStitch [24]	51.95	61.12	42.78	93.20	36.51	29.53	88.51	89.19	32.37	58.50	-	-	-
WonderWorld [108]	61.79	<u>72.69</u>	50.88	<u>92.98</u>	51.76	<u>71.25</u>	<u>86.87</u>	85.56	70.57	49.81	-	-	-
Voyager [40]	<u>66.08</u>	77.62	54.53	85.95	<u>66.92</u>	68.92	81.56	85.99	84.89	71.09	-	-	-
<i>Foundation video generation models</i>													
VideoCrafter2 [16]	50.03	52.57	47.49	28.92	39.07	72.46	65.14	61.85	43.79	56.74	47.12	30.40	29.39
EasyAnimate [99]	52.25	52.85	51.65	26.72	54.50	50.76	67.29	47.35	73.05	50.31	<u>75.00</u>	31.16	40.32
Allegro [123]	53.64	55.31	51.97	24.84	<u>57.47</u>	51.48	70.50	69.89	65.60	47.41	54.39	40.28	37.81
CogVideoX-I2V [106]	60.64	62.15	<u>59.12</u>	38.27	40.07	36.73	86.21	88.12	<u>83.22</u>	62.44	69.56	26.42	60.15
Vchitect-2.0 [26]	40.38	42.28	38.47	26.55	49.54	65.75	41.53	42.30	25.69	44.58	33.59	<u>33.81</u>	21.31
LTX-Video [35]	55.99	55.44	56.54	25.06	53.41	39.73	78.41	88.92	53.50	49.08	76.22	29.95	<u>71.09</u>
Wan2.1 [87]	55.21	57.56	52.85	23.53	40.32	45.44	78.74	78.36	77.18	59.38	54.27	33.26	38.05
Spatia (Ours)	69.73	72.63	66.82	75.66	52.32	69.95	86.40	<u>89.10</u>	80.09	54.86	54.83	24.75	80.26

conditioned) iteration and 72 frames for each subsequent (clip-conditioned) iteration, conditioned on 9 previously generated frames.

Evaluation. We evaluate our model from two aspects: (1) visual quality and (2) memory mechanism effectiveness. For (1), we adopt two benchmarks—WorldScore [22] and the RealEstate [122] test set. The WorldScore benchmark provides 3,000 test samples for text/image-to-video generation and includes a comprehensive suite of metrics to assess both static and dynamic visual quality. For the RealEstate test set, we randomly sample 100 videos, use the first frame of each as the conditioning image, generate corresponding videos, and report PSNR, SSIM, and LPIPS scores against the original videos. For (2), we randomly select 100 samples from the WorldScore benchmark and use each sample’s initial image to generate a closed-loop video—where the camera trajectory brings the final frame back to the initial viewpoint. We then compare the final frame with the initial image using PSNR, SSIM, and LPIPS.

4.1. Main Results

Visual Quality. Using the WorldScore [22] benchmark, we evaluate visual quality across three categories of models: (1) *static scene generation models*, which inherently preserve spatial consistency by producing explorable static worlds yet cannot capture motion dynamics; (2) *foundation video generation models*, which generally lack explicit memory mechanisms but effectively generate dynamic content; and (3) *our approach*—a video generation model endowed with spatial memory that integrates dynamic motion generation with long-term spatial coherence. Table 1 presents the comparison results of *Spatia* against the other two categories of models.

Meanwhile, for methods that cannot accept control sig-

Table 2. Evaluation on *RealEstate*. We reproduce the results of all baseline methods using their default configurations and evaluate them on the same test samples to ensure a fair comparison.

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
SEVA [121]	13.07	0.515	0.445
VMem [51]	14.62	0.522	0.426
ViewCrafter [113]	15.78	0.580	0.396
FlexWorld [17]	16.25	0.593	0.370
Voyager [40]	17.79	0.636	0.297
Spatia (Ours)	18.58	0.646	0.254

nals from the WorldScore benchmark or have not reported results on it, we evaluate them on the constructed RealEstate [122] test set. The results are presented in Table 2. Since RealEstate provides ground-truth videos, we report PSNR, SSIM, and LPIPS by comparing the generated videos against the corresponding ground-truths.

Memory Mechanism Evaluation. Since few existing methods address video generation with spatial memory, we evaluate our model against scene generation approaches that explicitly maintain spatial memory. The evaluation is conducted on a subset of the WorldScore benchmark containing 100 randomly selected samples. Specifically, we design a *closed-loop* setting where each sample’s initial image is used to generate a video in which the camera trajectory brings the final frame back to the initial viewpoint. We then report PSNR_C, SSIM_C, and LPIPS_C, denoting PSNR, SSIM, and LPIPS between the final frame and the initial image. In addition, we introduce an evaluation metric called Match Accuracy, which measures dense correspondences between the final frame and the initial image—where higher values indicate better spatial alignment. Details are provided in the appendix, and the results are shown in Table 3.

Table 3. *Memory Mechanism Evaluation on the WorldScore Subset*. Each test sample includes a ground-truth initial image. Using this image, we require the model to generate a *closed-loop* video, where the camera in the final frame returns to the initial viewpoint. We then compute PSNR, SSIM, LPIPS, and Match Accuracy between the final frame and the initial image to evaluate spatial memory consistency.

Method	PSNR _C ↑	SSIM _C ↑	LPIPS _C ↓	Match Acc ↑
ViewCrafter [113]	14.79	0.481	0.365	0.447
FlexWorld [17]	12.20	0.428	0.598	0.377
Voyager [40]	17.66	0.540	0.380	0.507
Spatia (Ours)	19.38	0.579	0.213	0.698

Table 4. Impact of incorporating scene projection videos and reference frames on spatial memory modeling. The “Camera Control” metric is adopted from the WorldScore benchmark.

Scene Reference Video	Reference Frames	Camera Control	PSNR _C	SSIM _C	LPIPS _C
		58.81	15.55	0.444	0.379
✓		80.13	17.18	0.500	0.295
	✓	61.38	15.64	0.444	0.393
✓	✓	84.47	19.38	0.579	0.213

Table 5. Effects of using different numbers of reference frames.

#Reference Frames	PSNR _C	SSIM _C	LPIPS _C	Match Acc
1	17.50	0.537	0.284	0.592
3	17.85	0.540	0.275	0.606
5	18.48	0.556	0.248	0.640
7	19.38	0.579	0.213	0.698

4.2. Ablation Studies

Ablation studies related to spatial memory are conducted on the WorldScore subset, with closed-loop videos generated to evaluate spatial memory, as described in Section 4.1. Other studies focusing on visual quality are performed on the RealEstate test set.

Spatial Memory. Given a camera trajectory, the previously generated frames, and the current spatial memory (i.e., the scene point cloud), we: (1) render a scene projection video along the specified camera path, and (2) retrieve up to K reference frames from prior generations that are spatially correlated with the current trajectory. The rendered scene video and the retrieved reference frames are then provided to our network to facilitate the next-step, memory-aware video generation. The impact of incorporating scene projection videos and reference frames is analyzed in Table 4.

Reference Frame Number. Table 5 analyzes the effect of varying the number of reference frames K . Increasing K provides more spatial memory cues; however, we observe no significant performance improvement when $K > 7$.

Long-Horizon Generation. Memory is essential for long-

Table 6. Memory mechanisms ensure spatial consistency and preserve visual quality in long-horizon generation.

Method	#Clips	Camera Control	PSNR _C	SSIM _C	LPIPS _C
Wan2.2 [87]	2	56.87	13.00	0.377	0.521
	4	46.43	11.32	0.328	0.611
	6	49.97	10.74	0.310	0.644
Spatia (Ours)	2	84.47	19.38	0.579	0.213
	4	83.97	18.23	0.546	0.253
	6	83.41	18.04	0.541	0.259

Table 7. Impact of point cloud density on visual quality. Metrics are computed between the generated videos and the ground-truth videos on the RealEstate test set.

Cube Side Length (m)	PSNR	SSIM	LPIPS
0.01	18.58	0.646	0.254
0.03	17.10	0.614	0.313
0.05	16.35	0.596	0.349
0.07	15.97	0.585	0.370

horizon generation, ensuring spatial consistency when the camera revisits the same location without visual degradation. To evaluate this, we generate videos of increasing length—2, 4, and 6 clips—under an auto-regressive setting. Each pair of clips moves the camera from left to right and then back to the original viewpoint. We compare our method with Wan2.2 (5B) [87], where the last frame of each generated clip is used as the starting frame for the next. Table 6 summarizes the results, and corresponding visualizations are provided in the appendix.

Scene Point Cloud Density. We maintain a global scene point cloud as spatial memory. Since the raw point cloud is typically dense, we investigate how its density affects generation quality. Let d denote the side length of each cube used for voxelization. For every cube, we aggregate all points within it to obtain a downsampled point cloud. As shown in Table 7, increasing d substantially reduces memory storage but leads to visual quality degradation due to the loss of fine-grained spatial guidance.

5. Conclusion

We introduce *Spatia*, a spatial memory-aware video generation framework that enables consistent, long-horizon synthesis. By maintaining an explicit 3D scene point cloud as persistent memory and iteratively updating it during generation, *Spatia* captures long-term geometric structure that conventional video models cannot preserve. This memory mechanism ensures spatial consistency across revisited locations, supports coherent dynamic content, and enables explicit camera control through 3D-aware conditioning. Extensive experiments demonstrate that *Spatia* significantly enhances long-horizon consistency while maintaining high visual quality in the generated videos.

Spatia: Video Generation with Updatable Spatial Memory

Supplementary Material

6. More Implementation Details

Reference Frame Retrieval. In Section 3.1.2 of the main paper, we describe how to select up to K spatially relevant frames (with $K = 7$ by default) from the candidate-frame set. The complete procedure is provided in Algorithm 1.

Augmentation of Preceding-Frame Latents. Spatia conditions on preceding frames to generate future frames, enabling long-horizon video generation. However, while training uses ground-truth preceding frames as conditions, inference relies on model-generated frames, creating a distribution gap between training and inference. To alleviate this mismatch, we introduce a simple augmentation strategy for preceding-frame latents during training. Specifically, we sample a timestep $t_{\text{aug}} \in [0, 50]$ from a low-noise interval using the same noise scheduler as in Flow Matching training, and add the corresponding noise to the clean preceding-frame latents. The resulting augmented latents are then used as the conditioning inputs in place of the clean latents.

Match Accuracy. In Tables 3 and 5 of the main paper, we include *Match Accuracy* as an additional metric to assess the effectiveness of the memory mechanism in closed-loop video generation, where the last frame is expected to reproduce spatially similar content to the initial frame. *Match Accuracy* quantifies the structural and spatial correspondence between two frames. In our implementation, we use RoMa [23], a robust dense feature-matching algorithm, to estimate correspondences between the first frame I_{first} and the last frame I_{last} . After obtaining the correspondence map, we discard low-confidence matches and count the remaining high-confidence correspondences as the number of valid matches. To ensure comparability across scenes, the final match accuracy is normalized by the number of high-confidence self-matches obtained by matching I_{first} with itself.

Dynamic-Static Disentanglement in the Inference Stage. Our model supports generating videos that contain dynamic entities while maintaining a spatial memory representing the static scene. During inference, to strictly enforce dynamic–static disentanglement, we first apply SAM2 [74] to track and segment dynamic entities in the initial conditioning image or previously generated video clips, and record their segmentation masks. These masks are then used to exclude dynamic regions when updating the spatial memory (i.e., the scene point cloud) with MapAnything [42].

7. Visualization

Qualitative Study on Spatial Memory in Long-Horizon Generation. In Table 4 of the main paper, we quanti-

Algorithm 1 Reference Frame Retrieval

Input: Target frames $\{T\}^N$, candidate frames $\{C\}^O$, view-specific scene point clouds $\{S_T\}^N$ and $\{S_C\}^O$, threshold ϵ , maximum number of reference frames K

Output: Retrieved reference-frame set $\{R\}$

- 1: Initialize $\{R\} \leftarrow \emptyset$
- 2: **for** each target frame $T_i \in \{T\}^N$ **do**
- 3: **if** $i \bmod K \neq 0$ **then**
- 4: **break** ▷ Operate every K frames.
- 5: **end if**
- 6: Initialize $s \leftarrow 0$ ▷ Maximal spatial overlap score.
- 7: Initialize $\hat{R} \leftarrow \emptyset$ ▷ Empty reference frame.
- 8: Identify the scene map $S_{T_i} \in \{S_T\}^N$
- 9: **for** each candidate frame $C_j \in \{C\}^O$ **do**
- 10: Identify the scene map $S_{C_j} \in \{S_C\}^O$
- 11: $s(T_i, C_j) \leftarrow \text{SPATIALOVERLAP}(S_{T_i}, S_{C_j})$
- 12: **if** $s(T_i, C_j) > s$ **then**
- 13: $s \leftarrow s(T_i, C_j)$
- 14: $\hat{R} \leftarrow C_j$
- 15: **end if**
- 16: **end for**
- 17: **if** $s > \epsilon$ **then**
- 18: $\{R\} \leftarrow \{R\} \cup \hat{R}$
- 19: **end if**
- 20: **end for**
- 21: **return** $\{R\}$
- 22: **function** SPATIALOVERLAP(x, y)
- 23: $y' \leftarrow \text{Register}(y, x)$ ▷ Register y to x space.
- 24: $s \leftarrow 3\text{DIoU}(x, y')$
- 25: **return** s
- 26: **end function**

tatively study two key factors for enabling spatial memory and achieving spatially consistent long-horizon generation: (1) the use of reference frames and (2) the use of scene videos. Figure 5 presents a qualitative comparison among three variants: (1) our default model incorporating both components, (2) a model that uses only scene videos without reference frames, and (3) a model that uses reference frames but excludes scene videos. As shown, our full model substantially outperforms both ablated variants, successfully preserving global scene consistency and structural integrity over long temporal sequences, while the baselines exhibit pronounced geometric drift.

Closed-Loop Generation Figure 6 shows visualizations of closed-loop video generation. In these examples, the camera follows a trajectory that returns to the initial viewpoint at the end of the sequence. This setup enables direct evalua-

tion of both visual and geometric consistency by examining whether the final frame spatially aligns with the first frame, thereby validating the effectiveness of our spatial memory in preserving global scene structure.

Generation of Dynamic Entities while Maintaining Static Scenes. Our model supports dynamic–static disentanglement by representing only the static scene in the spatial memory. This is accomplished by removing dynamic entities from the estimated scene point cloud, which is used as the spatial memory, while the original videos containing dynamic entities are used as training targets. Figure 7 illustrates several examples showing the static-only spatial memory alongside the corresponding generated videos, where dynamic entities perform actions within the same scenes.

3D-Aware Interactive Editing Maintaining a scene point cloud as spatial memory and conditioning on it during video generation also enables 3D-aware interactive editing. As shown in Figure 8, manipulating the estimated scene point cloud—such as removing objects, adding new ones, or modifying object colors—leads to corresponding and accurate changes in the generated videos.



Figure 5. Qualitative comparison of three variants for long-horizon video generation: (1) our default model Spatia, (2) a variant using only scene videos without reference frames, and (3) a variant using reference frames but no scene videos. The spatial memories shown in the figure are generated by Spatia.

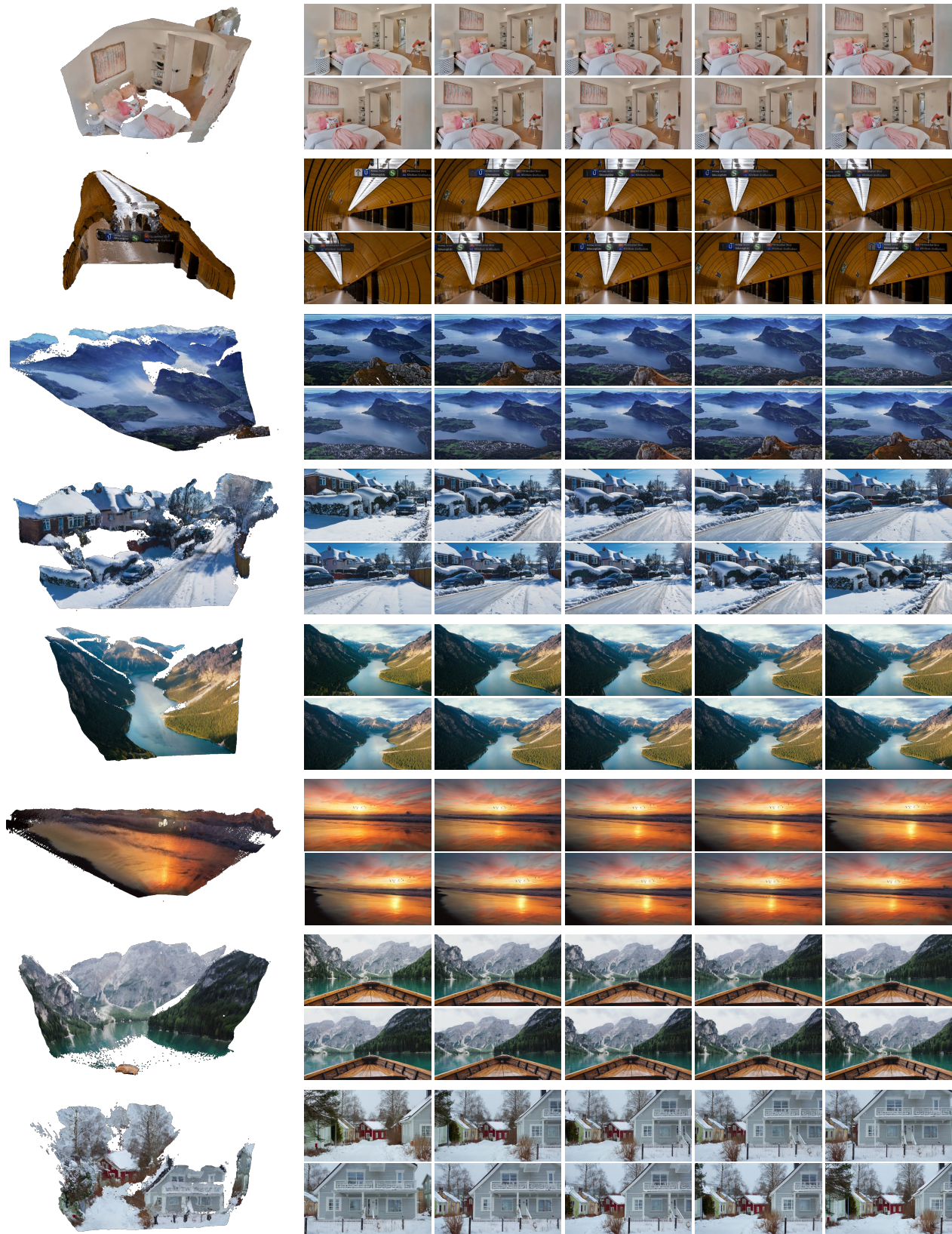


Figure 6. Visualization of closed-loop video generation. The camera follows a trajectory that returns to its initial viewpoint, enabling direct comparison between the first and final frames to evaluate the effectiveness of the spatial memory mechanism.



Figure 7. Visualizations of dynamic–static disentanglement. Our model maintains a spatial memory containing only the static scene point cloud while generating videos that include dynamic entities acting within the same scenes.



Figure 8. Demonstration of 3D-aware interactive editing. By directly modifying the spatial memory (i.e., the scene point cloud), users can achieve geometrically precise edits in the generated videos, such as removing an object (2nd row), adding a new object (3rd row), or altering object attributes (4th row).

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2, 3
- [2] 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, 3
- [3] Anthropic. Claude 3 model card. Technical report, Anthropic PBC, 2024. 3
- [4] Anthropic. Claude 4 model card (claude opus 4 & sonnet 4). Technical report, Anthropic PBC, 2025. 3
- [5] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 2
- [6] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuo Zhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. Recammaster: Camera-controlled generative rendering from a single video. *arXiv preprint arXiv:2503.11647*, 2025. 2, 3
- [7] 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. 3
- [8] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Krzysztof Choromanski, Tianli Ding, Danny Driess, Kumar Avinava Dubey, Chelsea Finn, Peter R. Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil J. Joshi, Ryan C. Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Sergey Levine, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Pierre Sermanet, Jaspier Singh, Anikait Singh, Radu Soricut, Huang Tran, Vincent Vanhoucke, Quan Ho Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Ted Xiao, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *ArXiv*, abs/2307.15818, 2023. 2
- [9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3
- [10] Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024. 2
- [11] Johann Cabon, Lucas Stofl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jerome Revaud, and Vincent Leroy. Must3r: Multi-view network for stereo 3d reconstruction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1050–1060, 2025. 3
- [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. 2
- [13] Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024. 3
- [14] Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Juncheng Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengchen Ma, et al. Skyreels-v2: Infinite-length film generative model. *arXiv preprint arXiv:2504.13074*, 2025. 3
- [15] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023. 3
- [16] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7310–7320, 2024. 3, 7
- [17] 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 flexible-view synthesis. *arXiv preprint arXiv:2503.13265*, 2025. 2, 3, 7, 8
- [18] Etched Decart, Quinn McIntyre, Spruce Campbell, Xinlei Chen, and Robert Wachen. Oasis: A universe in a transformer. URL: <https://oasis-model.github.io>, 2024. 2
- [19] Google DeepMind. Gemini 2.0 flash: A multimodal model with 1 million token context window. <https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-0-flash>, 2025. 3
- [20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019. 3
- [21] Yiran Ding, Li Lina Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. Longrope: Extending llm context window beyond 2 million tokens. *arXiv preprint arXiv:2402.13753*, 2024. 3
- [22] Haoyi Duan, Hong-Xing Yu, Sirui Chen, Li Fei-Fei, and Jiajun Wu. Worldscore: A unified evaluation benchmark for world generation. *arXiv preprint arXiv:2504.00983*, 2025. 7
- [23] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. RoMa: Robust Dense

- Feature Matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 9
- [24] Paul Engstler, Andrea Vedaldi, Iro Laina, and Christian Rupprecht. Invisible stitch: Generating smooth 3d scenes with depth inpainting. In *Arxiv*, 2024. 2, 7
- [25] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 3
- [26] Weichen Fan, Chenyang Si, Junhao Song, Zhenyu Yang, Yanan He, Long Zhuo, Ziqi Huang, Ziyue Dong, Jingwen He, Dongwei Pan, et al. Vchitect-2.0: Parallel transformer for scaling up video diffusion models. *arXiv preprint arXiv:2501.08453*, 2025. 7
- [27] 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. 2
- [28] Wanquan Feng, Jiawei Liu, Pengqi Tu, Tianhao Qi, Mingzhen Sun, Tianxiang Ma, Songtao Zhao, Siyu Zhou, and Qian He. I2vcontrol-camera: Precise video camera control with adjustable motion strength. *arXiv preprint arXiv:2411.06525*, 2024. 2, 3
- [29] 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. 3
- [30] Yu Gao, Haoyuan Guo, Tuyen Hoang, Weilin Huang, Lu Jiang, Fangyuan Kong, Huixia Li, Jiashi Li, Liang Li, Xiaojie Li, et al. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113*, 2025. 2
- [31] Google. Veo. <https://deepmind.google/models/veo/>, 2024. 2, 3
- [32] Yuchao Gu, Weijia Mao, and Mike Zheng Shou. Long-context autoregressive video modeling with next-frame prediction. *arXiv preprint arXiv:2503.19325*, 2025. 3
- [33] 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. 3
- [34] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 2, 3
- [35] 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. 2, 7
- [36] Asher Hancock, Allen Z. Ren, and Anirudha Majumdar. Run-time observation interventions make vision-language-action models more visually robust. *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9499–9506, 2024. 2
- [37] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024. 2, 3
- [38] 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. 2, 3
- [39] Roberto Henschel, Levon Khachatryan, Hayk Poghosyan, Daniil Hayrapetyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streaming2v: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*, 2024. 3
- [40] Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson WH Lau, Wangmeng Zuo, et al. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *arXiv preprint arXiv:2506.04225*, 2025. 2, 3, 7, 8
- [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. *arXiv preprint arXiv:2410.05954*, 2024. 3
- [42] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kotschieder. MapAnything: Universal feed-forward metric 3D reconstruction, 2025. *arXiv preprint arXiv:2509.13414*. 3, 5, 6, 9
- [43] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2
- [44] Jihwan Kim, Junoh Kang, Jinyoung Choi, and Bohyung Han. Fifo-diffusion: Generating infinite videos from text without training. *arXiv preprint arXiv:2405.11473*, 2024. 3
- [45] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Paul Foster, Grace Lam, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. *ArXiv*, abs/2406.09246, 2024. 2
- [46] 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. 2, 3

- [47] Kuaishou. Kling. <https://klingai.com>, 2024. 2, 3
- [48] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626, 2023. 3
- [49] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r, 2024. 3
- [50] Chengmeng Li, Junjie Wen, Yan Peng, Yaxin Peng, Feifei Feng, and Yichen Zhu. Pointvla: Injecting the 3d world into vision-language-action models. *ArXiv*, abs/2503.07511, 2025. 2
- [51] Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory. *arXiv preprint arXiv:2506.18903*, 2025. 2, 3, 7
- [52] Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chi-Hou Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. *ArXiv*, abs/2311.01378, 2023. 2
- [53] Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. Snapkv: Llm knows what you are looking for before generation. *Advances in Neural Information Processing Systems*, 37:22947–22970, 2024. 3
- [54] Tianming Liang, Kun-Yu Lin, Chaolei Tan, Jianguo Zhang, Wei-Shi Zheng, and Jian-Fang Hu. Referdino: Referring video object segmentation with visual grounding foundations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. 5
- [55] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024. 3
- [56] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 5
- [57] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024. 2
- [58] Hao Liu, Matei Zaharia, and Pieter Abbeel. Ring attention with blockwise transformers for near-infinite context. *arXiv preprint arXiv:2310.01889*, 2023. 3
- [59] Jinlai Liu, Jian Han, Bin Yan, Hui Wu, Fengda Zhu, Xing Wang, Yi Jiang, Bingyue Peng, and Zehuan Yuan. Infinitytar: Unified spacetime autoregressive modeling for visual generation. *arXiv preprint arXiv:2511.04675*, 2025. 3
- [60] Zhening Liu, Yingdong Hu, Xinjie Zhang, Rui Song, Jiawei Shao, Zehong Lin, and Jun Zhang. Dynamics-aware gaussian splatting streaming towards fast on-the-fly 4d reconstruction. *arXiv preprint arXiv:2411.14847*, 2024. 3
- [61] Yawen Luo, Jianhong Bai, Xiaoyu Shi, Menghan Xia, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, and Tianfan Xue. Camclonemaster: Enabling reference-based camera control for video generation. *arXiv preprint arXiv:2506.03140*, 2025. 3
- [62] Baorui Ma, Huachen Gao, Haoge Deng, Zhengxiong Luo, Tiejun Huang, Lulu Tang, and Xinlong Wang. You see it, you got it: Learning 3d creation on pose-free videos at scale. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2016–2029, 2025. 3
- [63] Guoqing Ma, Haoyang Huang, Kun Yan, Liangyu Chen, Nan Duan, Shengming Yin, Changyi Wan, Ranchen Ming, Xiaoniu Song, Xing Chen, et al. Step-video-t2v technical report: The practice, challenges, and future of video foundation model. *arXiv preprint arXiv:2502.10248*, 2025. 3
- [64] Meta-AI. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation. Meta AI Blog, 2025. 2, 3
- [65] MiniMax. Hailuo. <https://hailuoai.video/>, 2024. 2, 3
- [66] OpenAI. Sora. <https://openai.com/sora/>, 2024. 2, 3
- [67] OpenAI. Introducing gpt-4.1 in the api. <https://openai.com/index/gpt-4-1/>, 2025. 2, 3
- [68] 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. 2
- [69] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 3
- [70] Xiangyu Peng, Zangwei Zheng, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, Wenjun Li, et al. Open-sora 2.0: Training a commercial-level video generation model in \$200 k. *arXiv preprint arXiv:2503.09642*, 2025. 3
- [71] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018. 3
- [72] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [73] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020. 3
- [74] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-

- Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 9
- [75] 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. 3
- [76] Runway. Introducing gen-3 alpha: A new frontier for video generation, 2024. Accessed: 2025-02-24. 2, 3
- [77] Sand-AI. Magi-1: Autoregressive video generation at scale, 2025. 3
- [78] Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. Videovla: Video generators can be generalizable robot manipulators. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 2
- [79] Vincent Sitzmann, Semon Rezchikov, William T. Freeman, Joshua B. Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. In *Proc. NeurIPS*, 2021. 3
- [80] Kiwhan Song, Boyuan Chen, Max Simchowitz, Yilun Du, Russ Tedrake, and Vincent Sitzmann. History-guided video diffusion. *arXiv preprint arXiv:2502.06764*, 2025. 3
- [81] Wenqiang Sun, Fangyun Wei, Jinjing Zhao, Xi Chen, Zilong Chen, Hongyang Zhang, Jun Zhang, and Yan Lu. From virtual games to real-world play. *arXiv preprint arXiv:2506.18901*, 2025. 2
- [82] Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024. 3
- [83] Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025. 2
- [84] Meituan LongCat Team, Xunliang Cai, Qilong Huang, Zhuoliang Kang, Hongyu Li, Shijun Liang, Liya Ma, Siyu Ren, Xiaoming Wei, Rixu Xie, et al. Longcat-video technical report. *arXiv preprint arXiv:2510.22200*, 2025. 2, 3
- [85] Octo Model Team, Dibya Ghosh, Homer Rich Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Pannag R. Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. *ArXiv*, abs/2405.12213, 2024. 2
- [86] Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. *arXiv preprint arXiv:2408.14837*, 2024. 2, 3
- [87] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wenten Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 2, 3, 5, 6, 7, 8
- [88] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21686–21697, 2024. 3
- [89] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 3
- [90] Jiahao Wang, Yufeng Yuan, Rujie Zheng, Youtian Lin, Jian Gao, Lin-Zhuo Chen, Yajie Bao, Yi Zhang, Chang Zeng, Yanxi Zhou, et al. Spatialvid: A large-scale video dataset with spatial annotations. *arXiv preprint arXiv:2509.09676*, 2025. 6
- [91] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. *arXiv preprint arXiv:2501.12387*, 2025. 3
- [92] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 3
- [93] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. Pi3: Scalable permutation-equivariant visual geometry learning. *arXiv e-prints*, pages arXiv–2507, 2025. 3
- [94] Junjie Wen, Minjie Zhu, Yichen Zhu, Zhibin Tang, Jinming Li, Zhongyi Zhou, Chengmeng Li, Xiaoyu Liu, Yaxin Peng, Chaomin Shen, and Feifei Feng. Diffusion-vla: Generalizable and interpretable robot foundation model via self-generated reasoning. 2024. 2
- [95] Chenfei Wu, Jian Liang, Xiaowei Hu, Zhe Gan, Jianfeng Wang, Lijuan Wang, Zicheng Liu, Yuejian Fang, and Nan Duan. Nuwa-infinity: Autoregressive over autoregressive generation for infinite visual synthesis. *arXiv preprint arXiv:2207.09814*, 2022. 3
- [96] Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. Worldmem: Long-term consistent world simulation with memory. *arXiv preprint arXiv:2504.12369*, 2025. 2
- [97] Desai Xie, Zhan Xu, Yicong Hong, Hao Tan, Difan Liu, Feng Liu, Arie Kaufman, and Yang Zhou. Progressive

- autoregressive video diffusion models. *arXiv preprint arXiv:2410.08151*, 2024. 3
- [98] Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, et al. Effective long-context scaling of foundation models. *arXiv preprint arXiv:2309.16039*, 2023. 3
- [99] Jiaqi Xu, Xinyi Zou, Kunzhe Huang, Yunkuo Chen, Bo Liu, MengLi Cheng, Xing Shi, and Jun Huang. Easyanimate: A high-performance long video generation method based on transformer architecture. *arXiv preprint arXiv:2405.18991*, 2024. 7
- [100] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 2
- [101] Biao Yang, Bin Wen, Boyang Ding, Changyi Liu, Chenglong Chu, Chengru Song, Chongling Rao, Chuan Yi, Da Li, Dunju Zang, et al. Kwai keye-v1 1.5 technical report. *arXiv preprint arXiv:2509.01563*, 2025. 5
- [102] Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3
- [103] Mingyu Yang, Junyou Li, Zhongbin Fang, Sheng Chen, Yangbin Yu, Qiang Fu, Wei Yang, and Deheng Ye. Playable game generation. *arXiv preprint arXiv:2412.00887*, 2024. 2
- [104] Shiyuan Yang, Liang Hou, Haibin Huang, Chongyang Ma, Pengfei Wan, Di Zhang, Xiaodong Chen, and Jing Liao. Direct-a-video: Customized video generation with user-directed camera movement and object motion. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–12, 2024. 2, 3
- [105] Xiaoda Yang, Jiayang Xu, Kaixuan Luan, Xinyu Zhan, Hongshun Qiu, Shijun Shi, Hao Li, Shuai Yang, Li Zhang, Checheng Yu, et al. Omnicam: Unified multimodal video generation via camera control. *arXiv preprint arXiv:2504.02312*, 2025. 3
- [106] 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. 2, 3, 7
- [107] 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. *arXiv preprint arXiv:2412.07772*, 2, 2024. 3
- [108] Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T. Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. *arXiv:2406.09394*, 2024. 2, 7
- [109] Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snively, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6658–6667, 2024. 2, 7
- [110] Jiwen Yu, Jianhong Bai, Yiran Qin, Quande Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Context as memory: Scene-consistent interactive long video generation with memory retrieval. *arXiv preprint arXiv:2506.03141*, 2025. 3
- [111] 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. 2
- [112] 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. 2, 3
- [113] 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. 2, 3, 7, 8
- [114] Yan Zeng, Guoqiang Wei, Jiani Zheng, Jiabin Zou, Yang Wei, Yuchen Zhang, and Hang Li. Make pixels dance: High-dynamic video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8850–8860, 2024. 3
- [115] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 5, 6
- [116] Xinjie Zhang, Zhening Liu, Yifan Zhang, Xingtong Ge, Dailan He, Tongda Xu, Yan Wang, Zehong Lin, Shuicheng Yan, and Jun Zhang. Mega: Memory-efficient 4d gaussian splatting for dynamic scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 27828–27838, 2025. 2
- [117] Yifan Zhang, Chunli Peng, Boyang Wang, Puyi Wang, Qingcheng Zhu, Zedong Gao, Eric Li, Yang Liu, and Yahui Zhou. Matrix-game: Interactive world foundation model. *arXiv*, 2025. 2
- [118] Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, et al. H2o: Heavy-hitter oracle for efficient generative inference of large language models. *Advances in Neural Information Processing Systems*, 36:34661–34710, 2023. 3
- [119] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-via: A 3d vision-language-action generative world model. *ArXiv*, abs/2403.09631, 2024. 2
- [120] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024. 3
- [121] Jensen Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishtha, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. *arXiv preprint arXiv:2503.14489*, 2025. 3, 7

- [122] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018. [6](#), [7](#)
- [123] Yuan Zhou, Qiuyue Wang, Yuxuan Cai, and Huan Yang. Allegro: Open the black box of commercial-level video generation model. *arXiv preprint arXiv:2410.15458*, 2024. [7](#)