WORLD OBSERVER: JOINT ACTOR-OBSERVER GENERATION FOR PERSISTENT WORLD MODELING

Anonymous authors
Paper under double-blind review

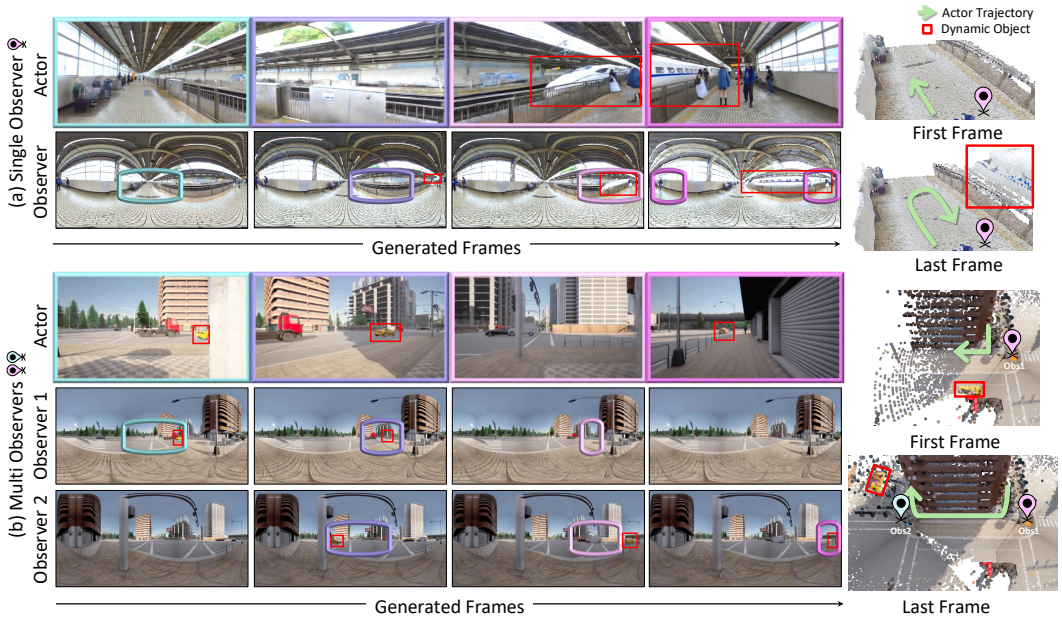


Figure 1: **Teaser.** World Observer tracks out-of-view dynamics via panoramic observers. A single observer maintains out-of-view state; multiple observers extend coverage for consistent re-entry.

ABSTRACT

How can a world model continuously observe regions beyond the actor's current view? Video world models simulate how an environment evolves from an agent's actions, yet remain actor-centric. Once an object leaves the actor's view, they lose direct evidence of its evolution, often failing to preserve its state and dynamics upon re-entry. To address this, we introduce **World Observer**, which decouples observing from acting by jointly generating a perspective *actor* for the agent-centric view with one or more panoramic *observers* that watch selected world regions. This allows objects that leave the actor's view to remain visually evolving in an observer, so their updated states are reflected when they re-enter. We ground both streams in the initial panorama through warping and introduce an *Observer Sink* of high-resolution perspective references that restores fine appearance when a region re-enters the actor view. Since the observers are decoupled from the actor, they can be placed freely across the scene and driven by control signals to steer out-of-view evolution. To evaluate out-of-view evolution, we further introduce world-space metrics and a benchmark spanning real and synthetic scenes. World Observer substantially improves out-of-view dynamics while remaining competitive in visual fidelity, camera control, and 3D adherence.

1 INTRODUCTION

World models (NVIDIA et al., 2025; Robbyant Team et al., 2026; Team HY-World et al., 2026; DreamX Team et al., 2026; Wang et al., 2026b; Dai et al., 2025; Seo et al., 2026) simulate how

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

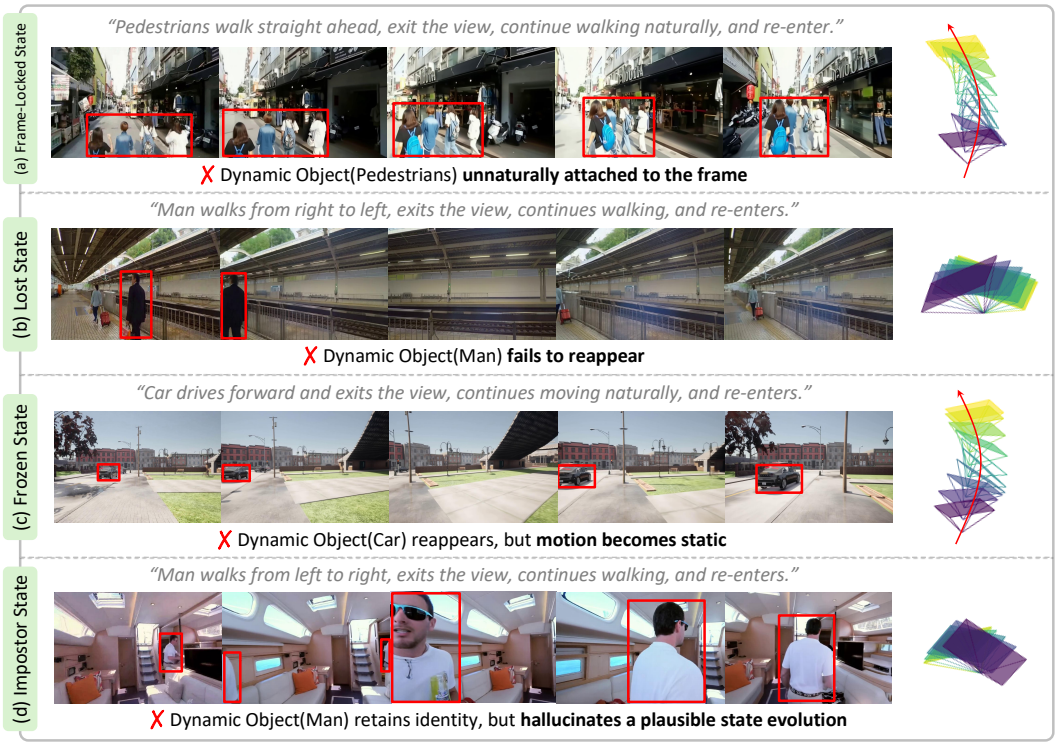


Figure 2: **Motivation.** When dynamic objects leave the actor's view, existing models exhibit four failures: (a) frame-locked, (b) lost, (c) frozen, and (d) impostor state. These failures show the difficulty of maintaining and evolving states without direct observations during out-of-view intervals.

environment evolves by generating future observations conditioned on an agent's actions. Existing models remain *actor-centric*, updating the world primarily through what the actor currently observes, so whatever lies outside that view is left without direct visual evidence. A persistent world model should thus keep the world evolving regardless of where the actor looks, as regions outside the view can still affect the scene (Fig. 1(a)) and dynamic content keeps evolving even after it goes out of sight (Fig. 1(b)). Such persistence matters for applications such as embodied navigation (Li et al., 2025a), interactive simulation (Robbyant Team et al., 2026; Wang et al., 2026b; NVIDIA et al., 2025; Zhu et al., 2026; Tang et al., 2026; DreamX Team et al., 2026), and long-horizon planning (Li et al., 2026a), where an agent must reason about a world beyond its current observation.

However, existing world models (NVIDIA et al., 2025; Robbyant Team et al., 2026; Team HY-World et al., 2026; DreamX Team et al., 2026; Wang et al., 2026b; Dai et al., 2025; Seo et al., 2026) often fail to maintain unobserved states. As illustrated in Fig. 2, we observe four characteristic failure modes that arise when an object moves out of the actor view. Fig. 2(a) shows a **frame-locked state**, where an object that should move out of view under camera motion instead remains unnaturally locked to the image frame. Lu et al. (2026) attributes this to models overemphasizing salient image-space subjects. Fig. 2(b) shows a **lost state**, where an object disappears after leaving the actor view and fails to reappear when the actor looks back. Fig. 2(c) shows a **frozen state**, where the object is retained but its dynamics stop evolving while unobserved, so it reappears near its last visible state. Fig. 2(d) shows an **impostor state**, where the object reappears with a plausible but incorrect state evolution that is inconsistent with what actually occurred while it was unobserved.

The core limitation is that world-state maintenance remains tied to the actor's observation, as which regions are continuously represented depends on where the actor looks. Once an object leaves the actor's observation, the model loses direct visual evidence of how its state evolves. Existing approaches compensate through internal memory (Chen et al., 2026b; Wang et al., 2026a; Wu et al., 2025), generative priors (Xu et al., 2026), or explicit state extrapolation (Duan et al., 2026). However, these approaches infer the unseen evolution from previously observed information rather than directly observing what happens after the object leaves the view. As unobserved intervals grow longer or scene dynamics grow more complex, this inference becomes increasingly uncertain, causing state drift (Lu et al., 2026; Ma et al., 2026) and temporal inconsistency (Chen et al., 2026b;a;

Ma et al., 2026). The key challenge is therefore not merely to remember the last observed state, but to *keep regions of interest observable regardless of where the actor looks*. This raises a natural question: *How can a world model continuously observe regions beyond the actor's current view?*

Our answer is to decouple *observing* from *acting*. We introduce **World Observer**, which jointly generates one or more *observers* that watch chosen regions together with an *actor* that renders the agent-centric view. To cover a broad surrounding region from a single observer location, we instantiate each observer as a 360° panoramic video. Generated jointly, the actor and observers share a single world state, so a state seen by the actor continues in its observers and vice versa. Objects covered by an observer thus remain represented and evolve even outside the actor's view and reappear with updated states, mitigating the frame-locked, lost, frozen, and impostor failures.

A panoramic observer provides broad spatial coverage, but transferring its evolving world state to the perspective actor requires bridging geometric and appearance differences between the two representations. By warping the initial panorama into each viewpoint, we provide explicit geometric correspondences between overlapping regions of the actor and observer views. However, the observer's panoramic distortion limits the fine appearance detail available to the actor. We therefore introduce an *Observer Sink*, a set of fixed high-resolution perspective references cropped from the initial panorama and accessed through shared attention. Together, the evolving observer and these appearance references help the actor render returning regions with updated states and fine detail.

Since observers are not tied to the actor, we can choose which regions to be observable independently of the actor's position and view. In Fig. 1(b), fixed observers watch regions the actor cannot see, allowing occluded objects to remain tracked and re-enter consistently. Decoupling the observer from the actor also enables independent observer control, allowing us to guide how out-of-view regions evolve (Fig. 1(a)). Training such observers requires viewpoint-synchronized captures that are difficult to obtain in the real world, so we complement real panoramic videos with a synthetic CARLA (Dosovitskiy et al., 2017) dataset that provides these observer configurations.

Finally, we evaluate whether objects continue to *evolve* correctly while outside the actor view. Existing benchmarks (Ma et al., 2026; Lu et al., 2026; Chen et al., 2026b) largely focus on single, centered targets, while VLM-based metrics (Lu et al., 2026; Ma et al., 2026; Chen et al., 2026a) mainly assess visual plausibility, so neither directly measures whether objects in moving, multi-object scenes evolve coherently while out of view. We therefore introduce world-space metrics that lift objects into 3D with a segmentation model (Carion et al., 2026) and a metric-scale depth estimator (Lin et al., 2025). *OOV-*D_{gt} measures whether out-of-view motion agrees with ground-truth dynamics, *OOV-*D_{self} measures whether generated dynamics stay self-consistent through the interval, and *OOV-F* reports valid leave-and-return cases. World Observer significantly improves out-of-view dynamics while remaining competitive on visual fidelity, camera control, and 3D adherence.

Our contributions are as follows.

- We introduce **World Observer**, which decouples acting from observing, generating a perspective actor with panoramic observers that keep the regions beyond the actor's view.
- We decouple the observer from the actor, allowing us to freely place observers alongside the actor, at fixed viewpoints, or across scenes for broader coverage and out-of-view control.
- We introduce *OOV-*D_{gt} and *OOV-*D_{self} for 3D out-of-view dynamic consistency, *OOV-F* for valid leave-and-return cases, and a benchmark spanning real and synthetic scenes.

2 Data Construction

World Observer is trained on temporally synchronized actor and observer videos, built from two sources. Real panoramic video grounds generation in real-world appearance, while a synthetic simulator provides the decoupled and multi-observer configurations that real captures rarely contain.

Real-world dataset. From an existing panorama corpus (Luo et al., 2026; Xia et al., 2025; Chen et al., 2024), we curate high-resolution videos with stable motion and stabilize each into an upright, non-rotating equirectangular observer sequences. We then render synchronized perspective actor videos from the stabilized panorama under randomized trajectories and estimate per-frame depth with Depth Anything 3 (Lin et al., 2025) for the geometry-aware condition (Sec. 3.2). Since both



Figure 3: **Dataset Overview.** Real-world (top) and synthetic (bottom) examples with moving and two-fixed-observer configurations, respectively. Colored boxes show cross-view correspondences, while the rightmost diagrams show camera configurations and trajectories.

views come from the same panorama, real pairs are co-located, so we use synthetic data for observers decoupled from the actor and for multi-observer settings. Details are in Appendix B.2.

Synthetic dataset. We render videos in CARLA (Dosovitskiy et al., 2017) using the same stabilization and augmentation as the real data. The simulator lets us place observers independently of the actor, from one decoupled observer to several synchronized observers at different locations, providing complementary coverage (Fig. 3). Details are in Appendix B.3.

3 METHOD

World Observer jointly generates video streams of two roles that share a single world. An *actor* renders the agent's local view, while one or more panoramic *observers* maintain broader world state and dynamics beyond it. Both streams are generated by a single pretrained video Diffusion Transformer (DiT) (NVIDIA et al., 2025) in the latent space of a 3D VAE, and each stream is conditioned on its own text prompt and camera trajectory. Long-horizon videos are generated autoregressively in chunks of T frames, using tail latents of the previous chunk as history. Fig. 4 shows the overview.

Given an initial actor view x_0^{a} and initial observer panoramas $\{x_0^{\mathrm{o}_n}\}_{n=1}^{N}$, World Observer generates an actor video $\mathbf{X}^{\mathrm{a}}=\{x_t^{\mathrm{a}}\}_{t=0}^{T-1}$ and N observer videos $\{\mathbf{X}^{\mathrm{o}_n}\}_{n=1}^{N}$. For clarity, we present the single-observer case ($N=1$) and write $\mathbf{X}^{\mathrm{o}}$. For multi-observer ($N>1$), each observer has its own stream, as described in Sec. 3.3. The actor stream takes a camera trajectory $\mathbf{C}^{\mathrm{a}}=\{c_t^{\mathrm{a}}\}_{t=0}^{T-1}$, a prompt $\mathbf{P}^{\mathrm{a}}$, and noisy latents $\mathbf{Z}^{\mathrm{a}}=\{z_l^{\mathrm{a}}\}_{l=0}^{L-1}$ to produce $\hat{\mathbf{Z}}^{\mathrm{a}}$, where L is the number of latents per chunk. The observer stream is defined analogously with $\mathbf{C}^{\mathrm{o}}$, $\mathbf{P}^{\mathrm{o}}$, and $\mathbf{Z}^{\mathrm{o}}$, producing $\hat{\mathbf{Z}}^{\mathrm{o}}$. For each subsequent chunk, both streams additionally condition on history latents $\mathbf{Z}^{\mathrm{a}}_{\mathrm{hist}}$ and $\mathbf{Z}^{\mathrm{o}}_{\mathrm{hist}}$ from the tail of the previous chunk, where H is the number of history latents.

3.1 JOINT ACTOR-OBSERVER WORLD MODELING

While the actor cannot see beyond its view, the world keeps evolving. We therefore model the panorama as a time-varying *observer* jointly generated with the actor, rather than a static reference. Objects leaving the actor's view keep evolving in the observer, and both streams share this state, so re-entering regions remain consistent with the observer instead of having its out-of-view evolution

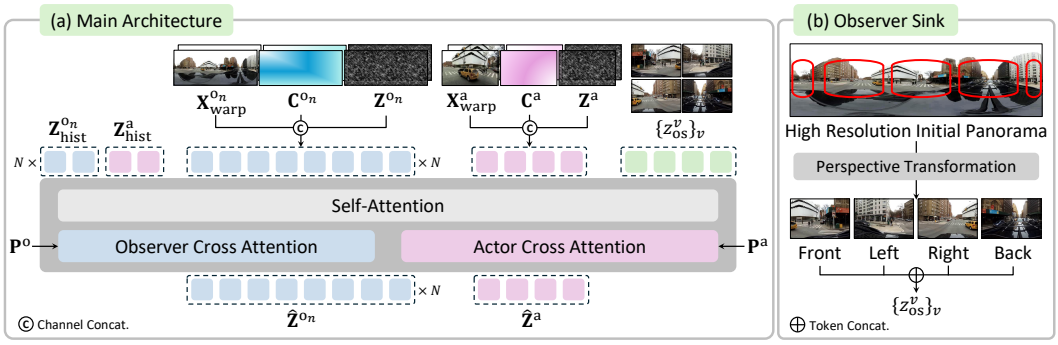


Figure 4: **Model Overview.** (a) World Observer jointly generates actor and panoramic observer streams with a shared DiT. (b) Observer Sink converts the high-resolution initial panorama into four perspective references for fine-grained appearance.

synthesized from scratch. The observer thus serves as an evolving world-state stream rather than static context.

View-time alignment. We concatenate actor and observer latents with their history latents from the previous chunk into a single sequence processed by shared DiT, adding learnable view embeddings to distinguish streams. Since both depict the same world at the same time, we assign matched actor and observer latents identical positions on the RoPE (Su et al., 2023) temporal axis, allowing self-attention to exchange information across corresponding timesteps. As shown in Fig. 5, a query on a re-entering object can attend to the observer region at the same timestep, where its out-of-view state is maintained.

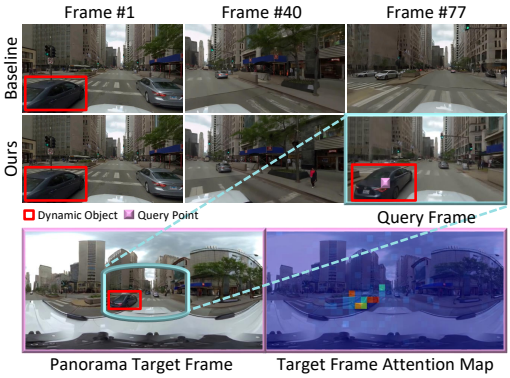


Figure 5: **Visualization of self-attention.** A query on a re-entering object attends to the observer region holding its out-of-view state, while the object fails to re-enter after out-of-view interval in non-joint model (Baseline).

Decoupled actor-observer prompting. The actor and the observer share one evolving world but describe different parts, so we condition them on separate prompts. The actor prompt $\mathbf{P}^{\mathrm{a}}$ specifies events within the actor's local view, while the observer prompt $\mathbf{P}^{\mathrm{o}}$ specifies events across the surrounding world, including regions the actor does not see. Since both streams are jointly generated and synchronized, this gives the observer two roles. It acts as *memory*, since events leaving the actor view continue evolving in the observer, so a subject seen walking away keeps moving while unobserved. It also acts as *control*, since the observer prompt can drive unseen events, such as an off-screen interaction, which the actor then observes when it looks there.

Training objective. We train World Observer with flow matching (Lipman et al., 2022). For each stream $s \in \{\mathrm{a}, \mathrm{o}\}$, we sample Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ and a shared timestep $k \in [0, 1]$, and construct the interpolated latent

$$\tilde{\mathbf{Z}}^s_k = (1-k)\mathbf{Z}^s + k\epsilon. \tag{1}$$

The objective is

$$\mathcal{L} = \mathbb{E}\left[\sum_{s\in\{\mathrm{a},\mathrm{o}\}} \left\| \mathbf{v}^s_\theta(\tilde{\mathbf{Z}}_{\mathrm{seq},k}, \mathbf{P}^{\mathrm{a}}, \mathbf{P}^{\mathrm{o}}, k) - (\epsilon - \mathbf{Z}^s) \right\|_2^2\right], \tag{2}$$

where $\tilde{\mathbf{Z}}_{\mathrm{seq},k}$ denotes the joint actor-observer sequence with interpolated target latents and clean history latents. The loss is applied only to the generated actor and observer latents.

3.2 Panoramic Observer Modeling

Since the panorama provides a shared view of the initial surroundings across all directions, we ground both streams in it through warping, fixing each stream's camera condition and aligning actor

and observer geometry. Because the observer only has to maintain the world state rather than serving as the displayed output, we generate it at a flexible resolution. A panorama, however, inevitably carries geometric distortion, which loses fine appearance and weakens the correspondence between the observer and the actor. We therefore introduce an Observer Sink, a perspective reference from panorama that restores this appearance and correspondence when a region re-enters the actor view.

Panoramic warping. Joint modeling shares information between the streams but does not fix the viewpoint each should render or how the actor and observer views correspond. We ground both in the initial panorama x_0^{o}, whose 360° coverage provide a shared view of the surrounding scene. Using its metric-scale depth D_0 from DA3 (Lin et al., 2025), we warp this shared source into each stream's viewpoints along its own trajectory. For a target frame of stream $s \in \{\mathrm{a}, \mathrm{o}\}$ at pose c_t^s,

$$x^s_{\mathrm{warp},t} = \mathrm{Render}\big(\mathrm{Unproj}(x_0^{\mathrm{o}},\, D_0^{\mathrm{o}}),\; c^s_{0\to t}\big), \tag{3}$$

where $c^s_{0\to t}$ transforms the initial observer panorama to the target pose. Each warped video is encoded by the 3D VAE and channel-wise concatenated with the corresponding noisy latents and a binary validity mask marking covered pixels. Since both streams use the same panorama and depth, they are geometrically consistent by construction, giving shared attention a direct correspondence between the actor view and its observer region. We further channel-wise concatenate a Plücker raymap (Sitzmann et al., 2022) with each stream's latent features to encode its viewpoint.

Decoupled resolution. Because the actor and observer play different roles, their resolutions need not match. The actor is the displayed output and must preserve visual detail, whereas the observer only maintains world state, so we generate it at a lower resolution $R^{\mathrm{o}} < R^{\mathrm{a}}$, where R^{a} is the actor resolution. Coarse spatial-temporal structure, object locations, motion, and visibility changes, is enough for the observer to track out-of-view dynamics, which keeps full-360° coverage tractable.

Observer sink. As a distorted representation, the panorama maintains the state of the full surroundings but lacks the fine appearance the actor needs when it renders a re-entering region. To supply this, we introduce an *Observer Sink*, a high-resolution perspective reference. We crop four perspective views from the high-resolution initial panorama, each 90° apart in yaw, and encode each to a latent z^v_{os} for $v \in \{0, 1, 2, 3\}$. The sink latents are kept fixed rather than denoised as target frames, and we append them to the actor-observer sequence so generated tokens can attend to them,

$$\mathbf{Z}_{\mathrm{seq}} \leftarrow \big[\, \mathbf{Z}_{\mathrm{seq}};\; \{z^v_{\mathrm{os}}\}_v \,\big], \qquad \mathbf{p}_{\mathrm{seq}} \leftarrow \big[\, \mathbf{p}_{\mathrm{seq}};\; \{\, H{+}L{+}\Delta_{\mathrm{os}} + v\,\delta_{\mathrm{os}} \,\}_v \,\big], \tag{4}$$

where Δ_{os} places the sink outside the generated-frame temporal position and δ_{os} denotes the positional stride between sink views. The observer captures evolution over time, while the Observer Sink preserves appearance from the initial panorama, allowing the actor to render re-entering regions with updated states and fine detail.

3.3 Flexible Observer Placement

Why flexible observers. An observer that simply follows the actor is constrained by the actor trajectory and cannot independently monitor a region of interest. However, important state updates may occur away from the actor, and such regions should remain observable regardless of where the actor moves. We therefore also consider a flexible single-observer setting, where the observer trajectory $\mathbf{C}^{\mathrm{o}}$ can be spatially separated from and move independently of the actor trajectory $\mathbf{C}^{\mathrm{a}}$, enabled by our synthetic data construction in Sec. 2. Beyond this, a single observer may still be insufficient when relevant regions are spatially separated or mutually occluded. We therefore further extend the formulation to $N > 1$ observers, allowing multiple observer streams $\{\mathbf{X}^{\mathrm{o}_n}\}_{n=1}^N$ to maintain evolving state across different parts of the shared world.

Multi-observer joint modeling. For $N > 1$, we instantiate one panoramic stream for each observer $\mathbf{X}^{\mathrm{o}_n}$ and add an observer-specific learnable embedding to distinguish the streams. The actor stream and all N observer streams are concatenated into a single sequence and jointly generated by the same DiT. For each actor trajectory frame, we select the initial observer panorama $x_0^{\mathrm{o}_n}$ whose corresponding observer pose is closest to the current actor position and use it as the geometric source for actor-view warping. This keeps the actor condition grounded in a nearby panorama for stable camera control while allowing the geometric reference to transition between observers as the actor moves through the scene. For each generation chunk, the Observer Sink is constructed from the observer selected for that chunk, so its high-resolution appearance reference remains consistent with the geometric source. Meanwhile, all observer streams continue to evolve jointly, maintaining a coherent shared world across their different regions of coverage.



Figure 6: **Qualitative Results.** World Observer supports (a) state memory, (b) observer-prompted control, (c) long-horizon generation, and (d) multi-observers. Insets show the observer views.

4 Experiments

4.1 Out-of-View Dynamics Metric

Existing evaluations rely largely on semantic cues or VLM judgments, which struggle to quantify how multiple objects move while out-of-view. We instead measure out-of-view dynamics explicitly in world space. Using a segmentation model (Carion et al., 2026) and a metric-scale depth estimator (Lin et al., 2025), we lift each object into 3D and track its position over time, turning unobserved motion into real-world displacement with comparable direction and magnitude across methods.

We measure two aspects of displacement for objects that validly exit and re-enter the actor's view. **OOV-F** reports how often objects follow the ground-truth exit-and-re-entry pattern, capturing the *frame-locked* and *lost* failures with no valid displacement. For objects that exit and return, **OOV-D** measures how well out-of-view motion matches a reference. We use two references. **OOV-D**$_{\text{gt}}$ compares motion with the ground-truth dynamics specified by the prompt, while **OOV-D**$_{\text{self}}$ checks whether the object continues pre-exit motion. Accordingly, OOV-D$_{\text{gt}}$ is more sensitive to frozen motion, whereas OOV-D$_{\text{self}}$ is more sensitive to impostor motion. Details are in Appendix C.3.

4.2 Implementation Details

Model and training setup. World Observer fine-tunes Cosmos-Predict2.5 (NVIDIA et al., 2025) with AdamW (Loshchilov & Hutter, 2019) at a learning rate of $4.8\text{e}{-5}$ on 8 NVIDIA H100 GPUs. We train on 138K clips sampled with diverse trajectories from 23K real videos (Luo et al., 2026; Xia et al., 2025; Chen et al., 2024) and with 72K clips from 12K synthetic videos (Dosovitskiy et al., 2017), with $T{=}77$ frames. The single-observer model trains for 12K iterations with batch size 16. The multi-observer model starts from this checkpoint and is trained for additional 6K iterations with total batch size 8 on synthetic data using $N{=}2$ observers. We render actor at $R^{\text{a}}{=}1280 \times 704$ and observer at $R^{\text{o}}{=}640 \times 320$, with Observer Sink offset $\Delta_{\text{os}}{=}50$. Further details are in Appendix B.1.

Evaluation benchmarks. We build held-out benchmarks on real panoramic video and synthetic CARLA video, unseen during training, with 100 sequences each (200 in total) of $T{=}77$ frames. We



Figure 7: **Qualitative comparisons with other models.** Red boxes mark target objects before exit and after re-entry. While existing models often lose, freeze, or inconsistently reconstruct them, World Observer better preserves object identity and state evolution across out-of-view intervals.

Table 1: **Quantitative comparison with existing methods.** Results are reported as Real-OOV-Bench / Synthetic-OOV-Bench. RotErr is - for panorama outputs cropped with the ground-truth rotation, where motion is fixed by construction. World Observer (Multi) is evaluated only on Synthetic-OOV-Bench, as multi-observer conditions are scarce in the real world.

Method	#Param	Visual & Temporal Fidelity			Camera Control		3D Adherence		Out-of-View Dynamics		
		FID↓	FVD↓	Img.Q.↑	RotErr↓	TransErr↓	mPSNR↑	mLPIPS↓	OOV-F↑	OOV-D_{gt} ↑	OOV-D_{self} ↑
Panorama World Models											
OmniRoam (Liu et al., 2026)	1.3B	42.13 / 31.56	404.7 / 418.9	0.387 / 0.390	- / -	0.616 / 0.072	18.523 / 15.934	0.317 / 0.388	0.563 / **0.875**	0.055 / 0.107	0.063 / 0.211
PanoWorld (Li et al., 2026b)	5B	62.48 / 55.73	677.4 / 557.2	0.397 / 0.377	- / -	0.659 / 0.090	17.055 / 15.426	0.383 / 0.418	0.476 / 0.289	0.177 / 0.292	0.142 / 0.216
Video World Models											
HyDRA (Chen et al., 2026b)	1.3B	50.99 / 67.94	615.9 / 944.4	0.637 / 0.656	0.140 / 0.164	0.561 / 0.510	14.859 / 13.455	0.407 / 0.490	0.416 / 0.258	0.274 / 0.266	0.292 / 0.253
HY-World 1.5 (HunyuanWorld, 2025)	8B	51.61 / 42.63	518.7 / 509.1	0.628 / 0.640	0.479 / 0.419	0.511 / 0.122	12.690 / 13.012	0.517 / 0.531	0.128 / 0.133	0.358 / 0.351	0.311 / 0.311
Dream-X (DreamX Team et al., 2026)	5B	42.91 / 38.95	470.5 / 635.6	0.646 / 0.651	0.138 / 0.110	0.478 / 0.066	14.054 / 13.572	0.473 / 0.522	0.484 / 0.352	0.280 / 0.308	0.213 / 0.310
FantasyWorld (Dai et al., 2025)	14B	45.55 / 35.72	511.9 / 385.3	0.641 / 0.679	0.197 / 0.123	0.616 / 0.105	13.241 / 13.848	0.472 / 0.481	0.288 / 0.633	0.382 / 0.338	0.379 / 0.390
LingBot-World (Robbyant Team et al., 2026)	28B	39.87 / 28.75	420.0 / 333.3	**0.668** / 0.675	0.249 / 0.160	0.548 / 0.098	13.329 / 14.094	0.493 / 0.493	0.333 / 0.570	0.337 / 0.340	0.343 / 0.460
Matrix-Game-3 (Wang et al., 2026b)	5B	49.41 / 33.00	541.1 / 393.7	0.659 / **0.689**	**0.089** / 0.053	0.589 / 0.148	11.890 / 12.354	0.507 / 0.517	**0.568** / 0.508	0.066 / 0.125	0.089 / 0.118
World Observer (Single)	2B	**29.30** / 19.65	**291.7** / 196.2	0.647 / 0.641	0.125 / 0.022	**0.372** / 0.029	**18.574** / 18.893	**0.278** / 0.276	0.492 / 0.580	**0.426 / 0.531**	**0.408 / 0.526**
World Observer (Multi)	2B	- / **16.06**	- / **147.0**	- / 0.650	- / **0.014**	- / **0.022**	- / **20.357**	- / **0.240**	- / 0.722	- / 0.528	- / 0.457

evaluate within a single chunk, where the full sequence fits in context and no memory retrieval is needed. This isolates whether the model genuinely maintains out-of-view dynamics, since under an unlimited-memory single chunk a failure cannot be attributed to limited context or missing retrieval. Each sequence uses a back-and-forth camera rotation with random angle and speed and contains one or more objects undergoing natural out-of-view motion, also provided as a text prompt. References come from held-out viewpoints so the model cannot access the ground-truth target view.

Baselines. We compare against general video world models (Robbyant Team et al., 2026; DreamX Team et al., 2026; Wang et al., 2026b; Dai et al., 2025; HunyuanWorld, 2025), panorama world models (Liu et al., 2026; Li et al., 2026b), and the out-of-view method (Chen et al., 2026b). Since these models do not all accept our inputs, we provide each baseline with its supported conditions. For panorama world models, whose output is a full panorama rather than a perspective view, we crop a perspective view from the output using the ground-truth camera rotation, making their output directly comparable to our perspective actor. Additional adaptation details are in Appendix C.5.

Metrics. Visual and temporal quality use FID (Heusel et al., 2018) and FVD (Unterthiner et al., 2019), and VBench (Huang et al., 2023) Image Quality. **Camera-following accuracy** uses Rotation Error (RotErr) and Translation Error (TransErr). **3D adherence** uses masked PSNR and LPIPS (Zhang et al., 2018) on static regions segmented with SAM3 (Carion et al., 2026), since generated dynamics need not match the ground truth. **Out-of-view dynamics** is measured with OOV-F, OOV-D_{gt}, and OOV-D_{self}, defined in Sec. 4.1, where OOV-F reports how often objects validly exit and re-enter and the two OOV-D scores measure how their out-of-view motion evolves.

4.3 Results

4.3.1 Generation results.

Fig. 6 shows the capabilities of World Observer. (a) **State memory**: an object leaving the actor's view keeps evolving in the observer and re-enters with a consistent, updated state rather than being

Table 2: **Ablation study on Synthetic-OOV-Bench.** We evaluate visual and temporal fidelity, camera-following accuracy, 3D adherence, and out-of-view dynamics. For a fair comparison, w/o Observer and w/o Actor generation models receive the same conditioning as the full model.

	Visual & Temporal Fidelity			Camera Control		3D Adherence		Out-of-View Dynamics		
Variant	FID↓	FVD↓	Img.Q.↑	RotErr↓	TransErr↓	mPSNR↑	mLPIPS↓	OOV-F↑	OOV-D_{gt} ↑	OOV-D_{self} ↑
Backbone (raymap only)	36.59	450.5	**0.674**	0.595	0.134	12.463	0.543	0.070	0.332	0.414
w/o Observer generation	24.82	243.1	<u>0.659</u>	0.057	0.073	16.997	0.337	<u>0.674</u>	0.387	0.430
w/o Actor generation	21.38	259.9	0.433	N/A	0.048	18.591	0.315	0.554	0.471	<u>0.499</u>
w/o Observer Sink	20.33	214.0	0.646	0.026	0.031	18.221	0.303	0.545	0.506	0.470
World Observer (Single)	<u>19.65</u>	<u>196.2</u>	0.641	<u>0.022</u>	<u>0.029</u>	<u>18.893</u>	<u>0.276</u>	0.580	**0.531**	**0.526**
World Observer (Multi)	**16.06**	**147.0**	0.650	**0.014**	**0.022**	**20.357**	**0.240**	**0.722**	<u>0.528</u>	0.457

lost, frozen, or hallucinated. (b) **State controllability**: the observer prompt drives objects outside the actor's field of view, and triggered events continue unfolding until re-entry. (c) **Autoregressive generation**: World Observer rolls out chunk by chunk over long-horizon while keeping the actor and observer consistent. (d) **Multi-Observer generation**: observers placed at specific locations cover regions occluded from the actor, so hidden objects remain tracked and re-enter consistently.

4.3.2 Comparison with other models.

As shown in Tab. 1 and Fig. 7, World Observer performs favorably in visual and temporal fidelity, camera-following accuracy, and 3D adherence, while achieving the strongest out-of-view dynamics. The metrics also reveal baseline failures. Low OOV-F means objects are frame-locked or lost and never validly exit and re-enter. High OOV-F with low OOV-D_{gt} and OOV-D_{self} means objects re-enter but match neither reference, being frozen or moving incoherently. High OOV-D_{gt} with lower OOV-D_{self} means impostor dynamics, where motion follows the prompt but departs from the object's own trajectory. World Observer keeps both OOV-F and OOV-D high, so its re-entering objects keep a consistent, continuing state instead of freezing or turning into impostors. Adding observers further raises OOV-F by covering more regions of interest, while both OOV-D metrics remain competitive.

4.4 Ablation Study

Representative qualitative ablations of the conditioning strategies are shown in Appendix D.2.

Joint generation. As shown in Tab. 2, the Backbone, which uses only raymap conditioning, shows unstable camera control with the lowest OOV scores overall. Removing the observer generates everything from the actor alone, and its relatively high OOV-F alongside low OOV-D_{gt} and OOV-D_{self} indicates that the actor tends to bring previously unseen objects back into frame but with an inconsistent state, an impostor tendency. Generating without the actor instead handles out-of-view content directly in the panorama and therefore reaches strong OOV scores, but the panorama is inherently distorted, so its camera control and visual fidelity fall short. Generating the actor and observer jointly combines these strengths, achieving the highest OOV-D with a stable OOV-F while remaining strong in camera control, 3D adherence, and visual fidelity.

Observer sink. Removing the sink lowers OOV-D_{self} while OOV-D_{gt} stays high, the impostor signature where a re-entering object follows the intended motion but drifts from its appearance. It also reduces structural consistency of re-entering regions.

5 Conclusion

We presented **World Observer**, which decouples observing from acting by jointly generating a perspective actor with panoramic observers that maintain world state beyond the actor's view. This formulation keeps out-of-view regions explicitly represented as the world evolves, allowing their updated states to be reflected when they later re-enter the actor view. Panorama grounding and the Observer Sink preserve geometric consistency and fine appearance across actor and observer views, while decoupled observers enable flexible placement and control of out-of-view regions. We further introduced world-space metrics and benchmarks on real and synthetic scenes to directly evaluate this capability. Across these evaluations, World Observer better preserves out-of-view dynamics while remaining competitive in visual fidelity, camera control, and 3D adherence.

GENERATIVE AI DISCLOSURE

In this work, we used generative AI tools (LLMs) to aid and polish the writing to improve readability and clarity. We have not used generative AI tools for generating synthetic datasets, developing theoretical models, formulating or proving mathematical claims, proposing hypotheses, designing experiments, or interpreting results. All AI-assisted content was thoroughly reviewed, verified, and edited by the authors to ensure accuracy and fidelity to the original research. The authors take full responsibility for the final content and claims presented in this paper.

REFERENCES

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts, 2026. URL https://arxiv.org/abs/2511.16719.

Hao Chen, Yuqi Hou, Chenyuan Qu, Irene Testini, Xiaohan Hong, and Jianbo Jiao. 360+x: A panoptic multi-modal scene understanding dataset, 2024. URL https://arxiv.org/abs/2404.00989.

Haoyu Chen, Kaichen Zhou, Hang Hua, Kaile Zhang, Jingwen Qian, Wufei Ma, Haonan Chen, Chunjiang Liu, Yizhou Zhao, Xiaoyuan Wang, Weiyue Li, Alan Yuille, Paul Pu Liang, and Yilun Du. Memobench: Benchmarking world modeling in dynamically changing environments, 2026a. URL https://arxiv.org/abs/2606.27537.

Kaijin Chen, Dingkang Liang, Xin Zhou, Yikang Ding, Xiaoqiang Liu, Pengfei Wan, and Xiang Bai. Out of sight but not out of mind: Hybrid memory for dynamic video world models, 2026b. URL https://arxiv.org/abs/2603.25716.

Ting-Hsuan Chen, Ying-Huan Chen, Tao Tu, Jie-Ying Lee, Cho-Ying Wu, Fangzhou Lin, Hengyuan Zhang, David Paz, Xinyu Huang, Yuliang Guo, Yu-Lun Liu, Yue Wang, and Liu Ren. Pantheon360: Taming digital twin generation via 3d-aware 360° video diffusion, 2026c. URL https://arxiv.org/abs/2605.25449.

Yixiang Dai, Fan Jiang, Chiyu Wang, Mu Xu, and Yonggang Qi. Fantasyworld: Geometry-consistent world modeling via unified video and 3d prediction, 2025. URL https://arxiv.org/abs/2509.21657.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 1–16. PMLR, 2017. URL https://proceedings.mlr.press/v78/dosovitskiy17a.html.

DreamX Team, Yancheng Bai, Rui Chen, Xiangxiang Chu, Rujing Dang, Hao Dou, Bingjie Gao, Qiwen Gu, Siyu Hong, Jiachen Lei, Geng Li, Jifan Li, Ruimin Lin, Qingfeng Shi, Bingze Song, Lei Sun, Jing Tang, Ruitian Tian, Jun Wang, Jiahong Wu, Pengfei Zhang, Shen Zhang, and Jiashu Zhu. Dreamx-world 1.0: A general-purpose interactive world model, 2026. URL https://arxiv.org/abs/2606.16993.

Zicheng Duan, Jiatong Xia, Zeyu Zhang, Wenbo Zhang, Gengze Zhou, Chenhui Gou, Yefei He, Feng Chen, Xinyu Zhang, and Lingqiao Liu. Liveworld: Simulating out-of-sight dynamics in generative video world models. *arXiv preprint arXiv:2603.07145*, 2026.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. URL https://arxiv.org/abs/1706.08500.

Xun Huang, Zhengqi Li, Guande He, Mfingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion, 2025. URL https://arxiv.org/abs/2506.08009.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Vbench: Comprehensive benchmark suite for video generative models, 2023. URL https://arxiv.org/abs/2311.17982.

Team HunyuanWorld. Hy-world 1.5: A systematic framework for interactive world modeling with real-time latency and geometric consistency. *arXiv preprint*, 2025.

Le Jiang, Xiangyu Bai, Bishoy Galoaa, Shayda Moezzi, Caleb James Lee, Tooba Imtiaz, Edmund Yeh, Jennifer Dy, Yanzhi Wang, and Sarah Ostadabbas. Panoworld: Geometry-consistent panoramic video world modeling, 2026. URL https://arxiv.org/abs/2605.15391.

Nikolai Kalischek, Michael Oechsle, Fabian Manhardt, Philipp Henzler, Konrad Schindler, and Federico Tombari. Cubediff: Repurposing diffusion-based image models for panorama generation, 2025. URL https://arxiv.org/abs/2501.17162.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. Hunyuanvideo: A systematic framework for large video generative models, 2025. URL https://arxiv.org/abs/2412.03603.

Bohan Li, Zhuang Ma, Dalong Du, Baorui Peng, Zhujin Liang, Zhenqiang Liu, Chao Ma, Yueming Jin, Hao Zhao, Wenjun Zeng, and Xin Jin. Omninwm: Omniscient driving navigation world models, 2025a. URL https://arxiv.org/abs/2510.18313.

Guangyuan Li, Bo Li, Jinwei Chen, Xiaobin Hu, Lei Zhao, and Peng-Tao Jiang. Magicworld: Towards long-horizon stability for interactive video world exploration, 2026a. URL https://arxiv.org/abs/2511.18886.

Haoyuan Li, Dizhe Zhang, Yuemei Zhou, Xiangkai Zhang, Haoran Feng, Xiaofan Lin, Wenjie Jiang, Bo Du, Ming-Hsuan Yang, and Lu Qi. Panoworld: Real-world panoramic generation, 2026b. URL https://arxiv.org/abs/2607.09661.

Lingen Li, Guangzhi Wang, Xiaoyu Li, Zhaoyang Zhang, Qi Dou, Jinwei Gu, Tianfan Xue, and Ying Shan. Cubecomposer: Spatio-temporal autoregressive 4k 360° video generation from perspective video, 2026c. URL https://arxiv.org/abs/2603.04291.

Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory, 2025b. URL https://arxiv.org/abs/2506.18903.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views, 2025. URL https://arxiv.org/abs/2511.10647.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Jinxiu Liu, Shaoheng Lin, Yinxiao Li, and Ming-Hsuan Yang. Dynamicscaler: Seamless and scalable video generation for panoramic scenes, 2025. URL https://arxiv.org/abs/2412.11100.

Yuheng Liu, Xin Lin, Xinke Li, Baihan Yang, Chen Wang, Kalyan Sunkavalli, Yannick Hold-Geoffroy, Hao Tan, Kai Zhang, Xiaohui Xie, Zifan Shi, and Yiwei Hu. Omniroam: World wandering via long-horizon panoramic video generation, 2026. URL https://arxiv.org/abs/2603.30045.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Jinpeng Lu, Dexu Zhu, Haoyuan Shi, Linghan Cai, Guo Tang, Yinda Chen, Jie Cao, Duyu Tang, Yi Zhang, Yong Dai, and Xiaozhu Ju. Current world models lack a persistent state core, 2026. URL https://arxiv.org/abs/2606.20545.

Rundong Luo, Matthew Wallingford, Ali Farhadi, Noah Snavely, and Wei-Chiu Ma. Beyond the frame: Generating 360 panoramic videos from perspective videos, 2026. URL https://arxiv.org/abs/2504.07940.

Ziqi Ma, Mengzhan Liufu, and Georgia Gkioxari. Out of sight, out of mind? evaluating state evolution in video world models, 2026. URL https://arxiv.org/abs/2603.13215.

NVIDIA, :, Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, Daniel Dworakowski, Jiaojiao Fan, Michele Fenzi, Francesco Ferroni, Sanja Fidler, Dieter Fox, Songwei Ge, Yunhao Ge, Jinwei Gu, Siddharth Gururani, Ethan He, Jiahui Huang, Jacob Huffman, Pooya Jannaty, Jingyi Jin, Seung Wook Kim, Gergely Klár, Grace Lam, Shiyi Lan, Laura Leal-Taixe, Anqi Li, Zhaoshuo Li, Chen-Hsuan Lin, Tsung-Yi Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Arsalan Mousavian, Seungjun Nah, Sriharsha Niverty, David Page, Despoina Paschalidou, Zeeshan Patel, Lindsey Pavao, Morteza Ramezanali, Fitsum Reda, Xiaowei Ren, Vasanth Rao Naik Sabavat, Ed Schmerling, Stella Shi, Bartosz Stefaniak, Shitao Tang, Lyne Tchapmi, Przemek Tredak, Wei-Cheng Tseng, Jibin Varghese, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Xinyue Wei, Jay Zhangjie Wu, Jiashu Xu, Wei Yang, Lin Yen-Chen, Xiaohui Zeng, Yu Zeng, Jing Zhang, Qinsheng Zhang, Yuxuan Zhang, Qingqing Zhao, and Artur Zolkowski. Cosmos world foundation model platform for physical ai, 2025. URL https://arxiv.org/abs/2501.03575.

NVIDIA, Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, Prithvijit Chattopadhyay, Mike Chen, Yongxin Chen, Yu Chen, Shuai Cheng, Yin Cui, Jenna Diamond, Yifan Ding, Jiaojiao Fan, Linxi Fan, Liang Feng, Francesco Ferroni, Sanja Fidler, Xiao Fu, Ruiyuan Gao, Yunhao Ge, Jinwei Gu, Aryaman Gupta, Siddharth Gururani, Imad El Hanafi, Ali Hassani, Zekun Hao, Jacob Huffman, Joel Jang, Pooya Jannaty, Jan Kautz, Grace Lam, Xuan Li, Zhaoshuo Li, Maosheng Liao, Chen-Hsuan Lin, Tsung-Yi Lin, Yen-Chen Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Yifan Lu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Seungjun Nah, Yashraj Narang, Abhijeet Panaskar, Lindsey Pavao, Trung Pham, Morteza Ramezanali, Fitsum Reda, Scott Reed, Xuanchi Ren, Haonan Shao, Yue Shen, Stella Shi, Shuran Song, Bartosz Stefaniak, Shangkun Sun, Shitao Tang, Sameena Tasmeen, Lyne Tchapmi, Wei-Cheng Tseng, Jibin Varghese, Andrew Z. Wang, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Jiashu Xu, Dinghao Yang, Xiaodong Yang, Haotian Ye, Seonghyeon Ye, Xiaohui Zeng, Jing Zhang, Qinsheng Zhang, Kaiwen Zheng, Andrew Zhu, and Yuke Zhu. World simulation with video foundation models for physical ai, 2025. URL https://arxiv.org/abs/2511.00062.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, Yihang Chen, Jie Liu, Yansong Cheng, Yao Yao, Jiayi Zhu, Yihao Meng, Kecheng Zheng, Qingyan Bai, Jingye Chen, Zehong Shen, Yue Yu, Xing Zhu, Yujun Shen, and Hao Ouyang. Advancing open-source world models, 2026. URL https://arxiv.org/abs/2601.20540.

Antoine Schnepf, Karim Kassab, Flavian Vasile, and Andrew Comport. Sphericaldreamer: Generating navigable immersive 3d worlds with panorama fusion, 2026. URL https://arxiv.org/abs/2605.19974.

Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4104–4113, 2016. doi: 10.1109/CVPR.2016.445.

Junyoung Seo, Hyunwook Choi, Minkyung Kwon, Jinhyeok Choi, Siyoon Jin, Gayoung Lee, Junho Kim, JoungBin Lee, Geonmo Gu, Dongyoon Han, Sangdoo Yun, Seungryong Kim, and Jin-Hwa Kim. Grounding world simulation models in a real-world metropolis, 2026. URL https://arxiv.org/abs/2603.15583.

Vincent Sitzmann, Semon Rezchikov, William T. Freeman, Joshua B. Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering, 2022. URL https://arxiv.org/abs/2106.02634.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023. URL https://arxiv.org/abs/2104.09864.

Junshu Tang, Jiacheng Liu, Jiaqi Li, Longhuang Wu, Haoyu Yang, Penghao Zhao, Siruis Gong, Xiang Yuan, Shuai Shao, Linfeng Zhang, and Qinglin Lu. Hunyuan-gamecraft-2: Instruction-following interactive game world model, 2026. URL https://arxiv.org/abs/2511.23429.

Team HY-World, Chenjie Cao, Xuhui Zuo, Zhenwei Wang, Yisu Zhang, Junta Wu, Zhenyang Liu, Yuning Gong, Yang Liu, Bo Yuan, Chao Zhang, Coopers Li, Dongyuan Guo, Fan Yang, Haiyu Zhang, Hang Cao, Jianchen Zhu, Jiaxin Lin, Jie Xiao, Jihong Zhang, Junlin Yu, Lei Wang, Lifu Wang, Lilin Wang, Linus, Minghui Chen, Peng He, Penghao Zhao, Qi Chen, Rui Chen, Rui Shao, Sicong Liu, Wangchen Qin, Xiaochuan Niu, Xiang Yuan, Yi Sun, Yifei Tang, Yifu Sun, Yihang Lian, Yonghao Tan, Yuhong Liu, Yuyang Yin, Zhiyuan Min, Tengfei Wang, and Chunchao Guo. Hy-world 2.0: A multi-modal world model for reconstructing, generating, and simulating 3d worlds, 2026. URL https://arxiv.org/abs/2604.14268.

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, Wente 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, 2025. URL https://arxiv.org/abs/2503.20314.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric challenges, 2019. URL https://arxiv.org/abs/1812.01717.

Alexander Veicht, Paul-Edouard Sarlin, Philipp Lindenberger, and Marc Pollefeys. Geocalib: Learning single-image calibration with geometric optimization, 2024. URL https://arxiv.org/abs/2409.06704.

Hanlin Wang, Hao Ouyang, Qiuyu Wang, Wen Wang, Qingyan Bai, Ka Leong Cheng, Yue Yu, Yixuan Li, Yihao Meng, Zichen Liu, et al. Worlddirector: Building controllable world simulators with persistent dynamic memory. *arXiv preprint arXiv:2607.02517*, 2026a.

Jiahao Wang, Luoxin Ye, TaiMing Lu, Junfei Xiao, Jiahan Zhang, Yuxiang Guo, Xijun Liu, Rama Chellappa, Cheng Peng, Alan Yuille, and Jieneng Chen. Evoworld: Evolving panoramic world generation with explicit 3d memory, 2025. URL https://arxiv.org/abs/2510.01183.

Zile Wang, Zexiang Liu, Jiaxing Li, Kaichen Huang, Baixin Xu, Fei Kang, Mengyin An, Peiyu Wang, Biao Jiang, Yichen Wei, Yidan Xietian, Jiangbo Pei, Liang Hu, Boyi Jiang, Hua Xue, Zidong Wang, Haofeng Sun, Wei Li, Wanli Ouyang, Xianglong He, Yang Liu, Yangguang Li, and Yahui Zhou. Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory, 2026b. URL https://arxiv.org/abs/2604.08995.

Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video world models with long-term spatial memory, 2025. URL https://arxiv.org/abs/2506.05284.

Ziyi Wu, Daniel Watson, Andrea Tagliasacchi, David J. Fleet, Marcus A. Brubaker, and Saurabh Saxena. 360anything: Geometry-free lifting of images and videos to 360deg, 2026. URL.

Yifei Xia, Shuchen Weng, Siqi Yang, Jingqi Liu, Chengxuan Zhu, Minggui Teng, Zijian Jia, Han Jiang, and Boxin Shi. Panowan: Lifting diffusion video generation models to 360° with latitude/longitude-aware mechanisms, 2025. URL https://arxiv.org/abs/2505.22016.

Tianshuo Xu, Yichen Xie, Depu Meng, Chensheng Peng, Quentin Herau, Bo Jiang, Yihan Hu, and Wei Zhan. Teaching video generators to remember: Eliciting dynamic memory for out-of-sight state evolution. *arXiv preprint arXiv:2605.25333*, 2026.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. Cogvideox: Text-to-video diffusion models with an expert transformer, 2025. URL https://arxiv.org/abs/2408.06072.

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, 2025a. URL https://arxiv.org/abs/2412.07772.

Yuyang Yin, HaoXiang Guo, Fangfu Liu, Mengyu Wang, Hanwen Liang, Eric Li, Yikai Wang, Xiaojie Jin, Yao Zhao, and Yunchao Wei. Panoworld-x: Generating explorable panoramic worlds via sphere-aware video diffusion, 2025b. URL https://arxiv.org/abs/2509.24997.

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, 2025. URL https://arxiv.org/abs/2506.03141.

Wei Yu, Runjia Qian, Yumeng Li, Liquan Wang, Songheng Yin, Sri Siddarth Chakaravarthy P, Dennis Anthony, Yang Ye, Yidi Li, Weiwei Wan, and Animesh Garg. Mosaicmem: Hybrid spatial memory for controllable video world models, 2026. URL https://arxiv.org/abs/2603.17117.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018. URL https://arxiv.org/abs/1801.03924.

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, 2024. URL https://arxiv.org/abs/2412.20404.

Yixuan Zhu, Jiaqi Feng, Wenzhao Zheng, Yuan Gao, Xin Tao, Pengfei Wan, Jie Zhou, and Jiwen Lu. Astra: General interactive world model with autoregressive denoising, 2026. URL https://arxiv.org/abs/2512.08931.

Hakan Çapuk, Andrew Bond, Muhammed Burak Kızıl, Emir Göçen, Erkut Erdem, and Aykut Erdem. Tandit: Tangent-plane diffusion transformer for high-quality 360° panorama generation, 2025. URL https://arxiv.org/abs/2506.21681.

APPENDIX

A Related Work

A.1 Video World Models

A.2 Panorama World Models

A.3 Out-of-State and Dynamics

B Implementation Details

B.1 Model and Training Details

B.2 Real Data Processing

B.3 Synthetic Data Processing

C Evaluation Details

C.1 Benchmark Construction

C.2 Evaluation Metrics

C.3 Out-of-View Dynamics Metrics

C.4 Comparison with existing Out-of-View Metrics

C.5 Baseline Adaptation

D Additional Results and Analyses

D.1 Additional Qualitative Results

D.2 Additional Qualitative Ablation Results

E Discussions

E.1 Limitations

E.2 Societal Impact

A Related Work

A.1 Video World Models

Building on recent advances in video generation (Team Wan et al., 2025; Zheng et al., 2024; Kong et al., 2025; Yang et al., 2025; Yin et al., 2025a; Huang et al., 2025), video world models adopt generative models to predict future observations from an actor-centric view. Some video world models (NVIDIA et al., 2025; Robbyant Team et al., 2026) rely primarily on the generative prior learned from large-scale pretraining. To maintain a coherent world beyond the actor currently observes, recent approaches additionally retrieve relevant past observations (Team HY-World et al., 2026; Yu et al., 2025; DreamX Team et al., 2026; Wang et al., 2026b) or maintain stored states in explicit spatial and geometric representations (Wu et al., 2025; Li et al., 2025b; Yu et al., 2026; Li et al., 2026a). However, generative priors, retrieved observations, and stored states still lack direct evidence of how dynamic content evolves once it leaves the actor view, leading to the *frame-locked*, *lost*, *frozen*, and *impostor* failures shown in Fig. 2. In contrast, we jointly generate a panoramic *observer* that continues to represent the evolving world beyond the actor's view, so that out-of-view dynamic content is explicitly generated.

A.2 Panorama World Models

To overcome the limited FoV of perspective models, prior work has explored panoramas either as intermediate representations for constructing coherent 3D worlds (Wang et al., 2025; Schnepf et al., 2026; Team HY-World et al., 2026) or as direct world representations through panoramic video generation (Yin et al., 2025b; Li et al., 2026b; Chen et al., 2026c; Liu et al., 2026). Producing panoramas as the final output introduces additional challenges: pretrained backbones are typically optimized for perspective imagery, which has motivated tangent-plane or cubemap-based formulations (Kalischek et al., 2025; Çapuk et al., 2025), while representing the full sphere under a fixed token budget can reduce the effective resolution available to each viewing direction (Li et al., 2026c; Liu et al., 2025). In contrast, we decouple the actor from the observer within an actor-observer framework and assign the observer the role of watching the region we want, while the perspective actor is generated jointly as the primary output. Because the observer maintains the surrounding world rather than serving as the actor view, it need not carry the actor's fidelity, which sidesteps these resolution and prior limitations while the perspective actor stays high-quality.

A.3 Out-of-View State and Dynamics

Recent benchmarks examine whether world models maintain evolving states when parts of the world become temporarily unobserved. STEVO-Bench (Ma et al., 2026) tests state and dynamics consistency under controlled observation interruptions. WRBench (Lu et al., 2026) evaluates state consistency when the camera moves away from a scene and later returns. MemoBench (Chen et al., 2026a) evaluates whether out-of-view objects re-enter with appropriately evolved states across diverse dynamic scenes, including physical state changes. These benchmarks, however, leave the judgment to a VLM, which rates whether a reappearance looks natural. In the dynamic, multi-object scenes world models produce, naturalness depends on where an object actually moved while unobserved, which such judgments cannot verify. A few approaches address interrupted observation directly. HyDRA (Chen et al., 2026b) and ReMind (Xu et al., 2026) use dynamic memory and generative priors to continue unseen dynamics, while LiveWorld (Duan et al., 2026) and WorldDirector (Wang et al., 2026a) explicitly advance hidden states through simulation or motion planning. Yet generative approaches may drift toward plausible but incorrect unseen states, memory-based methods remain limited by previously observed evidence, and explicit state advancement requires structured entity-level modeling. We instead use a panoramic video as an evolving global observer that continuously captures out-of-view dynamics.

B Implementation Details

B.1 Model and Training Details

B.1.1 Additional training details

World Observer is trained with teacher forcing (TF) throughout. During TF training, we inject small Gaussian noise (scale 0.1) to the conditioning history frames with 50% probability, reducing the gap between clean training inputs and self-generated inference history. Each chunk conditions on $H{=}2$ history latents, corresponding to 5 frames of the preceding chunk. The two training data sources are mixed via ratio-based interleaved sampling: real panoramic videos (Luo et al., 2026; Xia et al., 2025; Chen et al., 2024) (60%) and synthetic video from CARLA (Dosovitskiy et al., 2017) (40%). The multi-observer stage is trained on synthetic data only, as synchronized multi-observer recordings are difficult to obtain in the real world. All camera extrinsics are converted to a unified Right-Down-Forward (RDF) coordinate system and expressed relative to the first frame of each chunk.

The warping source is not restricted to the first frame of the training clip. We sample it from the full source video surrounding the clip, taking the nearest past or future frame beyond a minimum spatial distance from the target, so the condition is rarely a clean, well-aligned view of the target. This matches the autoregressive setting, where later chunks are warped from frames that are increasingly distant from the current viewpoint.

To enable classifier-free guidance (CFG) and ensure graceful degradation when certain conditions are unavailable, we apply the following dropout schedule during training: text captions are replaced with empty-string T5 embeddings with a probability of 20%, and Observer Sink tokens are zeroed out with a probability of 20%. At inference we apply the same guidance scale to the actor and observer streams.

B.1.2 Additional model details

World Observer fine-tunes Cosmos-Predict2.5 (NVIDIA et al., 2025), a 2B-parameter Diffusion Transformer (DiT) with 28 blocks, 16 attention heads, and a hidden dimension of 2048. The model operates in a 16-channel latent space produced by a 3D VAE with $4\times$ temporal and $8\times$ spatial compression, so a chunk of $T{=}77{=}1{+}4\times19$ frames yields $L{=}20$ latents per stream. We render the actor at $R_{\mathrm{a}}{=}1280\times704$ and the observer at $R_{\mathrm{o}}{=}640\times320$, so each observer contributes roughly a quarter of the actor's tokens; despite its wider 360° coverage, the observer is kept at low resolution for a compact token budget and faster training and inference, which also keeps the joint sequence tractable as observers are added.

The DiT input is assembled along the channel axis before patch embedding, concatenating the noisy target latent, the encoded warped video, a warp validity mask, a learnable view embedding, and any extra warp streams, in this order. The view embedding is a learnable 6-channel vector with one row for perspective and one for panoramic tokens, which distinguishes the two streams. To separate multiple observers, which share the panoramic row and would otherwise be indistinguishable, we add $n\,\mathbf{e}_{\mathrm{obs}}$ to the n-th observer of the same type, where n is a 0-based index and $\mathbf{e}_{\mathrm{obs}}$ is a learnable vector. Since the first observer takes $n{=}0$, the multi-observer model reduces exactly to the single-observer one, so the single-observer checkpoint warm-starts multi-observer training without any change to the embedding table.

Camera poses are encoded separately from these channels. We build a 6-channel Plücker ray map for each view at the latent resolution, equirectangular rays for the observer and pinhole rays for the actor, embed it with a patch embedder, and *add* it to the video tokens after patch embedding. This embedder is zero-initialized so that generation starts identically to the base model at the first step, keeping the warm-start clean.

The Observer Sink crops are taken at the actor resolution R_{a} and add four tokens in total, independent of the number of observers. They carry RoPE (Su et al., 2023) positions outside the current generation window, at the offset $\Delta_{\mathrm{os}}{=}50$ of eq. 4, with the four cardinal crops spaced by a stride of 5. Because attention within a chunk is bidirectional, they are visible to all target tokens regardless of their placement in the sequence, and no reordering is required.

B.2 Real Data Processing

B.2.1 Panorama Video Processing

360° video canonicalization. Raw 360° footage is unstable, as the rig rotates between frames with an arbitrary vertical axis, and warping it directly would entangle camera rotation with the scene geometry we want to recover. We therefore canonicalize every clip into an upright, gravity-aligned ERP sequence, following 360Anything (Wu et al., 2026). We render each ERP frame to a cubemap and recover per-frame rotation with COLMAP (Schönberger & Frahm, 2016) to cancel inter-frame spin, then estimate the gravity direction with GeoCalib (Veicht et al., 2024) and apply a rotation that aligns it with the ERP down axis.

Panoramic depth estimation. Building the warping condition requires a high-resolution depth map over the full panorama. Since monocular depth estimators are trained on perspective images and cannot be applied directly to the distorted ERP panorama, we decompose the sphere into perspective views, predict depth at high resolution on each, and merge them back onto the sphere. For each depth keyframe we render 110-FOV perspective views along the vertices of an icosahedron, whose overlapping frusta cover every direction with at least two views. We pass these views to Depth Anything 3 (Lin et al., 2025) in a single batch without external extrinsics, so the model observes all views jointly, resolves their relative poses internally, and returns depth in one common scale, removing the per-view scale alignment and the drift it introduces. The per-view depths are fused into one ERP depth map by weighted averaging, and camera poses are defined from the front-view extrinsics and completed over the full trajectory by SLERP and linear interpolation. The resulting depth and poses are used for raymap conditioning and warping at training time.

B.2.2 Text captioning

We generate text captions for all training clips using the Qwen3.6-27B vision-language model. Following the 4-view panoramic captioning strategy of PanoWorld (Jiang et al., 2026), we represent each panoramic frame using four 90°-FOV perspective crops corresponding to the Front, Right, Back, and Left directions, rather than directly captioning the raw equirectangular (ERP) frame. We adopt this representation to avoid directly exposing the vision-language model to the discontinuity at the ERP boundary and the strong projection distortion near the poles.

For each clip, we generate one panorama caption and one perspective caption for each camera trajectory. Each caption describes the environment, dynamic actors, and specific events in the scene. By comparing observations across multiple timestamps for each viewing direction, the caption records objects that enter, leave, or change over time, including events occurring outside the current field of view of the perspective cameras. Because these out-of-view events are recorded directly in the caption, it carries information about how the scene's dynamic state evolves outside a camera's current field of view. The panorama caption summarizes these dynamics over the full panoramic view, while each perspective caption focuses on the content visible along its corresponding camera trajectory. The panorama caption is further required to describe scene dynamics without referring to screen position or compass direction. Camera behavior is described directly in each perspective caption's prose, reflecting the sampled yaw, pitch, and field of view of the corresponding trajectory. We apply the same captioning procedure to the synthetic data.

B.2.3 Trajectory Construction

For each clip, we render six perspective-camera trajectories from the same temporal segment of a panoramic video. Each trajectory independently samples its field of view and pitch, which remain fixed throughout the clip, while the roll is fixed at 0°. The six trajectories consist of two static and four moving views. The ``origin'' trajectory remains fixed at the canonical forward direction with zero yaw and pitch, while `front_static` remains static at its sampled reference direction. The `yaw_right` and `yaw_left` trajectories continuously rotate to the right and left, respectively, with a randomly sampled rotation angle and varying rotation speed over time. The `out_back_right_first` and `out_back_left_first` trajectories first rotate in the corresponding direction and then turn back, with the outward and return angles sampled independently and the turning point randomly selected within the clip. For all trajectories except `origin`, the reference yaw is independently chosen to be either aligned with the canonical forward direction

or randomly sampled. This construction provides diverse starting viewpoints and camera motions while keeping all six perspective views temporally aligned with the source panorama.

B.3 Synthetic Data Processing

We use CARLA Dosovitskiy et al. (2017) to construct our synthetic video dataset, drawing on both the Unreal Engine 4 build (v0.9.15), from which we use `Town01`, `Town02`, `Town03`, `Town05`, and `Town10`, and the Unreal Engine 5 build (v0.10.0), from which we additionally use `Town10`. We randomly spawn vehicles and pedestrians before rendering to simulate realistic traffic, and render each clip under an independently sampled weather and lighting condition, combining clear and overcast presets with noon or sunset time of day and jittering cloudiness, sun altitude, and fog, so that appearance, illumination, and traffic all vary across the dataset. We additionally render metric depth maps and extract camera parameters per frame to support geometric referencing.

B.3.1 Panoramic actor-observer capture

CARLA does not natively expose an equirectangular (ERP) camera, so we assemble each panorama from a cubemap rig of six pinhole RGB-D cameras (90° FOV, 960×960 each) captured in lock-step within a single simulation tick and stitched into a $360^\circ \times 180^\circ$ ERP panorama and depth map. Each clip contains one moving actor and two static observers, all captured with such rigs at an eye height of 1.6–$1.8\,\mathrm{m}$ and sharing a common orientation. The actor walks at $1.3\,\mathrm{m/s}$ along a walkable grid derived from each map's OpenDRIVE sidewalk graph, and the two observers are fixed at its first-frame and last-frame poses. Because all three rigs advance under the same simulation clock, they witness the same vehicles and pedestrians at the same instant, giving time-aligned observations of one evolving world that real footage cannot provide. For each observer we store its $t{=}0$ panorama and depth as a forward-warp source.

C Evaluation Details

C.1 Benchmark Construction

The benchmark videos are built so that every clip poses the same well-defined challenge. The camera moves away from and then back toward its starting heading, causing one or more dynamic objects to leave the field of view and later re-enter it. Clips are drawn from the same real and synthetic (CARLA) sources used for training. We first restrict candidate scenes to those containing fewer than 10 dynamic objects in total, so that individual objects stay visually uncluttered and unambiguous to track and describe. Within each such scene, we crop a perspective view whose camera follows a controlled "out-and-back" pan specifically to force objects out of frame and back rather than relying on incidental motion. We then manually curate, per scene, 1–3 objects (human, vehicle, or animal) that unambiguously exit through a frame edge and later re-enter it, discarding any candidate whose exit or re-entry is ambiguous or does not occur. Each selected object is then given a caption of its appearance and behavior.

Across the 200 clips, most contain a single annotated object. 77 of the 100 real clips and 73 of the 100 synthetic clips have exactly one, with the remainder split between two objects (20 real, 26 synthetic) and, rarely, three (3 real, 1 synthetic). An average of 1.26 objects per clip on the real split and 1.28 on the synthetic split. The two splits differ markedly in which kind of object that tends to be. The real clips are dominated by humans (104 of 126 objects, appearing in 81 of the 100 clips), with vehicles and animals appearing only occasionally (17 and 5 objects). The synthetic clips are dominated by vehicles instead (81 of 128 objects, in 64 of the 100 clips), and humans appear in fewer than half the clips (47 objects, 44 clips).

C.2 Evaluation Metrics

We evaluate generation quality along four aspects: visual and temporal fidelity with FID Heusel et al. (2018) and FVD Unterthiner et al. (2019), image quality with VBench Huang et al. (2023), camera-following accuracy with RotErr and TransErr, and 3D adherence with masked PSNR and LPIPS Zhang et al. (2018). Both Real-OOV-Bench and Synthetic-OOV-Bench provide ground-truth videos, and all metrics are reported separately on the two benchmarks. For panorama world models,

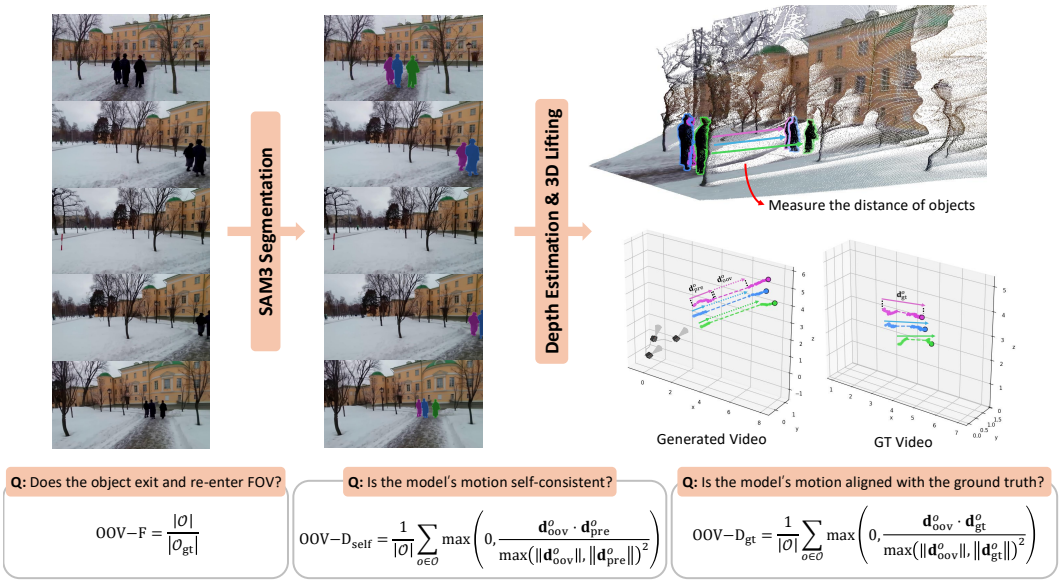


Figure 8: **Out-of-view dynamics evaluation protocol.** We track dynamic objects in 3D to evaluate their exit and re-entry (OOV-F) and motion consistency during out-of-view intervals (OOV-D_{self} and OOV-D_{gt}).

whose output is a full panorama, we crop a perspective view using the ground-truth camera rotation and compare it against the ground-truth perspective, matching our actor output. For the reference-based metrics, we resize each model's output and the ground truth to a common resolution so they are compared under the same spatial scale. Below we detail how each metric is computed.

3D adherence. PSNR and LPIPS are computed between the generated video and the ground-truth video. Because our model targets dynamic video generation grounded in real-world references rather than exact 4D reconstruction, dynamic objects in the generated video need not match those in the ground truth. To focus these metrics on how faithfully the model preserves static scene structure, we segment dynamic objects in both generated and ground-truth frames using SAM3 Carion et al. (2026) with text prompts for dynamic categories (pedestrian and vehicle), and compute PSNR and LPIPS only over the static regions.

Camera-following accuracy. We extract per-frame camera extrinsics from the generated frames using Depth Anything 3 Lin et al. (2025), and take the ground-truth extrinsics from the same estimator applied to the ground-truth frames. Running the same estimator on both sides removes any bias between predicted and reference poses; for the synthetic benchmark we likewise use the DA3 estimates rather than the simulator poses, so that both benchmarks are evaluated identically. We process frames in non-overlapping chunks and compute relative poses by setting the first frame as the identity. Each translation trajectory is independently normalized by its maximum translation norm to remove scale ambiguity. RotErr is the mean geodesic distance on $\mathrm{SO}(3)$ between the predicted and ground-truth relative rotations, and TransErr is the mean ℓ_2 distance between the scale-normalized relative translations, following Li et al. (2025b).

C.3 Out-of-View Dynamics Metrics

Protocol. Let $\mathcal{O}_{\mathrm{gt}}$ be the set of objects that exit and re-enter the FoV in the ground-truth video, and $\mathcal{O} \subseteq \mathcal{O}_{\mathrm{gt}}$ the subset that also does so in the generated video. For each $o \in \mathcal{O}$, we localize the object with its SAM3 mask and take the median depth within the mask, which we back-project to a single 3D position, using the median to reduce sensitivity to segmentation and depth noise. The out-of-view displacement $\mathbf{d}^o_{\mathrm{oov}}$ is then the difference between these 3D positions at the last frame before exit and the first frame after re-entry, as illustrated in Fig. 8. Displacement is undefined outside $\mathcal{O}$, since it requires an object to actually leave and return. All methods use the same segmentation and depth pipeline.

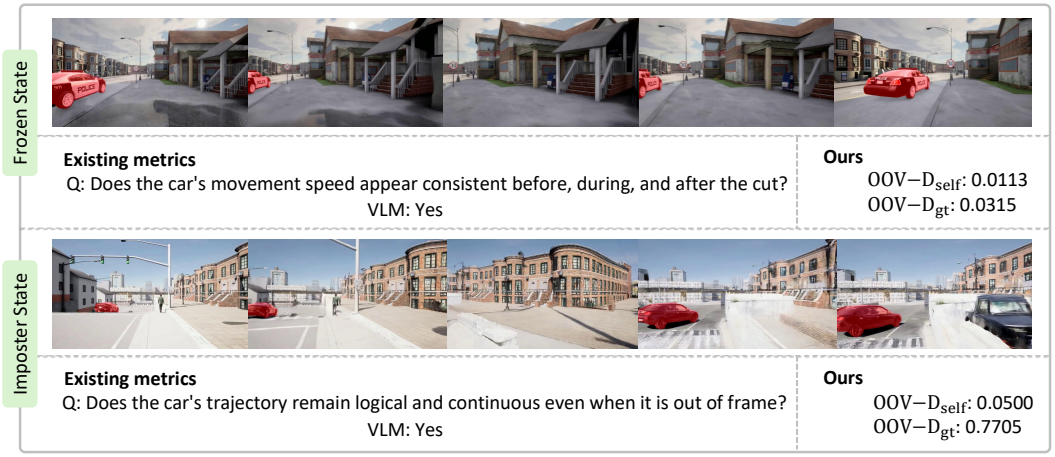


Figure 9: **Comparison with existing Out-Of-View metrics.** VLM-based metrics fail to identify frozen and impostor states as inconsistent, while our 3D displacement-based metrics quantitatively capture these failures.

OOV-F. An object may fail to leave the frame as the camera moves away, or fail to return afterward, corresponding to the frame-locked and lost states, neither of which yields a well-defined displacement. We therefore report

$$\text{OOV-F} = \frac{|\mathcal{O}|}{|\mathcal{O}_{\text{gt}}|}, \tag{5}$$

the fraction of objects that follow the ground-truth exit-and-re-entry pattern. It is reported alongside OOV-D, since the latter is conditioned on $\mathcal{O}$ and a model that drops or locks objects is otherwise evaluated on a smaller, easier subset.

OOV-D. We score each object's out-of-view motion by its agreement with a reference and average over $\mathcal{O}$, using the ground-truth trajectory $\mathbf{d}^o_{\text{gt}}$ (scaled to the generated out-of-view interval) and the object's own pre-exit motion $\mathbf{d}^o_{\text{pre}}$ as the two references,

$$\begin{aligned} \text{OOV-D}_{\text{gt}} &= \frac{1}{|\mathcal{O}|}\sum_{o\in\mathcal{O}} \max\left(0, \frac{\mathbf{d}^o_{\text{oov}}\cdot\mathbf{d}^o_{\text{gt}}}{\max(\|\mathbf{d}^o_{\text{oov}}\|,\|\mathbf{d}^o_{\text{gt}}\|)^2}\right), \\ \text{OOV-D}_{\text{self}} &= \frac{1}{|\mathcal{O}|}\sum_{o\in\mathcal{O}} \max\left(0, \frac{\mathbf{d}^o_{\text{oov}}\cdot\mathbf{d}^o_{\text{pre}}}{\max(\|\mathbf{d}^o_{\text{oov}}\|,\|\mathbf{d}^o_{\text{pre}}\|)^2}\right). \end{aligned} \tag{6}$$

Normalizing by the larger magnitude bounds each term in $[0, 1]$ after clipping, reaching 1 only when direction and magnitude match and 0 for wrong-direction motion. The two references emphasize different failures. OOV-D$_{\text{gt}}$ penalizes any motion that departs from the ground truth, whether frozen, under- or over-shooting, or moving in the wrong direction. OOV-D$_{\text{self}}$ instead penalizes motion that breaks from the object's own prior trajectory, which surfaces impostor dynamics that look plausible on their own but are inconsistent with how the object was moving.

C.4 Comparison with Existing Out-of-View Metrics

Existing out-of-view benchmarks, such as MemoBench (Chen et al., 2026a) and WRBench (Lu et al., 2026), use VLM judges to decide whether an object stays consistent while out of view. The core difficulty is that both the camera and the objects move, so an object's true motion cannot be read off the rendered frames directly. Judging whether a re-appearing object has evolved naturally requires recovering how it actually moved in 3D, which VLMs estimate poorly from frames alone. As a result, once an object re-appears, VLMs often cannot tell natural evolution from a frozen or impostor state, and misjudge exactly the cases these benchmarks target.

Fig. 9 shows two such cases from our evaluation. In the *frozen* case, asked "Is the object's motion consistent before, during, and after the interval?", the VLM answers "Yes," although the object has actually stopped. In the *impostor* case, asked "Does the car's trajectory remain logical and continuous even when it is out of frame?", it again answers "Yes," although the reappearing motion no

longer follows the object's own prior trajectory. We follow the question format of these benchmarks, yet both judgments are wrong.

Our metric instead measures 3D displacement explicitly. We lift each object with a segmentation and depth model, so we can quantify how much it actually moved out of view. OOV-D_{gt} exposes the frozen case, since the observed displacement is far smaller than the ground-truth reference, and OOV-D_{self} exposes the impostor case, since the out-of-view motion departs from the object's own pre-exit trajectory. Both failures the VLM judge misses are thus identified quantitatively.

C.5 Baseline Adaptation

Existing world models adopt different camera-conditioning interfaces, including continuous camera poses, Plücker ray embeddings, and discretized action controls. We therefore convert each benchmark camera trajectory into the representation expected by the corresponding model while preserving the underlying motion.

OmniRoam (Liu et al., 2026). OmniRoam uses a reference-space camera trajectory defined at the latent temporal resolution. Since its released panoramic conditioning path models camera translation without explicit rotational control, we re-base the benchmark trajectory to the first frame, retain only the translational component, and map it to OmniRoam's reference coordinate system.

PanoWorld (Li et al., 2026b). PanoWorld conditions its Action-Model on relative camera translation and yaw. We compute camera centers from the benchmark extrinsics, express their displacement in the first-camera coordinate system, and map the resulting motion to PanoWorld's panoramic coordinate convention. Relative yaw is measured with respect to the initial camera orientation, while translation is preserved at its original metric scale without additional normalization.

HyDRA (Chen et al., 2026b). HyDRA is evaluated using its native out-of-view reconstruction protocol rather than the shared start-frame generation protocol. For each sequence, we identify the trajectory apex and split the ground-truth video into a context segment before the apex and a target segment after it. Camera poses are then expressed relative to the first target frame and sampled at the model's latent temporal resolution. Since the model generates a fixed-length sequence, we evaluate only the frames corresponding to the valid target interval.

HY-World 1.5 (HunyuanWorld, 2025). HY-World 1.5 conditions generation on both continuous camera poses through PRoPE and discretized translation and rotation actions. We subsample the benchmark trajectory to the model's latent temporal resolution, express the poses relative to the first frame, and derive the corresponding discrete motion labels from the same relative trajectory. Pose normalization follows the convention used during training.

Dream-X (DreamX Team et al., 2026). Dream-X conditions generation on camera poses through Projective Rotary Position Embedding (PRoPE). We directly convert the benchmark world-to-camera trajectories into the model's RealEstate10K-style pose representation, express them relative to the first frame, and preserve the original metric translation scale.

FantasyWorld (Dai et al., 2025). FantasyWorld represents camera motion using per-pixel 6-channel Plücker ray embeddings. We rescale the benchmark intrinsics to the generation resolution and construct the camera conditioning using the model's official Plücker embedding implementation. Following the released inference procedure, camera translations are normalized using a scene scale estimated from monocular depth of the initial frame.

LingBot-World (Robbyant Team et al., 2026). LingBot-World conditions each transformer block on 6-channel Plücker ray maps. We convert the benchmark world-to-camera poses to camera-to-world poses and provide intrinsics that preserve the benchmark field of view in the model's reference coordinate system. Camera translation magnitude is preserved, while trajectory normalization is handled by the model's native inference pipeline.

Matrix-GAME 3.0 (Wang et al., 2026b). Matrix-GAME 3.0 conditions generation on interactive camera controls. We convert each benchmark camera trajectory into the corresponding control sequence by decomposing the relative camera motion into translational and rotational components and mapping them to the model's supported controls. The resulting sequence is applied throughout generation using the model's native inference pipeline, while preserving the direction and temporal progression of the benchmark trajectory.



Figure 10: **Additional qualitative comparisons with other models.** We provide additional comparisons of World Observer with existing world models on real and synthetic scenes involving out-of-view dynamics. For synthetic scenes, we additionally include results from World Observer (Multi). Red boxes highlight the target objects before exit and after re-entry. Across diverse objects, motions, and scenes, existing models exhibit out-of-view failures such as lost, frozen, and impostor states, whereas World Observer consistently preserves object identity and state evolution throughout the out-of-view interval.

D Additional Results and Analyses

D.1 Additional Qualitative Results

Fig. 10 provides additional comparisons on both synthetic and real-world scenes with diverse out-of-view motions. Existing models often fail to preserve object identity or motion once the target leaves the actor's view, leading to inconsistent re-entry. In contrast, World Observer maintains the target state throughout the unseen interval in both single- and multi-observer settings. The multi-observer variant further demonstrates consistent generation as the actor moves across regions covered by different observers. Additional qualitative results are provided in **the supplementary video materials**.

D.2 Additional Qualitative Ablation Results

Fig. 11 presents qualitative ablation results demonstrating the complementary roles of the actor and observer streams in preserving out-of-view dynamics and generation fidelity. We additionally examine how these ablation effects become more pronounced over autoregressive rollouts.

Effect of joint generation. The state-memory role of joint generation is most pronounced under autoregressive rollout. Without joint generation, the only information available to the actor is the Observer Sink and the recent actor history, so the model cannot maintain how the state evolves outside the field of view, and all memory is ultimately confined to the actor view. Joint generation instead lets the actor generate the dynamics within its view while tracking out-of-view dynamics through the observer. Retrieving past frames during rollout, as prior models do, only recalls observations from specific past timesteps. The model must then generate dynamics without knowing how the state progressed out of view between those observations and the present. Consistent with this, removing joint generation fails to keep out-of-view motion self-consistent and markedly lowers OOV-D_{self} (Tab. 2). In contrast, from the history panorama alone, our model goes beyond static



Figure 11: **Qualitative Ablation Results.** We show the effect of removing each component on generation quality. Without the observer, the model fails to maintain out-of-view dynamics, causing the dynamic object to disappear when it re-enters the view. Without the actor, panorama-based generation followed by cropping leads to reduced visual fidelity. World Observer preserves evolving out-of-view dynamics while maintaining high-fidelity generation.

memory and continuously evolves the out-of-view dynamics that retrieval-based approaches cannot carry forward (Fig. 12 (b), (c)).

Effect of the Observer Sink. The effect of the Observer Sink becomes more visible as the camera moves more aggressively. To show this, we run inference on a trajectory that moves forward while continuously rotating to the right, sweeping a full 360°. Without the Observer Sink, generation proceeds from only the appearance cues in the first chunk's frontal condition frame, so fine appearance quality degrades as the rotation grows and errors accumulate over the rollout. The Observer Sink supplies four high-resolution cardinal references from the initial panorama, so that regions undergoing such large viewpoint changes are still rendered faithfully when they re-enter the actor view (Fig. 12 (d)).

Effect of decoupled Actor-Observer prompting. The world perceived by the actor and the world perceived by the observer are inevitably different. Prior world models can only be controlled through the actor prompt, so a prompt describing something outside the actor's view is forced to hallucinate within the region the actor currently sees. We instead decouple the observer prompt so that control acts on the broader world the observer covers. As shown in Fig. 13 (a), for an event that can only be expressed with knowledge of the full surrounding scene, placing it in the actor prompt hallucinates it inside the actor's view, whereas placing it in the observer prompt lets the observer interpret and generate the event across the world, so that the actor later observes it when it comes into view. Also, as shown in Fig. 13 (b), applying the same unified prompt to both streams can hallucinate duplicate instances of the subject. The observer generates an additional Santa while another instance is already seated in the armchair. Decoupled prompting avoids such duplication by assigning each stream a prompt consistent with what it should observe.



Figure 12: **Qualitative Ablation Results.** We compare ground truth with variants without the observer and Observer Sink under a long camera trajectory with a full 360° sweep. Without the observer, out-of-view object dynamics are not maintained, while removing the Observer Sink degrades fine appearance under large viewpoint changes. Our full model preserves both dynamic objects (red/yellow) and appearance details (green) upon re-entry.

E Discussions

E.1 Limitations

World Observer currently assumes panoramic observations as conditioning inputs, which restricts the formulation to settings where such views are available. This requirement could be relaxed by first outpainting a perspective observation into a panorama, allowing the same framework to operate when only perspective inputs are given. The same approach can be applied to multi-observer settings, where multiple panoramic conditions could be constructed from perspective observations at different locations.

Because the observer budget is limited, observing the entire world remains difficult. Nevertheless, flexible observer placement allows this limited budget to be allocated to regions of interest, preserving their dynamics independently of where the actor moves.

E.2 Societal impact

Decoupling observing from acting enables world models to maintain the evolving state of selected regions independently of where the actor moves or looks. This capability can benefit embodied navigation, interactive simulation, and planning, where relevant parts of the world may need to remain continuously represented even when they are not currently observed by the acting agent. Flexible and multi-observer configurations further allow a limited observation budget to be allocated to re-



Figure 13: **Qualitative Prompt Ablation Results.** We compare (a) actor-concentrated, (b) unified, and (c) decoupled actor-observer prompting. Actor-concentrated prompting misplaces off-view events in the actor view, while unified prompting can hallucinate duplicate subjects in the observer. Decoupled prompting instead produces a consistent event across both streams.

gions of interest, supporting more persistent and efficient world-state maintenance without requiring the actor itself to revisit those regions.