From Zero to Hero: Training-Free Custom Concept Spawning in Interactive World Models

Kiymet Akdemir **Pinar Yanardag**
Virginia Tech
spawn-world.github.io



Figure 1: **Controllable concept spawning with SPAWN.** Given an input frame and rightward camera motion, the baseline WorldPlay populates the unseen region with its priors, ignoring the user's request (top row). SPAWN instead renders the requested concept into the world via either a text prompt (middle row) or a concept image (bottom row), integrated with consistent lighting and perspective.

Abstract

Autoregressive world models have emerged as a powerful paradigm for interactive video generation, allowing users to navigate dynamically generated environments through actions. These models are typically conditioned on a text prompt and/or a single reference frame, from which the entire world is generated. Yet the moment the user navigates beyond what is visible in that frame, the unseen regions are populated by the base model's priors, with no mechanism for the user to specify what should appear and where. This is a fundamental limitation for applications such as gaming, interactive storytelling, and simulation, where controllable scene composition is essential. We refer to this missing capability as concept spawning: introducing a user-specified visual concept into a world model, analogous to spawning in a game engine. We introduce SPAWN (**S**wapping **P**inned **A**nchor with **W**indowed i**N**jection), a training-free method for concept spawning. SPAWN exploits a structural property of image-to-video backbones: the first slot of the context memory is pinned to the reference frame and acts as a foundational anchor for every generated chunk. By swapping this anchor with an external concept latent over a short injection window, we cause the concept to propagate naturally through the rollout via the model's own memory. SPAWN supports concepts from fine-grained entities such as characters and props to large elements such as buildings and landscapes, and accepts either an image or a text description as input. Experiments show that SPAWN integrates concepts with consistent lighting, scale,

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and perspective while preserving identity and temporal coherence, demonstrating that controllable concept spawning is achievable without any training in image-conditioned autoregressive video world models.

1 Introduction

Autoregressive world models have rapidly gained popularity for their potential to power immersive applications such as gaming, interactive storytelling, simulation, and embodied AI, where users can navigate dynamically generated environments through actions in real time [3, 5, 31, 36]. These models are typically conditioned on a single reference frame or a text prompt, from which the entire surrounding world is generated chunk by chunk as the user navigates. The visible portion of the input frame anchors the scene in front of the camera. However, once the user turns to look elsewhere, the unseen regions must be hallucinated based on the base model's priors. As a result, the model fills in the world with whatever it deems a plausible continuation of the scene, without any mechanism for the user to control what appears or where. We refer to this missing capability as concept spawning: the ability to introduce a user-specified visual concept into an already-running world at a chosen moment, analogous to spawning in a game engine.

Table 1: Capability comparison across interactive world models. SPAWN is the only method that supports image-conditioned concept spawning without training.

Method	Concept Spawning	Out-of-View	Image-Cond.	Training-Free	Public
WorldPlay	✗	✗	✗	✗	✓
GameGen-X	✓	✗	✗	✗	✗
GameCraft-2	✓	✗	✗	✗	✗
SPAWN	✓	✓	✓	✓	✓

This is a fundamental limitation since the ability to compose and control scenes is central to level design, interactive storytelling, embodied AI simulation, and any application where generated worlds must be deliberately controlled instead of passively sampled. Existing work has made progress toward controllability through mechanisms such as text-conditioned scene generation [5] and instruction-following world models [32]. However, these approaches provide only limited control (see Tab. 1): they can instantiate concepts via text prompts within the current field of view, but fail to maintain consistent, spatially grounded control as the user navigates the environment. Moreover, these methods do not support user-specified visual concepts, instead relying on abstract textual descriptions, and often require costly training on curated interactive data.

We introduce SPAWN (**S**wapping **P**inned **A**nchor with **W**indowed i**N**jection), a training-free method for concept spawning in a world model. Our approach is motivated by a key observation: autoregressive world models built on image-to-video backbones have the first slot of the context memory pinned to the reference frame across the entire rollout, where it acts as a foundational anchor that every generated chunk attends to. This slot effectively serves as the model's notion of ground truth for the scene, analogous to attention sinks observed in large language models, where specific tokens disproportionately anchor attention and shape downstream generation [26, 35]. SPAWN exploits this property by swapping the pinned anchor with an external concept latent over a short windowed injection, then letting the original anchor return. During the injection window, the concept propagates into the temporal and spatial memory caches via the model's normal generation process; once the window closes and the original anchor is restored, the concept persists in those downstream caches and continues to appear naturally as the camera navigates the world. SPAWN does not require architectural modifications, fine-tuning, or curated training data.

A user can drive SPAWN with either a concept image, when they have a specific visual reference in mind (e.g., this specific horse), or a text description (e.g., a lake on the right). Figure 1 illustrates the resulting capability: given the same input frame and rightward camera motion, the baseline world model fills the unseen region with priors-driven content, while SPAWN renders the requested lake or specific horse directly into the scene, integrated with consistent lighting, scale, and perspective. We evaluate SPAWN on WorldPlay across 200 generated rollouts spanning diverse scenes, concept categories, and visual styles. SPAWN supports concepts from fine-grained entities such as characters and props to large-scale scene elements such as buildings and landscapes, and its effects persist over long rollouts, across multiple sequential injections and camera revisits. Our experiments show that SPAWN preserves visual identity, temporal coherence, and scene integration without any training or architectural changes to the base model, suggesting that controllable concept spawning is an

inference-time capability of autoregressive world models that retain a persistent reference anchor. Our main contributions are as follows:

- We introduce SPAWN, a training-free mechanism that exploits the anchor slots of the context memory to spawn concepts into a world from a user-provided concept image or a text prompt.
- We demonstrate on WorldPlay that SPAWN spawns concepts at arbitrary granularity with consistent lighting, scale, and perspective, preserves identity and temporal coherence across long rollouts, and supports multiple sequential injections, all without any architectural modification or additional training.
- We release our code and a benchmark dataset of 200 rollouts spanning diverse scenes and concept categories, enabling systematic evaluation of concept spawning in generated worlds.

2 Related work

Video generation. Diffusion models [10, 25, 30] are the dominant paradigm for visual generative modeling, with early video work adopting latent diffusion frameworks [28] for efficient, temporally coherent synthesis in compressed latent spaces [2, 7, 11]. The field has since transitioned from U-Net to scalable Diffusion Transformer (DiT) backbones [25], with large-scale models such as CogVideoX [39], Wan [33], and HunyuanVideo [16] generating high-quality but non-interactive sequences. Autoregressive video generation [9, 15, 37] extends video length by streaming frame-by-frame; Diffusion Forcing [6] introduces per-frame noise levels in place of the uniform-noise full-sequence paradigm, and Self-Forcing [12] closes the training-inference gap by conditioning on the model's own previously generated frames to reduce long-horizon error accumulation. Text-guided video editing [4, 21, 27] similarly steers diffusion priors toward targeted scene modifications, though it operates on fixed sequences rather than interactive environments.

Interactive world models. World video models extend video generation to interactive settings, generating temporally coherent sequences conditioned on user actions. Early action-conditioned generators established the paradigm in narrow game domains with limited action vocabularies and short horizons [23, 42]. Subsequent work expanded the interaction and environmental scope through multi-modal control [5] and richer open-world environments [8, 43]. A parallel line addresses the long-horizon consistency that sliding-window approaches fail to preserve, through external memory banks [36], explicit 3D surface reconstruction [19], retrieval-based context extension [41], and selective KV-cache management for prompt adaptability [38]. WorldPlay [31] retrieves geometrically relevant frames via camera pose overlap and reassigns their temporal embeddings to maintain consistency across extended rollouts. Most recently, the Hunyuan-GameCraft series [17, 32] adds language-guided world steering to action control, though it requires dedicated training and accepts only text instructions. Despite this progress, no existing approach supports image-based concept customization; SPAWN addresses this by repurposing the model's anchor slots at inference time.

3 Method

Our method spawns concepts by replacing the anchor slots of the context memory with a concept latent over a short injection window. We first describe the context-memory structure, then detail the injection mechanism. See Fig. 2 for a method overview.

3.1 Preliminaries

We build on WorldPlay [31], an autoregressive world model with a HunyuanVideo [16] backbone that performs next-chunk prediction $N_\theta(x_t \mid O_{t-1}, A_{t-1}, a_t, r)$, where $x_t \in \mathbb{R}^{C\times 4\times H\times W}$ is the next chunk of four latents (16 video frames after VAE decoding), $O_{t-1} = \{x_0, \ldots, x_{t-1}\}$ is the past, $A_{t-1} = \{a_0, \ldots, a_{t-1}\}$ and a_t are past and current actions, and r is the reference image. r is encoded by a causal 3D VAE $\mathcal{E}$ into a single latent, the first latent of the rollout, which occupies the first slot of the context memory and anchors the rollout.

Conditioning on all of O_{t-1} is intractable, so the model conditions on a reconstituted context memory: a sequence of latents drawn from past chunks that combines a fixed *anchor memory* C^{anc}, a *temporal*

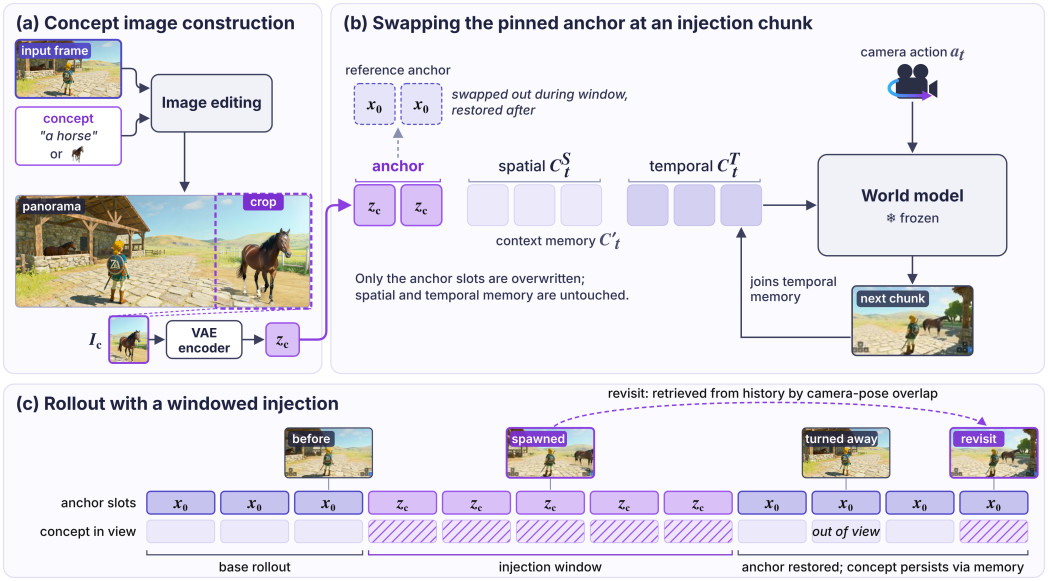

Figure 2: **Overview of SPAWN. (a)** Given an input frame and a concept described by text or an image, we use an image editing model to place the concept in a panorama of the scene. The crop at the spawn direction is encoded as the concept latent z_c. **(b)** We temporarily replace the anchor slots in the context memory with z_c, which leads the frozen world model to render the concept into the scene without any training. **(c)** After a short injection window, the original anchors return, yet the concept persists in the model's memory and reappears when the camera revisits it.

memory C_t^T, and a *spatial memory* C_t^S,

$$C_t = C^{\text{anc}} \cup C_t^T \cup C_t^S, \qquad C_t^S \subseteq O_{t-1} \setminus (C^{\text{anc}} \cup C_t^T). \tag{1}$$

The anchor memory consists of the first latents of the rollout, beginning with the reference latent, and is held fixed throughout generation (four latents in WorldPlay). The temporal memory holds the latents of the most recent chunks for short-term motion smoothness, while the spatial memory samples non-adjacent latents by camera-pose overlap and proximity for long-range geometric consistency. Each slot of C_t holds one latent, and latents are ordered by their absolute temporal index, so the anchor slots come first. Because the spatial-retrieval set is reselected at every chunk as the camera moves, the visual KV cache is recomputed from C_t at the start of every chunk's denoising,

$$\text{KV}_t \leftarrow N_\theta^{\text{KV}}(C_t), \tag{2}$$

where N_θ^{KV} denotes the encoding pass that produces keys and values from the context latents.

3.2 Swapping Pinned Anchor with Windowed Injection

The anchor role of the first slot. In autoregressive video diffusion models with image conditioning, the first slot of the context memory is structurally distinguished. It holds the reference image as a fixed, zero-noise latent throughout the rollout [16, 33, 39], serving as the primary anchor that every chunk attends to, from the very first prediction to the last. Its content conditions every chunk the model generates and exerts a strong influence throughout the rollout, consistent with the attention-sink phenomenon originally observed in LLMs [26, 35], where a small number of early positions concentrate a large share of attention mass. Recent autoregressive video generation methods [20, 38, 40] leverage this persistent influence as a *stabilizer*: a fixed early region of the KV cache is held constant throughout the rollout to prevent long-range collapse and preserve global consistency. We instead repurpose the first slot as a *control surface*. By overwriting the latent in the first anchor slot with an external concept before the chunk's KV recompute, that concept inherits the anchor role and propagates through the model's downstream generation. Replacing any other anchor slot alone does not produce this effect (see Appendix A.4).

Anchor replacement. Let I_c be a concept image and $z_c = \mathcal{E}(I_c) \in \mathbb{R}^{C\times 1\times H\times W}$ its single-latent VAE encoding. At each chunk $t \in \mathcal{T}$ at which the concept is to be injected, we replace the first two

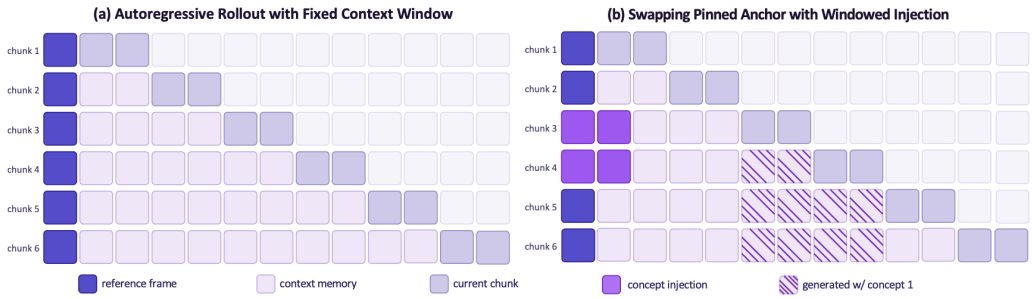

Figure 3: **(a) Standard rollout:** context memory evolves without intervention; the first slot holds the reference frame and anchors the rollout. **(b) SPAWN:** anchor slots are replaced with a concept latent during injection chunks, and diagonal hatching marks slots showing how the concept propagates through subsequent generation. The figure shows simplified memory size and concept counts.

anchor slots of C_t in place with z_c:

$$C'_t = \big(z_c,\ z_c,\ C_t[2],\ \ldots,\ C_t[|C_t| - 1]\big). \tag{3}$$

Replacing the first anchor slot alone is sufficient to spawn the concept into the rollout; we additionally replace the second anchor slot, which we find strengthens identity preservation (see Appendix A.4). All other slots, including the remaining anchor slots, the temporal memory, and the spatial memory, are left unchanged, so scene context, motion continuity, and long-range geometric consistency are preserved.

Recaching. With C'_t in place of C_t, the per-chunk KV recompute of Eq. 2 becomes

$$\mathrm{KV}_t \ \leftarrow\ N_\theta^{\mathrm{KV}}(C'_t). \tag{4}$$

Each DiT block of N_θ computes self-attention with two complementary positional encodings, namely 3D RoPE for video tokens and PRoPE [18] for camera frustums:

$$\mathrm{Attn}_1 = \mathrm{Attn}\big(R^\top \odot Q,\ R^{-1} \odot K,\ V\big), \tag{5}$$

$$\mathrm{Attn}_2 = D_{\mathrm{proj}} \odot \mathrm{Attn}\big((D_{\mathrm{proj}})^\top \odot Q,\ (D_{\mathrm{proj}})^{-1} \odot K,\ (D_{\mathrm{proj}})^{-1} \odot V\big), \tag{6}$$

with block output $\mathrm{Attn}_1 + \mathrm{zero_init}(\mathrm{Attn}_2)$. The substitution changes only the latent content of the replaced slots: each slot retains the temporal index, RoPE rotation R, and camera pose (D_{proj}) of the original anchor it replaces. After the recompute, the keys and values at these positions are therefore derived from z_c but positioned and projected exactly as the original anchors, so the current chunk's queries Q attend to a cache whose anchor region holds the concept at the reference frame's position and viewpoint. All other slots receive their original latents as input. The substitution requires no architectural change, no additional masking, and no fine-tuning, since N_θ already consumes an arbitrary C_t at every chunk under both attention paths.

Concept persistence after injection. Once the rollout passes the last chunk in $\mathcal{T}$, no further substitution takes place: for $t \notin \mathcal{T}$, the anchor slots of C_t hold the original anchor latents again. Yet the concept persists in the rollout, for two reasons. First, the chunks generated during the injection window are themselves rendered with the concept present in the scene, and their latents enter C_t^T for subsequent chunks through the standard temporal-memory pathway, carrying the concept forward as long as they remain in the temporal memory. Second, once those latents fall out of C_t^T, the spatial memory C_t^S retrieves them whenever the camera returns to a viewpoint that overlaps with the chunks generated during injection. The concept therefore continues to appear naturally through the same memory mechanism that maintains long-term geometric consistency in the base model, with no additional bookkeeping. In effect, the injection window seeds the model's memory, and the rollout then propagates the concept on its own (Fig. 3).

Multi-concept spawning. The formulation extends to multiple concepts $\{(I_c^j, \mathcal{T}^j)\}_{j=1}^M$ with disjoint schedules $\mathcal{T}^j \cap \mathcal{T}^l = \emptyset$ for $j \neq l$. At each $t \in \mathcal{T}^j$, the anchor slots of C_t are replaced with $z_c^j = \mathcal{E}(I_c^j)$ as in Eq. 3. Concepts intended to appear together at the same chunk can be combined into a single concept image and injected jointly.

Concept image. The concept image I_c can be specified either by a text description or by an example image of the concept. We first extend the reference frame into a panorama and insert the concept at



Figure 4: SPAWN supports both image-based (a) and text-based (b) control. Each row injects a concept, specified by an image or a text prompt, with consistent lighting and perspective.

the user-chosen location with an image editing model such as Qwen-Image-Edit [34], conditioned on the text description or example image. We then crop the region corresponding to the spawn direction to obtain I_c. This serves two purposes: the user directly controls where the concept appears in the world through its placement in the panorama, and the concept image is generated within the context of the surrounding scene.

4 Experiments

Baselines. Several recent works share aspects of our setting, but none target the exact same task: Hunyuan-GameCraft-2 [32] introduces entities from text prompts but requires dedicated training, while GameGen-X [5] follows text-based instructions at inference. Neither is publicly available at the time of writing, ruling out direct comparison. We therefore evaluate against three available baselines: HunyuanVideo [16] and Wan 2.2 [33], two strong image-to-video (I2V) foundation models testing whether a general video backbone can introduce the requested concept under text conditioning alone, and WorldPlay [31], the autoregressive world model on which we build. All baselines receive a text prompt describing the scene with the concept present; for HunyuanVideo and Wan 2.2, which do not expose a separate action interface, the target camera motion is encoded into the same prompt. We additionally compare against baselines conditioned directly on the edited view in Appendix A.5.

Evaluation set. We evaluate on 200 generated rollouts of 8 seconds each, spanning a range of scenes, concepts, and visual styles. The 8-second horizon is chosen for fair comparison with the I2V baselines above, which are trained for short-clip generation and do not support extended autoregressive rollout. All experiments run on two NVIDIA H200 GPUs. See Appendix A.1 for details.

Metrics. We report VBench [13] metrics (Aesthetic Quality, Background Consistency, Dynamic Degree, Imaging Quality, Motion Smoothness, Subject Consistency) along with the I2V Subject, I2V Background, and Camera Motion metrics of VBench++ [14]. Since VBench does not measure whether the requested concept actually appears, we additionally evaluate concept spawning directly with GroundingDINO [22], prompted with the concept's text description on every generated chunk. We report *r@55/65/75*, the fraction of rollouts in which the concept is detected with confidence above 0.55, 0.65, and 0.75, respectively, and *Conf*, the highest detection confidence in each chunk averaged over chunks and rollouts.

4.1 Qualitative experiments

Qualitative results. Our method injects a broad range of concept categories into a generated world, including living entities such as a cowboy and a bear, objects such as a cauldron and a camping

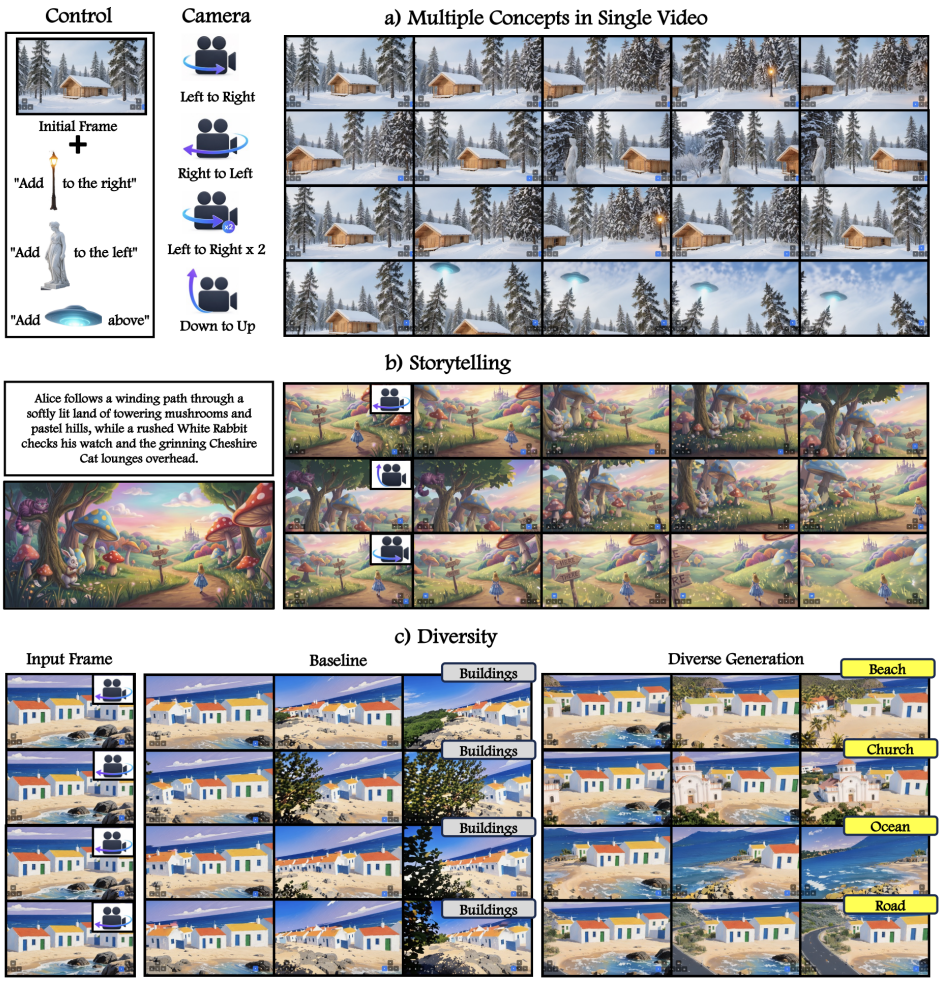

Figure 5: Use cases of SPAWN. **Multiple Concepts:** distinct concepts spawned into a single rollout. **Storytelling:** a story-derived panorama seeds both the reference image and the concepts that populate the world. **Diversity:** spawning populates the world with diverse landscape features.

tent, and physically complex elements such as fire and glass (Fig. 4). Under image-based control (Fig. 4(a)), the method preserves the visual identity of the concept image, retaining fine-grained details of the spawned concept throughout the rollout. See Appendix A.2 for more examples and Appendix A.3 for concepts that conflict with the scene geometry.

Fig. 5 highlights three workflows enabled by our method. First, multiple distinct concepts can be spawned at different points within a single rollout. In Fig. 5(a), three concepts are introduced sequentially into a 30-second rollout of a snowy forest world. Notably, the lamppost is injected only once, yet reappears in the final frames after the camera turns away and then back: the concept is carried forward by the temporal memory and later retrieved by the spatial memory, without any further injection. Second, our method supports a story-driven workflow (Fig. 5(b)), in which a panorama generated from a short story description seeds both the reference image and the concepts that populate it. This turns an illustrated scene into a navigable world, with each concept appearing at its authored location as the user explores. Third, beyond individual entities, our method enables broader environmental modifications. In Fig. 5(c), the base model defaults to generating additional buildings even under varied prompts, whereas our method spawns distinct scene elements such as a beach, a church, an ocean, and a road, producing different navigable worlds.

Qualitative comparison. Figure 6 compares our method against HunyuanVideo, Wan 2.2, and WorldPlay on four scenes spanning small props, large objects, buildings, and signage, each pairing a camera motion with a prompt-specified concept. HunyuanVideo omits the concept in all four scenes.

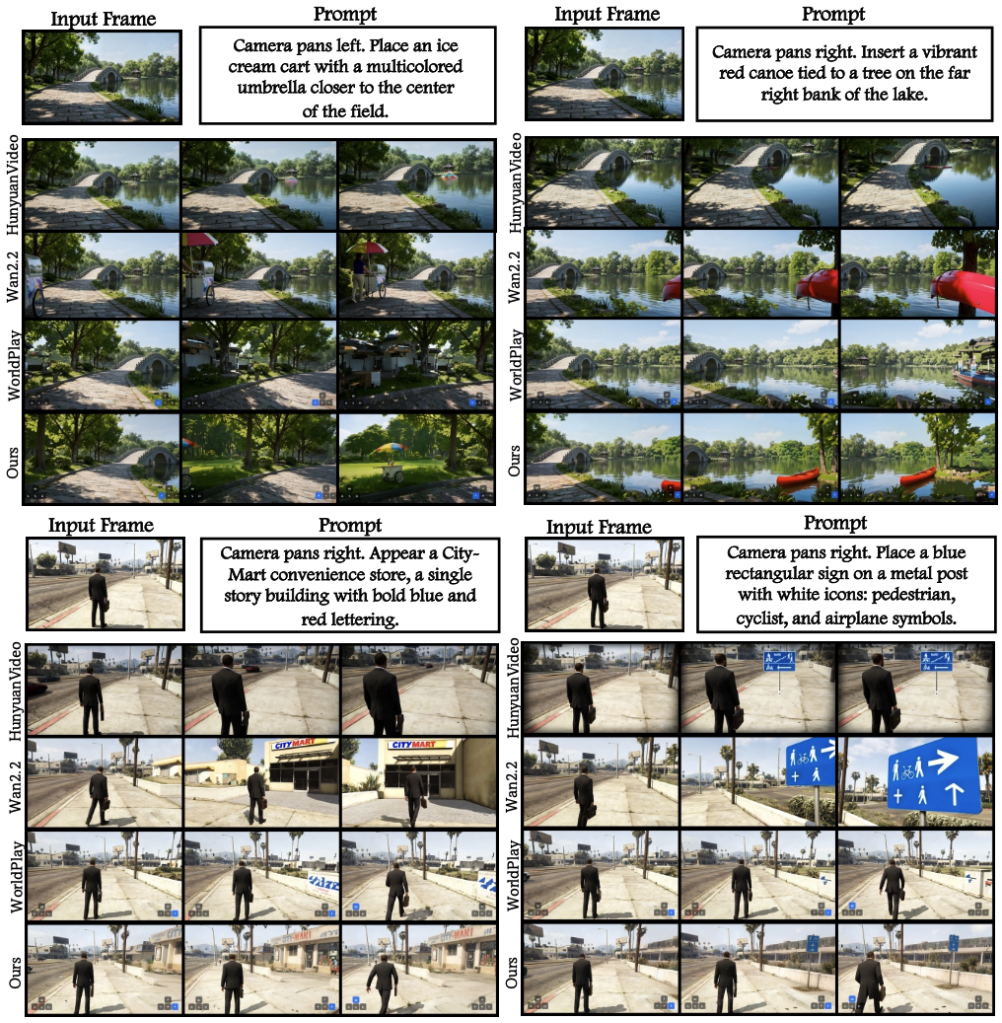

Figure 6: Qualitative comparison against baselines. Our method produces correct placement and action. Baselines omit the concept (HunyuanVideo, WorldPlay) or place it incorrectly (Wan 2.2).

Wan 2.2 introduces it but renders it inaccurately: in the canoe scene (Fig. 6, top-right), the canoe appears in the foreground rather than on the far right bank, and in the sign scene (bottom-right), the rendered icons do not match the prompt. WorldPlay executes the camera trajectory faithfully but typically fails to render the concept (e.g., no convenience store in Fig. 6, bottom-left). Our method introduces the concept in every scene with correct placement while following the camera trajectory.

4.2 Quantitative results

Table 2 reports VBench metrics across the four methods. Our method achieves an Overall Score comparable to Wan2.2 (0.901 vs. 0.905) and leads on Dynamic Degree and Camera Motion, where HunyuanVideo and Wan2.2 score substantially lower than our method. Subject Consistency and I2V Subject/Background reward rollouts whose visual content remains close to the reference or previous frame. Concept spawning, by construction, introduces content not present in the reference frame, so reductions on these dimensions are a predictable consequence. Our method nonetheless improves over WorldPlay on perceptual-quality dimensions, including Aesthetic Quality and Imaging Quality.

To directly measure concept fidelity, we compute *I2V Subject/Background* with the concept image in place of the reference image, capturing how consistently the concept itself is rendered. Our method scores 0.885 on concept-conditioned I2V Subject and 0.907 on I2V Background under this configuration; the baselines do not accept a concept image as input, so this metric applies only to

Table 2: Evaluation across VBench metrics: baselines (top) and ablation study (bottom). Columns: **AQ**: Aesthetic Quality, **BC**: Background Consistency, **DD**: Dynamic Degree, **IQ**: Imaging Quality, **MS**: Motion Smoothness, **SC**: Subject Consistency, **CM**: Camera Motion, **I2VS**: I2V Subject, **I2VB**: I2V Background. Overall: aggregate score reported by VBench. Best in **bold**, second-best underlined.

Method	AQ	BC	DD	IQ	MS	SC	CM	I2VS	I2VB	Overall
Comparison										
HunyuanVideo	0.640	0.928	0.565	0.745	**0.989**	0.893	0.445	0.964	0.969	0.868
Wan 2.2	**0.704**	**0.960**	0.670	**0.755**	0.981	**0.919**	0.860	**0.982**	**0.988**	**0.905**
WorldPlay	0.601	0.923	**1.000**	0.735	0.985	0.880	**1.000**	0.968	0.969	0.895
Ours	0.640	0.924	**1.000**	0.749	0.985	0.862	**1.000**	0.965	0.981	0.901
Ablation: which slots to replace?										
Slot 0	0.616	0.914	1.000	0.747	0.983	0.862	1.000	0.962	0.966	0.891
Slot 1	0.606	0.920	1.000	0.735	0.984	0.875	1.000	0.965	0.967	0.892
All 4 anchors	0.617	0.914	1.000	0.746	0.983	0.865	1.000	0.963	0.966	0.891
Middle slot	0.606	0.923	1.000	0.736	0.984	0.883	1.000	0.966	0.967	0.894
Last slot	0.603	0.901	1.000	0.747	0.982	0.832	1.000	0.813	0.835	0.801
Slot 0 w/o pano	0.602	0.924	1.000	0.749	0.982	0.878	1.000	0.968	0.967	0.894
Last w/o pano	0.596	0.912	1.000	0.748	0.982	0.844	1.000	0.888	0.896	0.843

Table 3: Concept spawning evaluated by detection (left) and human ratings (right). r@τ: fraction of rollouts with the concept detected by GroundingDINO above confidence τ; Conf: mean per-chunk maximum detection confidence. User study ratings are on a 1–5 scale; Fidelity measures how faithfully the concept image is rendered and applies only to methods that accept one.

	Detection				User study			
Method	r@55	r@65	r@75	Conf	Quality ↑	Prompt ↑	Motion ↑	Fidelity ↑
HunyuanVideo	0.79	0.65	0.42	0.61	3.180±1.15	1.554±1.06	1.387±0.94	N/A
Wan 2.2	0.92	0.82	0.64	0.66	3.219±1.15	2.393±1.34	3.058±1.42	N/A
WorldPlay	0.43	0.15	0.08	0.43	3.058±1.14	1.458±1.01	4.438±0.87	N/A
Ours	**0.96**	**0.92**	**0.79**	**0.71**	**3.283±1.21**	**3.961±1.13**	**4.587±0.72**	**3.836±0.96**

our method. Overall, our method successfully introduces user-specified concepts into the world, improves over the base model on perceptual quality.

Beyond video quality, we evaluate whether the requested concept actually appears in the rollout by running GroundingDINO with the concept text prompt on every generated chunk (Table 3). Our method detects the concept in 0.96 of rollouts at r@55 and 0.79 at r@75, whereas WorldPlay, the base model we build on, rarely renders it (0.43 and 0.08). The I2V baselines render the concept more often (0.79 and 0.92 at r@55 for HunyuanVideo and Wan 2.2) but drop sharply at the stricter threshold (0.42 and 0.64 at r@75).

Long-term persistence. We use 50 rollouts with a scripted revisit trajectory: the camera starts at the reference view (frame 1), turns toward the spawn direction (frame 113), away from it (frame 177), and back (frame 241). We report the per-frame maximum GroundingDINO confidence, averaged over rollouts, against WorldPlay given the same reference frame, actions, and a text prompt describing the concept. Where the concept is not visible (frames 1 and 177), both methods sit at the detector's noise floor (0.47–0.50). At the spawn view, SPAWN reaches 0.74 versus 0.45 for WorldPlay, and it retains 0.71 at revisit while WorldPlay stays at 0.43. The DINOv2 [24] similarity between concept crops at first appearance and at revisit is 0.86, indicating that the same instance is retrieved from memory (cf. the lamppost in Fig. 5(a)).

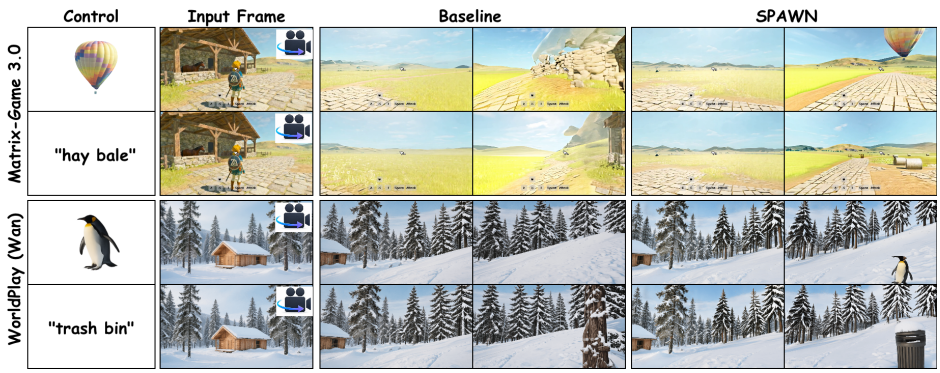

Figure 7: **Cross-model generalization.** SPAWN applied to Matrix-Game 3.0 and WorldPlay (Wan backbone) without per-model tuning. Given the same input frame, camera action, and concept, the base model fills the unseen region with its priors, while SPAWN spawns the requested concept.

Cross-model generalization. We apply SPAWN to two more image-conditioned world models with a persistent anchor: Matrix-Game 3.0 [29] and WorldPlay [31] with a Wan [33] backbone. SPAWN improves r@65 on both (0.60→0.90 and 0.26→0.48). See Fig. 7 for examples.

Ablation study. We ablate our two-slot design against alternative slot-replacement strategies (Table 2, bottom; Fig. 8). **Slot 0** spawns the concept but with weaker identity preservation. **Slot 1** fails to spawn the concept correctly, and the **last slot** causes it to appear abruptly. Our two-slot variant (**Slots 0+1**) introduces the concept smoothly with strong identity preservation. The full ablation is provided in Appendix A.4, including slot replacement, panoramic grounding, and the number of injection chunks, memory insertion, and alternative concept-image construction.



Figure 8: Slot replacement ablation.

User study. We conducted a user study on 20 videos with 30 raters. Raters scored each video on a 1–5 scale across visual quality, prompt following, motion control, and concept image fidelity. Table 3 shows that our method achieves the highest score on all criteria, with the largest margin on prompt following. See Appendix A.6 for details.

5 Limitations

Our method operates entirely at inference time on top of a pretrained autoregressive world model. The available camera trajectories, interaction primitives, and supported action space are therefore inherited from the underlying model, and concept spawning does not extend them. For example, when the base model cannot reliably execute a long forward trajectory through the scene, our method cannot spawn concepts at locations beyond that reachable region.

6 Conclusion

We presented SPAWN, a training-free method for concept spawning in autoregressive world models. SPAWN exploits the structural role of the anchor slots in the context memory, swapping them with an external concept latent over a short injection window before restoring the original anchors. The injected concepts integrate into the scene with consistent lighting, scale, and perspective, while preserving visual identity and temporal coherence across long rollouts and supporting multiple sequential injections, with initial evidence that the approach transfers across world models. In doing so, we unlock an untapped capacity for concept spawning that already exists in autoregressive world models built on image-to-video backbones with a persistent reference anchor, opening a path toward controllable concept introduction without the data and training costs of learning-based approaches.

Acknowledgments and Disclosure of Funding

This work is supported by the National Science Foundation under Grant No. 2543524. We also gratefully acknowledge fal.ai for providing compute support.

References

[1] Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)

[2] Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv preprint arXiv:2311.15127 (2023)

[3] Bruce, J., Dennis, M.D., Edwards, A., Parker-Holder, J., Shi, Y., Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., et al.: Genie: Generative interactive environments. In: Forty-first International Conference on Machine Learning (2024)

[4] Ceylan, D., Huang, C.H.P., Mitra, N.J.: Pix2video: Video editing using image diffusion. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 23206–23217 (2023)

[5] Che, H., He, X., Liu, Q., Jin, C., Chen, H.: Gamegen-x: Interactive open-world game video generation. arXiv preprint arXiv:2411.00769 (2024)

[6] Chen, B., Martí Monsó, D., Du, Y., Simchowitz, M., Tedrake, R., Sitzmann, V.: Diffusion forcing: Next-token prediction meets full-sequence diffusion. Advances in Neural Information Processing Systems **37**, 24081–24125 (2024)

[7] Guo, Y., Yang, C., Rao, A., Liang, Z., Wang, Y., Qiao, Y., Agrawala, M., Lin, D., Dai, B.: Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. arXiv preprint arXiv:2307.04725 (2023)

[8] He, X., Peng, C., Liu, Z., Wang, B., Zhang, Y., Cui, Q., Kang, F., Jiang, B., An, M., Ren, Y., et al.: Matrix-game 2.0: An open-source real-time and streaming interactive world model. arXiv preprint arXiv:2508.13009 (2025)

[9] Henschel, R., Khachatryan, L., Poghosyan, H., Hayrapetyan, D., Tadevosyan, V., Wang, Z., Navasardyan, S., Shi, H.: Streamingt2v: Consistent, dynamic, and extendable long video generation from text. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 2568–2577 (2025)

[10] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)

[11] Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. Advances in neural information processing systems **35**, 8633–8646 (2022)

[12] Huang, X., Li, Z., He, G., Zhou, M., Shechtman, E.: Self forcing: Bridging the train-test gap in autoregressive video diffusion. Advances in Neural Information Processing Systems **38**, 167283–167308 (2025)

[13] Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., Wang, Y., Chen, X., Wang, L., Lin, D., Qiao, Y., Liu, Z.: VBench: Comprehensive benchmark suite for video generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)

[14] Huang, Z., Zhang, F., Xu, X., He, Y., Yu, J., Dong, Z., Ma, Q., Chanpaisit, N., Si, C., Jiang, Y., Wang, Y., Chen, X., Chen, Y.C., Wang, L., Lin, D., Qiao, Y., Liu, Z.: VBench++: Comprehensive and versatile benchmark suite for video generative models. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025). https://doi.org/10.1109/TPAMI.2025.3633890

[15] Kim, J., Kang, J., Choi, J., Han, B.: Fifo-diffusion: Generating infinite videos from text without training. Advances in Neural Information Processing Systems **37**, 89834–89868 (2024)

[16] 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)

[17] Li, J., Tang, J., Xu, Z., Wu, L., Zhou, Y., Shao, S., Yu, T., Cao, Z., Lu, Q.: Hunyuan-gamecraft: High-dynamic interactive game video generation with hybrid history condition (2025), https://arxiv.org/abs/2506.17201

[18] Li, R., Yi, B., Liu, J., Gao, H., Ma, Y., Kanazawa, A.: Cameras as relative positional encoding. arXiv preprint arXiv:2507.10496 (2025)

[19] Li, R., Torr, P., Vedaldi, A., Jakab, T.: Vmem: Consistent interactive video scene generation with surfel-indexed view memory. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 25690–25699 (2025)

[20] 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)

[21] Liu, S., Zhang, Y., Li, W., Lin, Z., Jia, J.: Video-p2p: Video editing with cross-attention control. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8599–8608 (2024)

[22] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499 (2023)

[23] Mao, X., Lin, S., Li, Z., Li, C., Peng, W., He, T., Pang, J., Chi, M., Qiao, Y., Zhang, K.: Yume: An interactive world generation model. arXiv preprint arXiv:2507.17744 (2025)

[24] Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)

[25] Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4195–4205 (2023)

[26] Peng, R., Li, R., Chen, M., Zhou, Y., Guo, Q., Qiu, X.: How attention sinks emerge in large language models: An interpretability perspective. arXiv preprint arXiv:2603.06591 (2026)

[27] Qi, C., Cun, X., Zhang, Y., Lei, C., Wang, X., Shan, Y., Chen, Q.: Fatezero: Fusing attentions for zero-shot text-based video editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15932–15942 (2023)

[28] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)

[29] Skywork AI Matrix-Game Team: Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory. Technical report (2026), https://github.com/SkyworkAI/Matrix-Game/blob/main/Matrix-Game-3/assets/pdf/report.pdf

[30] Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=PxTIG12RRHS

[31] 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)

[32] Tang, J., Liu, J., Li, J., Wu, L., Yang, H., Zhao, P., Gong, S., Yuan, X., Shao, S., Zhang, L., et al.: Hunyuan-gamecraft-2: Instruction-following interactive game world model. arXiv preprint arXiv:2511.23429 (2025)

[33] Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)

[34] Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., Yin, S.m., Bai, S., Xu, X., Chen, Y., et al.: Qwen-image technical report. arXiv preprint arXiv:2508.02324 (2025)

[35] Xiao, G., Tian, Y., Chen, B., Han, S., Lewis, M.: Efficient streaming language models with attention sinks. In: International Conference on Learning Representations. vol. 2024, pp. 21875–21895 (2024)

[36] 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)

[37] Yan, W., Zhang, Y., Abbeel, P., Srinivas, A.: Videogpt: Video generation using vq-vae and transformers. arXiv preprint arXiv:2104.10157 (2021)

[38] Yang, S., Huang, W., Chu, R., Xiao, Y., Zhao, Y., Wang, X., Li, M., Xie, E., Chen, Y., Lu, Y., et al.: Longlive: Real-time interactive long video generation. arXiv preprint arXiv:2509.22622 (2025)

[39] 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)

[40] Yi, J., Jang, W., Cho, P.H., Nam, J., Yoon, H., Kim, S.: Deep forcing: Training-free long video generation with deep sink and participative compression. arXiv preprint arXiv:2512.05081 (2025)

[41] 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. In: Proceedings of the SIGGRAPH Asia 2025 Conference Papers. pp. 1–11 (2025)

[42] Yu, J., Qin, Y., Wang, X., Wan, P., Zhang, D., Liu, X.: Gamefactory: Creating new games with generative interactive videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11590–11599 (2025)

[43] Zhang, Y., Peng, C., Wang, B., Wang, P., Zhu, Q., Kang, F., Jiang, B., Gao, Z., Li, E., Liu, Y., et al.: Matrix-game: Interactive world foundation model. arXiv preprint arXiv:2506.18701 (2025)

Table of Contents

A Supplementary material **15**

A.1 Experiment details 15

A.2 Additional qualitative results 15

A.3 Challenging scenarios 15

A.4 Ablation study 15

A.5 Comparison with edited-view conditioning 18

A.6 User study details 21

A.7 Broader impacts 21

A Supplementary material

A.1 Experiment details

To evaluate SPAWN at scale across 200 rollouts spanning diverse scenes and concept categories, we automate the construction of evaluation prompts and concept images. For each scene in our evaluation set, we use GPT-4o [1] to generate a diverse set of concepts that are physically plausible and visually grounded in the scene context, spanning categories such as people, animals, vehicles, street fixtures, signage, and natural elements. The model is prompted to write specific edit instructions describing what to insert and where, with explicit constraints to avoid abstract or surreal elements and to ensure each prompt introduces a distinct concept.

Given the resulting prompt set, we render the corresponding concept images by editing the panoramic crop of each scene with Qwen-Image-Edit [34]. This pipeline allows us to construct the full evaluation set without manual curation while maintaining concept diversity and grounding.

A.2 Additional qualitative results

Figure A2 shows additional rollouts from SPAWN across two scenes and a range of concept types. In the snowy forest scene (rows 1–4), we spawn an ice sculpture, a motorcycle, a stack of canoes, and a pair of ice skates, each rendered with consistent lighting and snow accumulation appropriate to the environment. In the lakeside park scene (rows 5–8), we spawn a lily pad with a blooming flower, a turtle on a rock, a red umbrella with a bench, and a hanging lantern, each integrated into the scene with plausible placement and reflections on the water surface. Across both scenes, the spawned concepts persist as the camera continues its trajectory and remain coherent with the surrounding environment, demonstrating that SPAWN generalizes across concept categories from small props to environmental fixtures without any per-concept tuning.



Figure A1: Spawning geometrically incompatible concepts into indoor scenes. SPAWN introduces each concept with lighting and perspective consistent with the surrounding scene, even when the concept is implausible in that setting.

A.3 Challenging scenarios

To probe the limits of SPAWN, we spawned concepts that conflict with the scene geometry, such as cars, helicopters, trees, and trucks, into indoor scenes (Fig. A1). In a user study with 12 participants rating each video on a 1–5 scale, the resulting rollouts received 3.34 for visual quality, 4.20 for concept appearance, and 4.78 for motion quality. SPAWN therefore still introduces the requested concept and maintains stable camera motion even when the concept is semantically or physically implausible in the scene, while the lower visual-quality score reflects the difficulty of integrating such concepts convincingly.

A.4 Ablation study

We ablate the design choices of SPAWN along two axes. First, using VBench metrics on the full evaluation set (Table A1), we vary which slots are replaced, the role of panoramic grounding, and the length of the injection window (see Fig. A3 and Fig. A4 for qualitative results). The I2VSC and I2VBC columns directly measure concept fidelity by comparing the rollout against the concept image, and provide the clearest signal across these ablations. Second, using concept-specific detection and identity metrics, we compare our anchor replacement against alternative ways of integrating the same concept image into the world model.



Figure A2: Additional qualitative results of SPAWN in two scenes. In a snowy forest (rows 1–4), we spawn an ice sculpture, a motorcycle, a stack of canoes, and a pair of ice skates; in a lakeside park (rows 5–8), a lily pad with a blooming flower, a turtle on a rock, a red umbrella with a bench, and a hanging lantern. Each concept integrates with the scene's lighting and surfaces, e.g., snow accumulation and reflections on the water, and persists as the camera moves.

Table A1: Evaluation across VBench metrics. Columns: **AQ**: Aesthetic Quality, **BC**: Background Consistency, **DD**: Dynamic Degree, **IQ**: Imaging Quality, **MS**: Motion Smoothness, **SC**: Subject Consistency, **CM**: Camera Motion, **I2VS**: I2V Subject, **I2VB**: I2V Background, **I2VSC**: I2V Subject (Concept), **I2VBC**: I2V Background (Concept). Best in **bold**.

Method	AQ	BC	DD	IQ	MS	SC	CM	I2VS	I2VB	I2VSC	I2VBC	Overall
Which slots to replace?												
Slot 0	0.616	0.914	1.000	0.747	0.983	0.862	1.000	0.962	0.966	0.860	0.876	0.891
Slot 1	0.606	0.920	1.000	0.735	0.984	0.875	1.000	0.965	0.967	0.798	0.819	0.892
All anchors	0.617	0.914	1.000	0.746	0.983	0.865	1.000	0.963	0.966	0.853	0.870	0.891
Middle	0.606	0.923	1.000	0.736	0.984	**0.883**	1.000	0.966	0.967	0.798	0.815	0.894
Last	0.603	0.901	1.000	0.747	0.982	0.832	1.000	0.813	0.835	0.719	0.763	0.801
Effect of panorama cut												
Slot 0 w/o pano.	0.602	0.924	1.000	**0.749**	0.982	0.878	1.000	**0.968**	0.967	0.651	0.666	0.894
Last w/o pano.	0.596	0.912	1.000	0.748	0.982	0.844	1.000	0.888	0.896	0.649	0.669	0.843
How many chunks to inject?												
1 chunk	0.608	0.919	1.000	0.744	**0.985**	0.874	1.000	0.965	0.969	0.793	0.801	0.894
2 chunks	0.613	0.916	1.000	0.745	**0.985**	0.873	1.000	0.967	0.968	0.815	0.821	0.894
10 chunks	0.624	0.914	1.000	0.748	0.984	0.856	1.000	0.963	0.966	0.870	0.885	0.892
SPAWN: slots 0+1, 5 chunks, with panorama												
SPAWN	**0.640**	**0.924**	**1.000**	**0.749**	**0.985**	0.862	**1.000**	0.965	**0.981**	**0.885**	**0.907**	**0.901**

Which slots to replace. Replacing Slot 0 alone spawns the concept but with weaker identity preservation (I2VSC 0.860, I2VBC 0.876) than our two-slot variant. Replacing Slot 1 alone or a middle slot fails to spawn the concept, reflected in the sharp drop in concept fidelity (I2VSC $\approx$0.80, I2VBC $\approx$0.82). Replacing all four anchors preserves scene fidelity (I2VS 0.963) but lowers concept fidelity (I2VSC 0.853, I2VBC 0.870) relative to our two-slot variant. Beyond the anchor slots, we also replace a slot at the middle of the context and the last slot, which holds the most recent temporal-memory latent. Replacing the last slot performs worst on every concept-fidelity metric: because this slot feeds directly into the next chunk, the concept appears abruptly instead of being blended into the scene. Our two-slot variant (Slots 0+1) introduces the concept smoothly while preserving identity (I2VSC 0.885, I2VBC 0.907). Detection metrics confirm this: with the identical concept image, replacing Slot 1 alone reaches only 0.37 r@65, whereas SPAWN reaches 0.92, with substantially higher identity preservation (DINOv2 similarity between the detected concept and the concept image of 0.14 vs. 0.63). If the concept image alone determined appearance and placement, changing only the injection slot could not produce such a large difference; this isolates the contribution of the anchor replacement from the image-editing pipeline.

Panoramic grounding. Without panoramic grounding, the concept image is the bare concept on a blank background, with no surrounding scene context. In this setting, concept fidelity drops sharply (I2VSC $\approx$0.65, I2VBC $\approx$0.67 for both slot configurations), since the model has no spatial context to ground the concept in the scene. Panoramic grounding provides the surrounding scene context that allows the anchor substitution to take hold.

Number of injection chunks. General video-quality and scene-consistency metrics remain stable across injection windows of 1, 2, and 10 chunks. The metric that responds is concept fidelity (I2VSC/I2VBC), which rises from 0.793/0.801 at a single chunk, peaks at our 5-chunk setting (0.885/0.907), and declines slightly at 10 chunks (0.870/0.885). We therefore use 5 chunks: long enough to establish the concept faithfully, yet short enough to avoid the subject-consistency and concept-fidelity drop seen at 10 chunks, where the prolonged injection begins to crowd out scene context. This setting also yields the highest Overall score among all variants (0.901). Combining these choices (Slots 0+1, 5-chunk injection, and panoramic grounding) yields our SPAWN configuration.

Alternative integration strategies. We compare SPAWN (0.92 r@65, Conf 0.71, DINO 0.63) with two alternative ways of integrating the same concept image into WorldPlay. *(i) Permanent anchor replacement.* Keeping the concept latent in the anchor slots for the entire rollout keeps the concept detectable (0.88 r@65) but reduces identity preservation (DINO 0.59), indicating that a bounded injection window better balances concept insertion against preservation of the evolving world state, consistent with the injection-length ablation above. *(ii) Appending to memory.* In WorldPlay, each memory entry carries a camera pose, a temporal index, and 3D RoPE and PRoPE encodings, so a concept image cannot be appended independently of the world representation. We therefore insert the concept latent as an additional entry at the beginning of the context memory while retaining the original anchors. This shifts every subsequent entry by one position, so the positional encodings no longer match the layout the model was trained with. Although the concept is often detected (0.70 r@65), both detection confidence (0.59) and identity preservation (DINO 0.41) drop substantially, and the concept flickers during the rollout.

Concept-image construction. To test whether SPAWN depends on our specific panoramic editing procedure, we construct the concept image by pasting the concept directly onto the original reference frame, preserving the surrounding scene while removing the panoramic editing stage entirely. This simple construction performs comparably to the full pipeline (0.84 vs. 0.92 r@65; DINO 0.65 vs. 0.63; CLIP 0.86 vs. 0.80) and is rated highly in a user study with 12 participants (4.30 visual quality, 4.84 concept appearance, 4.88 motion quality on a 1–5 scale). Together with the blank-background ablation above, this indicates that SPAWN requires a scene-consistent concept image, but not the particular procedure used to produce it.

Robustness to photometric mismatch. We test whether SPAWN requires a concept image that is already photometrically matched to the scene. Before injection, we perturb the concept image with a -1 EV exposure offset, alone or combined with a $+2000$ K white-balance shift (10 rollouts per condition), so that the concept enters the rollout visibly inconsistent with its surroundings. We measure the photometric gap between the concept region and the surrounding scene before injection and after the rollout, using luminance (L*) and CIE-Lab ΔE. As shown in Table A2, the rollout reduces the injected mismatch by about a third in both conditions (e.g., luminance gap 21.00 $\rightarrow$ 14.03 and ΔE 27.68 $\rightarrow$ 19.46 for -1 EV), and the size of this correction grows with the size of the perturbation: the unperturbed concept image, whose initial gap is smaller, is corrected less (11.51 $\rightarrow$ 9.47). SPAWN therefore does not require a perfectly matched concept image; the autoregressive rollout itself adapts the concept toward the scene's lighting and color.

Table A2: Photometric gap between the concept region and the surrounding scene before injection (in) and after the rollout (out). The rollout reduces the mismatch, and the correction grows with the size of the perturbation.

	Luminance (L*)			ΔE		
Condition	In	Out	Reduction	In	Out	Reduction
No perturbation	11.51	9.47	2.04	20.24	15.13	5.11
-1 EV	21.00	14.03	6.97	27.68	19.46	8.22
-1 EV, $+2000$ K	20.52	13.74	6.78	28.46	19.85	8.61

A.5 Comparison with edited-view conditioning

A natural alternative to SPAWN is to condition a model directly on the edited, scene-grounded view in which the concept is already present. This changes the task: the rollout starts from an edited observation containing the concept from the first frame, rather than introducing the concept into an already-running world from the user's original reference image. We nevertheless evaluate this setting, giving HunyuanVideo, Wan 2.2, and WorldPlay the same edited view that SPAWN uses to construct its concept image (Table A3). Once the edited view is the conditioning input, all baselines reproduce the concept with high success and fidelity, as expected since it is already present in their first frame. SPAWN reaches comparable detection rates while starting from the original reference image and introducing the concept during the rollout, so the edited view is not required as the conditioning input for the concept to be realized.

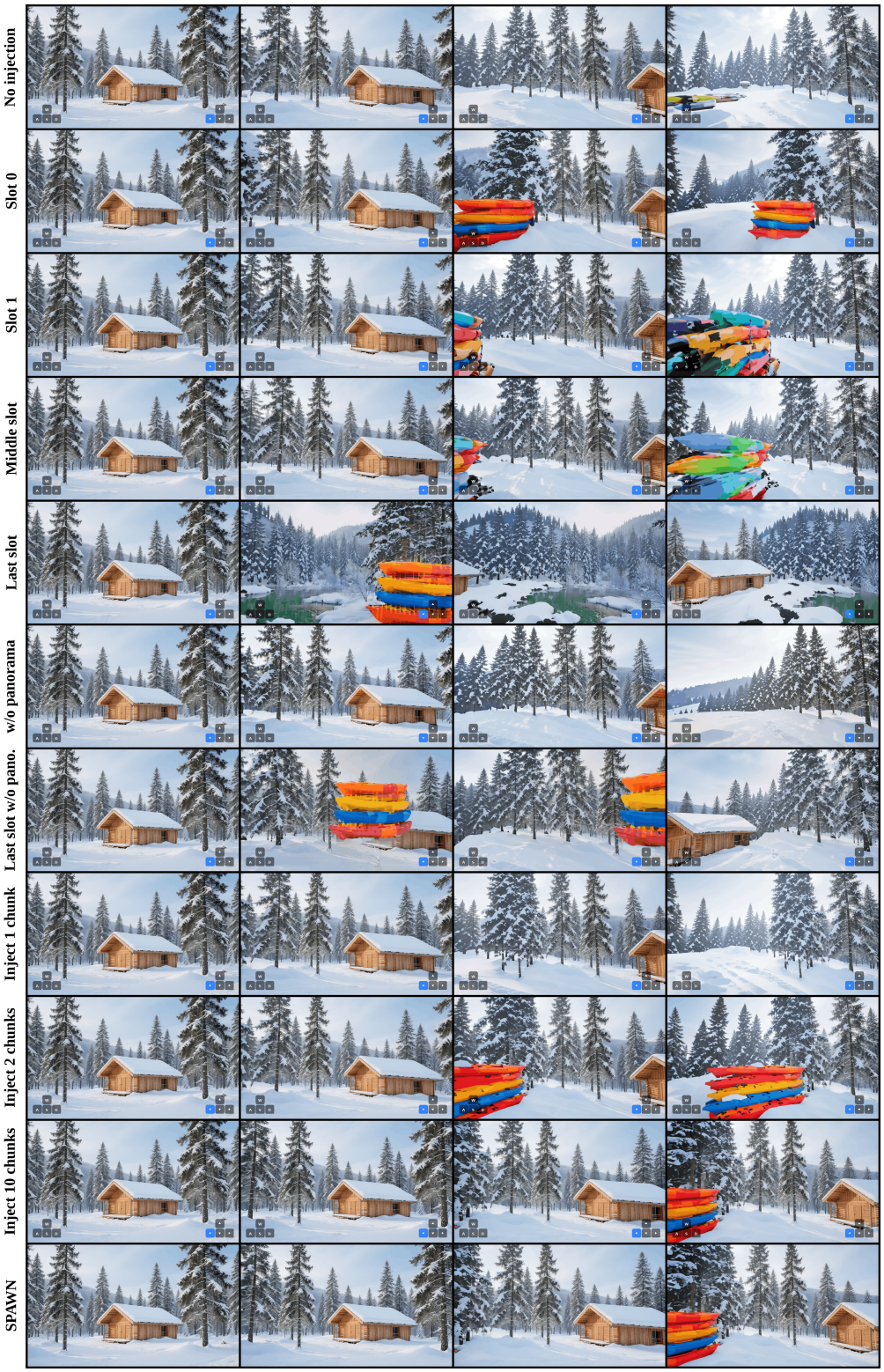

Figure A3: Qualitative ablation in a snowy forest scene, spawning a stack of canoes. Each row shows frames from one variant over the rollout. Replacing Slot 1 or a middle slot yields fragmented, distorted renderings of the concept, and replacing the last slot makes the concept appear abruptly while disrupting the scene. Without panoramic grounding, the concept fails to appear or is placed inconsistently with the scene. A single-chunk injection is too short to establish the concept. SPAWN (bottom) introduces the canoes coherently while preserving the surrounding scene.

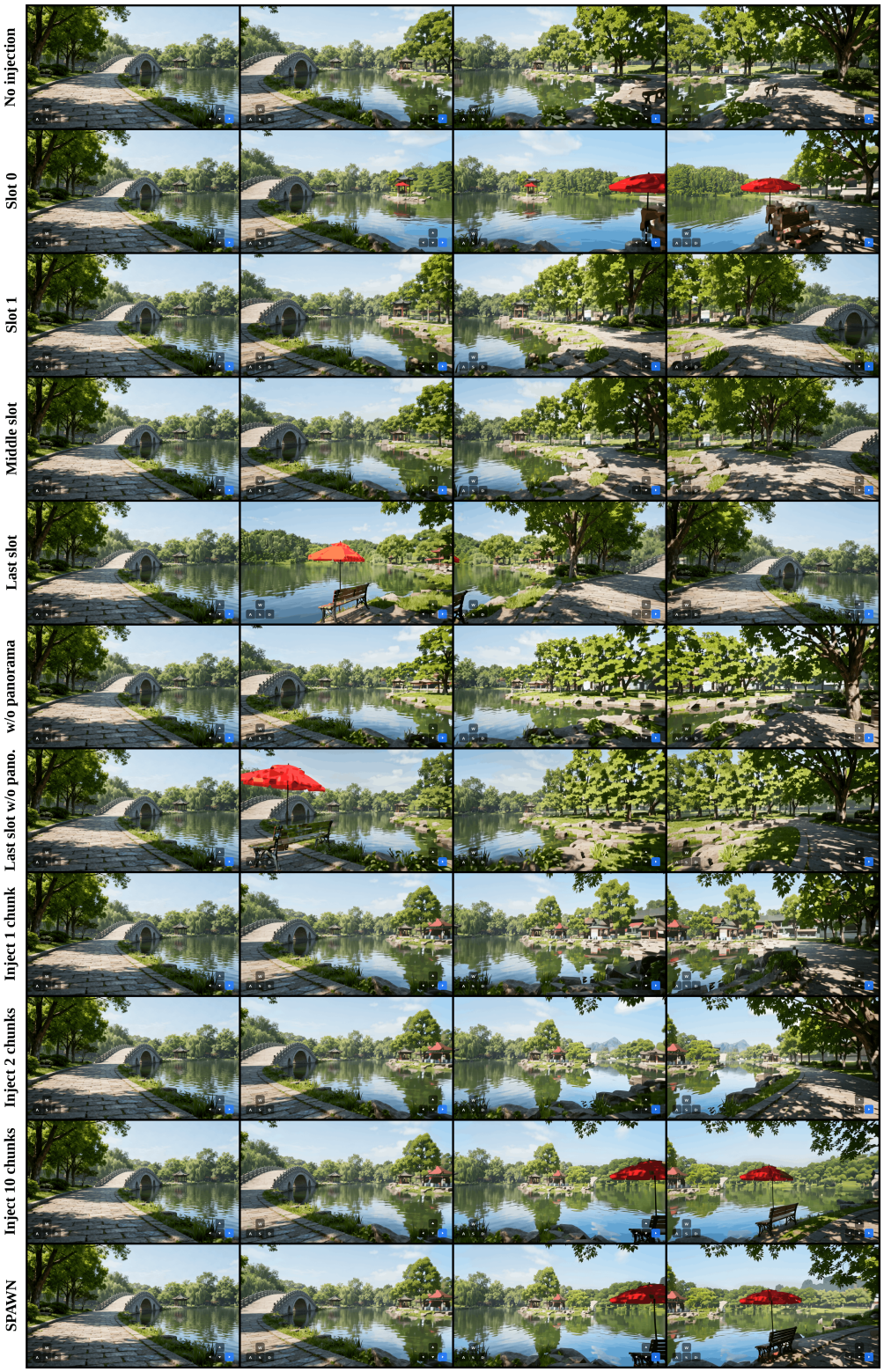

Figure A4: Qualitative ablation in a lakeside park scene, spawning a red umbrella with a bench. Replacing Slot 1 or a middle slot fails to spawn the concept, while replacing the last slot introduces it abruptly. Short injections of 1 or 2 chunks produce an unrelated structure instead of the requested concept, whereas longer injections render it faithfully. SPAWN (bottom) introduces the umbrella and bench smoothly and keeps them consistent as the camera moves.

Table A3: Comparison with baselines conditioned on the edited view, in which the concept is present from the first frame. SPAWN instead starts from the original reference image and introduces the concept during the rollout.

Method	r@55	r@65	r@75	Conf	DINO	CLIP
HunyuanVideo (edited view)	0.98	0.94	0.74	0.78	0.82	0.87
Wan 2.2 (edited view)	0.96	0.92	0.78	0.77	0.79	0.91
WorldPlay (edited view)	0.98	0.92	0.84	0.74	0.68	0.86
SPAWN	0.96	0.92	0.79	0.71	0.63	0.80

A.6 User study details

Study design. We recruited 30 raters through Prolific, compensated in line with Prolific's fair pay guidelines. Each rater evaluated 20 generated videos sampled from our evaluation set, covering a range of scenes, concepts, and visual styles. For each video, raters were shown the text prompt describing the requested concept and the camera action, and where applicable, the concept image used to drive SPAWN. Videos from all four methods (HunyuanVideo, Wan 2.2, WorldPlay, and SPAWN) were included and presented in randomized order to mitigate ordering effects.

Rating dimensions. Raters scored each video on a 1–5 scale across four dimensions:

- **Visual quality.** *"Rate the visual quality of this video (1 = many artifacts, 5 = no noticeable artifacts). Consider flickering, distortion, warping, and broken geometry."*
- **Prompt following.** *"Rate how well this video follows the text prompt (1 = ignores the prompt, 5 = follows it perfectly)."*
- **Camera action.** *"Rate how accurately this video executes the camera action (1 = wrong direction or no motion, 5 = exact match)."* The specific camera action (e.g., *turn right*, *pan left*) was shown alongside the question.
- **Concept image fidelity.** For SPAWN only, raters were shown the concept image alongside the video and asked to rate how faithfully the concept appears in the rollout, on a 5-point scale from *Very poor* to *Excellent*. Baselines do not accept a concept image as input and were marked N/A on this dimension.

Interface. Each video was embedded inline above its rating questions, with the text prompt and camera action presented immediately above the video. For concept-fidelity questions, two concept images were displayed side by side, labelled a) and b), with the rollout below, and raters rated how faithfully the video reproduces a specified one of them. Screenshots of the rating interface are shown in Figures A5 and A6.

Results. Across all four dimensions, SPAWN scores highest, with the largest margin on prompt following (3.961 vs. 2.393 for the next-best method). The visual-quality scores are tightly clustered across methods (3.06–3.28), indicating that none of the rollouts exhibit catastrophic artifacts and that the differences between methods are concentrated in controllability rather than image quality. Motion-control scores reflect the gap between general I2V backbones and autoregressive world models: HunyuanVideo and Wan 2.2, which encode camera motion through the text prompt rather than an action interface, score substantially lower (1.39 and 3.06) than WorldPlay and SPAWN (4.44 and 4.59), consistent with the camera-motion findings in our automatic evaluation. Concept-image fidelity, applicable only to SPAWN, averages 3.84, confirming that the concept rendered in the rollout is recognizably the one provided as input.

A.7 Broader impacts

SPAWN lowers the cost of controllable world generation, which can benefit game prototyping, interactive storytelling, and simulation for embodied AI. As with other generative video methods, it could also be misused to create misleading footage, for example by inserting objects or people into realistic scenes. SPAWN inherits the safeguards and biases of the underlying world model and image editor, and we encourage releasing outputs with clear disclosure that they are synthetic.

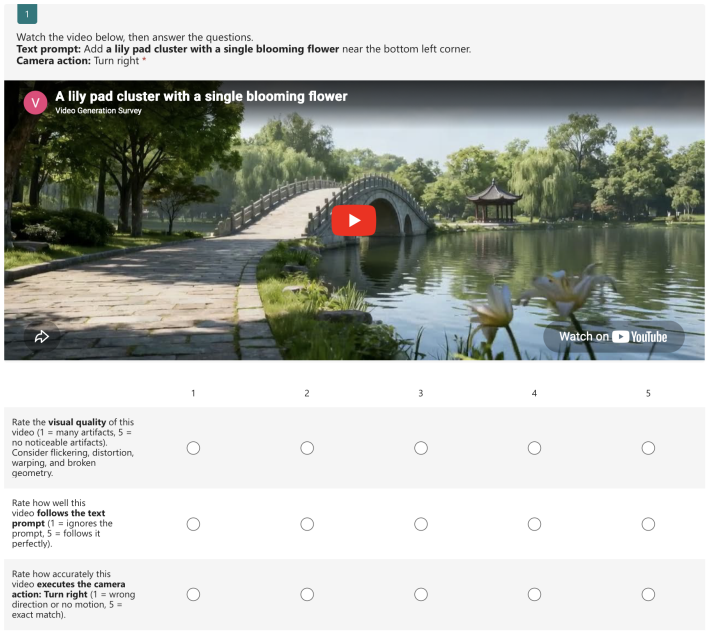

Figure A5: User study interface for general rating questions. Raters watched each video and scored, on a 1–5 scale, its visual quality, how well it follows the text prompt, and how accurately it executes the specified camera action.

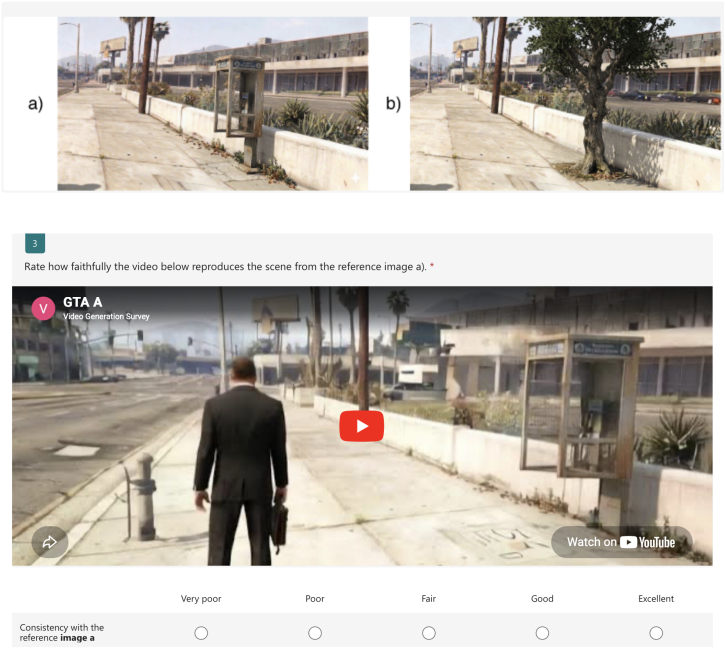

Figure A6: User study interface for concept-image fidelity. Two concept images are shown side by side, labelled a) and b), followed by the rollout. Raters rate how faithfully the video reproduces the specified concept image on a five-point scale from *Very poor* to *Excellent*.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The claims outlined in the abstract and introduction are well supported by the paper's results and contributions, as demonstrated through both quantitative and qualitative evaluations.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Section 5 discusses limitations inherited from the base world model, such as the camera trajectories and action space it supports. Appendix A.3 examines challenging scenarios in which concepts conflict with the scene's geometry, along with the resulting drop in visual quality.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: No formal theoretical results, assumptions, or proofs are provided in the paper. Rather, it introduces a training-free inference-time mechanism for concept spawning in autoregressive world models, validated through comprehensive qualitative and quantitative experiments. As such, this criterion does not apply.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: The paper describes the experimental setup, methodology, and evaluation procedures in detail to allow the main results to be reasonably reproduced.

Guidelines:

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

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

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

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

   Answer: [Yes]

   Justification: The experimental setup section includes detailed instructions for reproducing the experiments. Our code will be released publicly, with a README, license file, and usage instructions to ensure reproducibility and clarity.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: The paper describes the experimental setup and evaluation procedures clearly, enabling understanding and interpretation of the results, while traditional training-specific details are not applicable.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: User-study ratings are reported as mean and standard deviation across raters. VBench and detection metrics are reported as means over 200 rollouts.

Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The experimental setup section includes details about the computational resources used, such as the type of hardware, providing context for how the experiments were conducted.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The work conforms to the NeurIPS Code of Ethics. Human raters were recruited through Prolific with informed consent and fair compensation, and no personal data was collected.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: The paper discusses both positive and negative societal impacts, highlighting potential benefits while also acknowledging risks such as biases and possible misuse.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: We release inference code and a benchmark of generated rollouts built on publicly available models. We do not release new model weights, so SPAWN poses no risk beyond that of the underlying models.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

Justification: All external resources, including datasets, models, and code, are properly credited in the paper. Their usage complies with the respective terms.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

Answer: [Yes]

Justification: We release two new assets: (i) the SPAWN code, with a README, license, and usage instructions; and (ii) a benchmark dataset of 200 rollouts with their reference images, concept images, prompts, and camera actions.

Guidelines:

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

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

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

Answer: [Yes]

Justification: We conducted three user studies. The paper details the evaluation procedure and the questions presented to participants, while screenshots of the interface and instructions are provided in the supplementary material. Participants were compensated in line with Prolific's fair pay guidelines.

Guidelines:

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

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

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

    Answer: [Yes]

    Justification: All user studies involved only rating generated videos on a 1–5 scale and posed minimal risk, with no risks beyond everyday computer use. All participants provided informed consent, and no personally identifiable information was collected.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: GPT-4o is used to generate the evaluation concepts and edit instructions for our benchmark. It is used only to build the evaluation set and is not part of the method itself.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.