OUROWORLD: BRINGING ANY 3D WORLD ALIVE AS DIVERSE, ENDLESSLY LOOPING 3D CINEMAGRAPHS

Anonymous authors
Paper under double-blind review

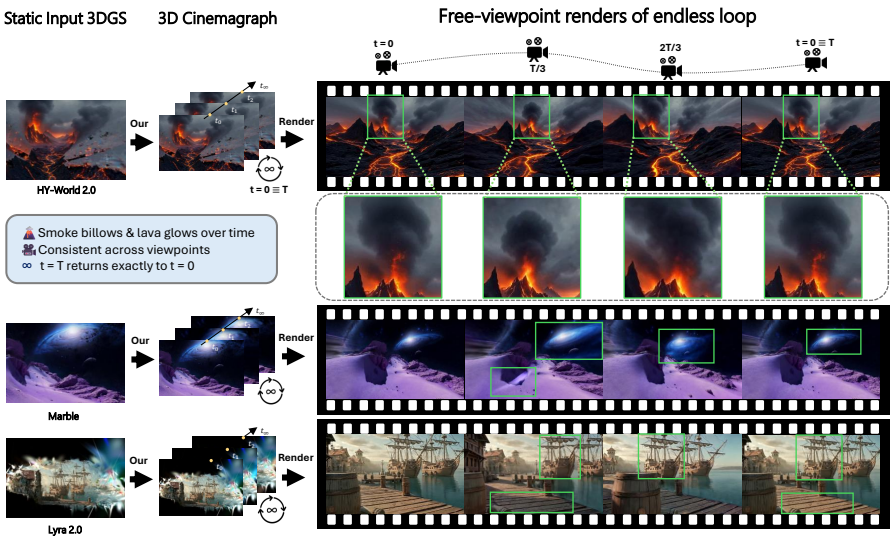


Figure 1: **OuroWorld brings static 3D worlds alive.** *Left*: Given a static 3DGS scene from any source, e.g., the 3D world models HY-World 2.0, Marble, and Lyra 2.0, OuroWorld converts it into a 3D cinemagraph: a dynamic scene with vivid, diverse motion that loops endlessly. *Right*: free-viewpoint renders along a moving camera, with **green boxes** highlighting regions with noticeable dynamics. Smoke billows in the volcano (top), the galaxy scatters stardust (middle), and the ships bob up and down on the water (bottom). Motion stays consistent across viewpoints, and the scene at $t{=}T$ returns to $t{=}0$, so the loop is seamless. See the supplementary videos for full results.

ABSTRACT

Recent 3D world models generate photorealistic, explorable scenes that remain frozen in time. OuroWorld is a mask-free framework that turns any static 3D Gaussian Splatting scene into a 3D cinemagraph: a dynamic scene with vivid, diverse motion looping seamlessly from any viewpoint. A vision-language model infers plausible dynamics and guides a video model to synthesize a reference video, which we lift and complete into multi-view videos. To learn from this imperfect supervision, we propose Inconsistency-Robust Periodic 4DGS: a Fourier-series deformation field guarantees looping by construction, while a Grounded Drift Field anchored at the reference view absorbs cross-view inconsistency. Unlike prior Eulerian methods limited to fluid-like motion, we capture general deformation, object motion, and illumination change. We introduce a ground-truth-free evaluation covering vividness, naturalness, loop seam coherence, and scene quality. On 39 reconstructed and generated scenes, OuroWorld outperforms all baselines and wins 70.8%–99.0% of user-study comparisons.

1 INTRODUCTION

Recent 3D world models, such as HY-World 2.0 (HY-World et al., 2026), Marble (World Labs, 2025), and Lyra 2.0 (Shen et al., 2026), generate photorealistic, freely explorable 3D scenes, yet these worlds are frozen in time: leaves do not sway, mist does not drift, and lights do not flicker. Our goal is to bring them alive, endlessly (Fig. 1).

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.** *Left:* 3D world models generate frozen worlds; 3D cinemagraphs bring them alive. *Middle:* Existing 3D cinemagraph methods rely on Eulerian flow and are limited to fluid-like motion. *Right:* We turn any static 3D scene into a 3D cinemagraph, which requires (1) seamless looping and (2) vivid, diverse dynamics.

Method	Mask-free	True 3D	Mathematical Looping	Diverse Motion	Illumination
Gaussians-to-Life	✗	✓	✗	✓	✗
3D Cinemagraphy	✗	✗ (LDIs)	✓	✗ (fluid only)	✗
LoopGaussian	✗	✓	✓	✗ (soft material only)	✗
3D-MOM	✗	✓	✗	✗ (fluid only)	✗
Ours	✓	✓	✓	✓	✓

Table 1: **Capability comparison.** *Mask-free*: no user-provided 2D/3D motion mask. *True 3D*: free-viewpoint rendering, not pseudo-3D *Mathematical looping*: the 4D output is guaranteed to loop. *Diverse motion*: not limited to one motion type. *Illumination*: time-varying lighting. Ours is the only method that satisfies all five properties.

In 2D, cinemagraphs achieve this by adding seamlessly looping motion to a photograph (Liao et al., 2013; Holynski et al., 2021; Mahapatra et al., 2026), but their 3D extensions remain limited. Multiplane or layered-depth methods (Ma et al., 2023; Li et al., 2023; Shen et al., 2023) are pseudo-3D with small viewpoint ranges, while LoopGaussian (Li et al., 2024a) and 3D-MOM (Jin et al., 2025) use 3DGS (Kerbl et al., 2023) but rely on Eulerian flow, which captures only fluid-like motion and cannot express general deformation, object motion, or illumination change. All of them also require a motion mask and cannot decide where and how the scene should move (Fig. 2 and Table 1).

Video generation models (Kong et al., 2024; Wan et al., 2025) synthesize diverse, realistic dynamics that fill this gap, but turning them into a 3D cinemagraph poses two challenges: (1) generated videos are only approximately periodic, so fitting them leaves a visible seam; and (2) a single view cannot constrain a 4D scene, yet generated multi-view videos are mutually inconsistent, so fitting them blurs the scene and produces erratic motion.

We present OuroWorld, a mask-free framework that converts any static 3DGS scene into a 3D cinemagraph (Fig. 3). A vision-language model (Singh et al., 2025) infers plausible dynamics and guides a video model to generate a reference video; we extend this to multi-view videos and fit them with an Inconsistency-Robust Periodic 4DGS. It resolves (1) with a Fourier-series deformation that loops by construction, and (2) with a Grounded Drift Field that absorbs cross-view inconsistency while anchored at the reference view.

Since no ground truth exists for this task, we further propose a ground-truth-free evaluation of vividness, naturalness, loop seam coherence, and scene quality. On 30 world-model-generated and 9 reconstructed Mip-NeRF 360 (Barron et al., 2022) scenes, our method outperforms all baselines and is preferred in 70.8%–99.0% of user-study comparisons.

Our contributions are as follows:

- The first mask-free 3D cinemagraph framework, turning any static 3DGS scene into a vivid, seamlessly looping dynamic scene with general deformation, motion, and illumination change beyond Eulerian methods.
- Inconsistency-Robust Periodic 4DGS guaranteeing looping via a Fourier-series deformation and learns robustly from inconsistent generated videos via a novel Grounded Drift Field.
- A perceptually-aligned ground-truth-free evaluation framework for arbitrary scenes, covering scene quality, loop seam coherence, vividness, and naturalness, under which our method outperforms all baselines on both this framework and the user study.

2 Related Work

Cinemagraphs and looping videos. A cinemagraph is a photograph in which selected regions move in a seamless loop. Early systems extract such loops from an input video by finding seamless frame

or per-pixel periods (Schodl et al., 2023; Tompkin et al., 2011; Yeh & Li, 2012; Liao et al., 2013; 2015; Oh et al., 2017), optionally de-animating static regions (Bai et al., 2012). Animating a single photograph instead requires a motion prior: stochastic motion textures (Chuang et al., 2005), class-specific physical models (Jhou & Cheng, 2015), detected periodic patterns (Halperin et al., 2021), learned time-lapse (Endo et al., 2019) or Eulerian-flow predictors (Holynski et al., 2021; Mahapatra & Kulkarni, 2022; Fan et al., 2023; Mahapatra et al., 2023; Choi et al., 2024), and frequency-domain oscillations (Li et al., 2024c). Image-to-video diffusion models (Blattmann et al., 2023; Xing et al., 2024) now provide far richer priors, and loop-specific variants cyclically shift latents or positional embeddings (Bi et al., 2025; Dong et al., 2026), animate salient objects (Wang et al., 2024a), or condition the first and last frames on the same photograph (Mahapatra et al., 2026), which we adopt to obtain a reference loop. All of these, however, animate a *2D* image plane, and the Eulerian ones are confined to fluid-like motion. We instead produce a *3D* scene renderable from arbitrary viewpoints, with loop closure enforced by the representation itself rather than by the 2D generator.

Static 3D scene representations and generated worlds. Single-image view synthesis lifts a photo into a shallow proxy—a point cloud (Niklaus et al., 2019; Wiles et al., 2020), multiplane image (Zhou et al., 2018; Tucker & Snavely, 2020), or layered depth image (Shade et al., 1998; Tulsiani et al., 2018; Shih et al., 2020)—with limited parallax, whereas neural radiance fields (Mildenhall et al., 2021; Barron et al., 2022) and 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023; Yu et al., 2024c; Huang et al., 2024a) reconstruct complete scenes, the latter in real time. They assume *consistent* captured observations, and some work handles inconsistency with per-image latent codes (Martin-Brualla et al., 2021; Park et al., 2021a; Kulhanek et al., 2024) or ICP-estimated per-frame drift (Höllein & Nießner, 2026). 3DGS is also the output of scene generators that build explorable worlds by progressive inpainting and lifting (Höllein et al., 2023; Chung et al., 2023; Yu et al., 2024b; 2025a; Zhou et al., 2024; Liang et al., 2025) or with large-scale world models (Team et al., 2025; HY-World et al., 2026; Huang et al., 2025c; World Labs, 2025; Bahmani et al., 2026; Shen et al., 2026).

Dynamic 3D representations and 4D generation. Dynamic radiance fields and 3DGS add time via canonical-space deformation (Pumarola et al., 2021; Park et al., 2021a;b; Yang et al., 2024b; Wu et al., 2024), scene flow (Li et al., 2021), factorized space-time grids (Fridovich-Keil et al., 2023; Cao & Johnson, 2023; Wang et al., 2022), per-frame optimization (Luiten et al., 2024), sparse control points (Huang et al., 2024b), native 4D primitives (Yang et al., 2024a; Duan et al., 2024), or per-Gaussian temporal bases (Li et al., 2024b; Lin et al., 2024; Kratimenos et al., 2024; Chen et al., 2026; Shih et al., 2024; Zhang et al., 2026), and now handle casual monocular videos (Wang et al., 2025c; Lei et al., 2025). 4D generation synthesizes objects via score distillation (Singer et al., 2023; Bahmani et al., 2024b; Ling et al., 2024; Ren et al., 2023; Bahmani et al., 2024a) or multi-view video diffusion (Xie et al., 2025; Ren et al., 2024; Jiang et al., 2024), and recently whole scenes (Yu et al., 2024a; Wang et al., 2025a; Sun et al., 2024; Zhao et al., 2025; Wu et al., 2025; Liu et al., 2025; Pan et al., 2025), with Free4D (Liu et al., 2025) supervising generated views with diffusion-refined renders as we do. We restrict deformation to a Fourier series, making motion *periodic* by construction; our Grounded Drift Field absorbs generative inconsistency as a view-conditioned drift, anchored at the reference view and discarded at inference.

3D cinemagraphs and scene animation. 3D cinemagraphs splat single-image 2D Eulerian flow on layered depth (Li et al., 2023; Shen et al., 2023), assemble loops from asynchronous multi-view captures (Ma et al., 2023), or move 3D Gaussians along Eulerian fields (Li et al., 2024a; Jin et al., 2025); all rely on a user- or network-provided *mask* and stationary flow. Scene animation instead borrows motion from video diffusion (Lee et al., 2024; Wimmer et al., 2025), physics simulation (Xie et al., 2024; Zhang et al., 2024; Huang et al., 2025b; Liu et al., 2024), or language-model-written motion fields (Kiray et al., 2026), mostly within user-specified regions or, as in the concurrent AniGS (Cheng et al., 2026), segmentation masks; none targets a seamless loop. Our method is *mask-free*: a vision-language model describes the motion, a video diffusion model realizes it, and a periodic 4D representation guarantees an endless loop from any viewpoint (Table 1).

3 METHOD

Given a static 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) scene from any source, we automatically convert it into a 3D cinemagraph. There are three challenges: (1) obtaining diverse dynamics, (2) guaranteeing looping motion, and (3) learning 4D from inconsistent generated multi-view supervision. We propose **OuroWorld**, a 3-stage framework (Fig. 3) including looping video generation (Section 3.1), multi-view video generation (Section 3.2), and Inconsistency-Robust Periodic 4DGS optimization (Sections 3.3 to 3.4). See Appendix B for details.

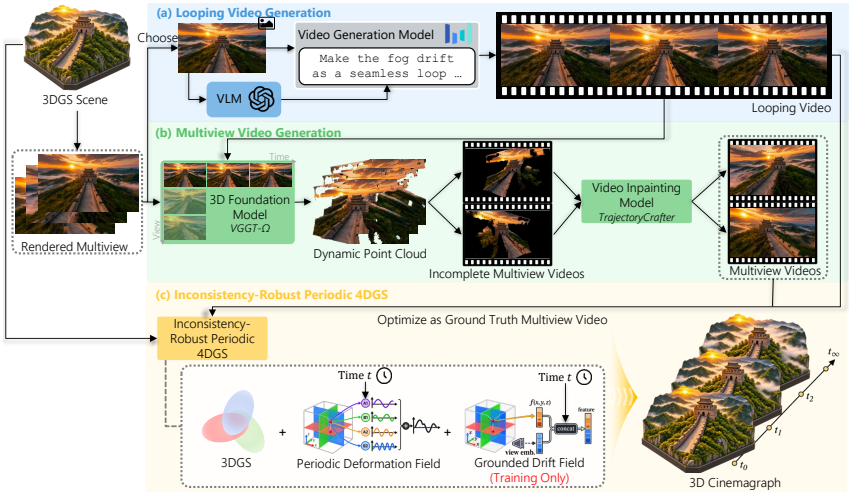


Figure 3: **Overview of OuroWorld.** **(a) Looping video generation.** A VLM selects a reference view rendered from the input 3DGS, infers plausible scene dynamics, and prompts a video generation model to synthesize an approximately looping reference video. **(b) Multi-view video generation.** A 3D foundation model (VGGT-Ω) lifts the reference video into a dynamic point cloud, using static multi-view renders to improve depth. The point cloud is rendered from N surrounding viewpoints into incomplete videos with disoccluded holes, which a video inpainting model (TrajectoryCrafter) completes into multi-view videos. **(c) Inconsistency-Robust Periodic 4DGS.** The 3D cinemagraph is represented as a canonical 3DGS deformed by a Fourier-parameterized Periodic Deformation Field, which loops by construction. A Grounded Drift Field absorbs cross-view inconsistency of the generated videos during training and is discarded at inference.

3.1 Looping Video Generation

Unlike Eulerian flow methods (Holynski et al., 2021; Li et al., 2023; Jin et al., 2025; Li et al., 2024a), which are limited to fluid-like motion and require manual masks, modern video diffusion models (Kong et al., 2024; Wan et al., 2025) synthesize diverse, realistic dynamics, including approximately looping videos. As shown in Fig. 3a, given a reference image I_{ref} rendered from the input 3DGS, a vision-language model (VLM; GPT-5.5 (Singh et al., 2025)) infers what should move and how, and writes a prompt describing natural cyclic motion (details in Appendix B). Conditioned on this, a video generation model (Seedance 2.0 (Seedance et al., 2026)) produces a 10-second reference video V_{ref} with I_{ref} as both first and last frame, and the VLM filters out results with unnatural motion. This module accepts any video generator capable of producing approximately looping videos, including prompt-controllable ones (Bi et al., 2025; Mahapatra et al., 2026).

3.2 Multi-view Video Generation

A single view is insufficient to constrain a full 4D scene. We therefore extend the reference video into multi-view videos. As shown in Fig. 3b, we feed the reference video together with multi-view renders from input 3DGS into a 3D foundation model (VGGT-Ω (Wang et al., 2026; 2025b)), to estimate a per-frame dynamic point cloud of the reference video (Wang et al., 2024b; Zhang et al., 2025); the static renders serve only to improve depth estimation. We then sample $N{=}20$ viewpoints uniformly within $\pm 20^\circ$ of horizontal rotation around a pivot and render the dynamic point cloud from these viewpoints into incomplete videos containing disoccluded holes. Finally, the video inpainting model (TrajectoryCrafter (Yu et al., 2025b)), conditioned on the reference video, inpaints each view, producing multi-view videos$\{V_v\}_{v=1}^{N}$.

3.3 Inconsistency-Robust Periodic 4DGS

As shown in Fig. 3c, we represent the 3D cinemagraph with Inconsistency-Robust Periodic 4DGS, comprising a canonical 3DGS $\mathcal{G}$, a Periodic Deformation Field $\mathcal{P}$, and a Grounded Drift Field Δ. Each Gaussian in $\mathcal{G}$ is parameterized by position $\mathbf{x} \in \mathbb{R}^3$, rotation $\mathbf{q}$, scale $\mathbf{s}$, opacity σ, and color $\mathbf{c}$.

3.3.1 Periodic Deformation Field

We parameterize the temporal variation of each Gaussian with a Fourier series (Fig. 3c), which is periodic by construction and can approximate arbitrary smooth periodic motion, enabling diverse dynamics beyond the fluid-like motion of Eulerian flow.

Following factorized spatial encodings (Chan et al., 2022; Fridovich-Keil et al., 2023; Cao & Johnson, 2023; Wu et al., 2024), we query a triplane at the canonical center $\mathbf{x}$ and obtain a feature $\mathbf{F}(\mathbf{x}) \in \mathbb{R}^{(2K+1)C}$, which we split into per-frequency coefficients $\mathbf{a}_0, \{\mathbf{a}_k, \mathbf{b}_k\}_{k=1}^{K} \in \mathbb{R}^C$. The time-dependent feature is

$$\mathbf{f}(\mathbf{x}, t) = \mathbf{a}_0 + \sum_{k=1}^{K} \left[\mathbf{a}_k \cos\left(\frac{2\pi k t}{T}\right) + \mathbf{b}_k \sin\left(\frac{2\pi k t}{T}\right)\right], \tag{1}$$

where T is the loop period and K the temporal bandwidth. An MLP decodes it into attribute offsets $\mathcal{P}(\mathbf{x}, t) = (\delta\mathbf{x}, \delta\mathbf{q}, \delta\mathbf{s}, \delta\mathbf{c}) = \mathrm{MLP}_{\mathcal{P}}(\mathbf{f}(\mathbf{x}, t))$. Since $\mathbf{f}(\mathbf{x}, t{+}T) = \mathbf{f}(\mathbf{x}, t)$, $\mathcal{P}$ is T-periodic and $\mathcal{G} + \mathcal{P}(\mathbf{x}, t)$ loops seamlessly regardless of the optimization outcome. Unlike prior Fourier-domain dynamic representations (Lin et al., 2024; Li et al., 2024b), which use such bases for fitting capacity, we exploit periodicity to guarantee looping.

3.3.2 Grounded Drift Field

Drift. The generated multi-view videos are inconsistent across views. Fitting them directly forces the deformation field to average conflicting observations, blurring the scene and producing erratic motion (Fig. 4a). Video-to-World (Höllein & Nießner, 2026) explains such inconsistency as per-view *generative drift*, estimates it with ICP (Besl & McKay, 1992), and adds the drift before computing the loss, so the drift is not baked into the scene. However, ICP only aligns geometry, missing drift in attributes such as *scale* and *color*, and assumes a static scene, failing on dynamic content.

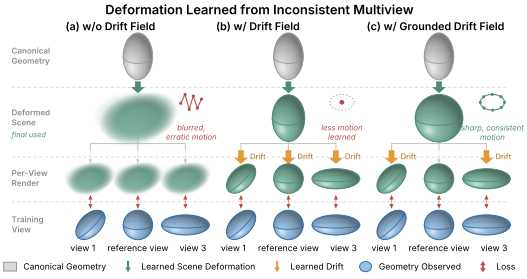


Figure 4: **Learning deformation from inconsistent multi-view videos.** Rows: canonical geometry, deformed scene at inference, per-view training renders, and generated training views. **(a) w/o Drift Field:** conflicting views are averaged, causing blur and erratic motion. **(b) w/ Drift Field:** drift on every view absorbs too much of the dynamics and is discarded at inference, reducing motion. **(c) w/ Grounded Drift Field (ours):** with no drift at the reference view, the deformation field must reproduce the reference dynamics, yielding a sharp scene with consistent motion.

We instead *learn* drift jointly with the 4D scene. A drift field Δ, sharing the triplane-plus-MLP architecture of $\mathcal{P}$, predicts view-specific, time-varying offsets to position, *scale* and *color*, $\Delta_v(\mathbf{x}, t) = \mathrm{MLP}_{\Delta}\big(\mathbf{F}_{\Delta}(\mathbf{x}), \gamma(t), \mathbf{e}_v\big)$, from a drift spatial feature $\mathbf{F}_{\Delta}(\mathbf{x})$, a time embedding $\gamma(t)$, and a per-view embedding $\mathbf{e}_v$. Adding Δ_v when rendering view v lets the loss attribute view-specific inconsistency to Δ_v rather than the shared scene, without any static-scene assumption. At inference, we discard Δ and render $\mathcal{G} + \mathcal{P}(\mathbf{x}, t)$, which retains dynamics that are largely free of cross-view inconsistency.

Grounding. Learning drift in dynamic scenes introduces an ambiguity: since both $\mathcal{P}$ and Δ depend on time, drift applied to all views can more easily explain most dynamics, leaving the final scene nearly static (Fig. 4b). We resolve this by grounding the optimization in the reference view, whose dynamics come directly from the video generation model and are the most reliable:

$$\mathcal{G}_{t,v} = \begin{cases} \mathcal{G} + \mathcal{P}(\mathbf{x}, t), & v = \text{ref}, \\ \mathcal{G} + \mathcal{P}(\mathbf{x}, t) + \Delta_v(\mathbf{x}, t), & \text{otherwise}. \end{cases} \tag{2}$$

$\mathcal{P}$ must thus reproduce the reference dynamics by itself, while Δ_v only absorbs each view's residual deviation from the reference (Fig. 4c).

3.4 Scene-View Consistent Optimization

While the Grounded Drift Field handles inconsistencies among generated views, the generated content may also be inconsistent with the input 3DGS. Following Free4D (Liu et al., 2025), we supervise highly consistent images, namely the reference video V_{ref} and the $t{=}0$ multi-view renders of the input

3DGS, with an ℓ_1 loss $\mathcal{L}_1$. For the remaining generated frames, we perturb the rendering with noise and denoise it with an image diffusion model (Stable Diffusion v1.5 (Rombach et al., 2022)) guided by the generated frame, yielding refined images. We then supervise the rendering with an LPIPS loss (Zhang et al., 2018) $\mathcal{L}_\text{p}$ against the refined image. The total objective is $\mathcal{L} = \lambda_1\mathcal{L}_1 + \lambda_\text{p}\mathcal{L}_\text{p}$; details are provided in the Appendix C.

4 Experiments

We compare our method with three 3D cinemagraph methods (Li et al., 2023; 2024a; Jin et al., 2025) and one method that brings 3D scenes to life with video priors (Wimmer et al., 2025). The evaluation uses ground-truth-free automatic metrics we proposed together with a user study.

4.1 Dataset

We evaluate on 39 scenes, each consisting of a static 3DGS and a designated reference camera pose. Since reconstructed and generated 3DGS differ substantially in Gaussian density, geometric fidelity, and layout, we draw 9 scenes from Mip-NeRF 360 (Barron et al., 2022) and 10 each from HY-World 2.0 (HY-World et al., 2026), Marble (World Labs, 2025), and Lyra 2.0 (Shen et al., 2026) (details in Appendix A). The scenes span indoor rooms, landscapes, architecture, and stylized environments, covering both the fluid and cloth-like dynamics targeted by Eulerian methods and motions beyond them, such as flickering light and twinkling scattered stardust.

4.2 Baselines

We compare against four methods. Gaussians-to-Life (Wimmer et al., 2025) lifts video-diffusion motion (Blattmann et al., 2023) to 3D point flow that drives a 3DGS, but does not target looping. 3D Cinemagraphy (Li et al., 2023) and 3D-MOM (Jin et al., 2025) animate a single image with an Eulerian motion prior trained on fluid videos (Holynski et al., 2021), rendered via a layered depth image (Shade et al., 1998) and 4D Gaussians (Wu et al., 2024), respectively; the latter's optimized 4D output is not guaranteed to loop. LoopGaussian (Li et al., 2024a) heuristically moves only Gaussian positions along a bidirectionally blended Eulerian field. For fairness, we also apply the same GPT-5.5 filtering to all baselines. Full descriptions are in Appendix D.

Input alignment. All baselines require inputs beyond a static 3DGS, which we derive from our pipeline for a fair comparison. *Input images* are rendered from the input 3DGS at the reference pose; *text prompts* are the GPT-5.5 (Singh et al., 2025) dynamics prompts from Section 3.1; for *motion masks*, GPT-5.5 names the objects to move and SAM 3 (Carion et al., 2026) segments them into 2D masks, which we back-project with the rendered 3DGS depth for methods requiring 3D masks. Further details are in Appendix D.

4.3 Evaluation Protocol

Since all methods output scene representations, we render each under identical camera settings: a 20-second, 30 FPS video (600 frames) at 672×384 along two trajectories per scene. The **static camera** stays fixed at the reference pose. The **orbit camera** evaluates 3D consistency: it faces a pivot, obtained by back-projecting a selected pixel at the designated pose, at $0.95\times$ the camera-to-pivot distance, and sweeps in yaw through $0^\circ \to 20^\circ \to -20^\circ \to 0^\circ$ over 600 uniformly interpolated views. Both trajectories span many motion periods. We keep each method's native period and repeat it to fill 20 seconds, rather than normalizing periods, since changing the period alters motion speed and introduces artifacts unrelated to the method (Appendix D).

4.4 Evaluation Metrics

Prior 3D cinemagraph works measure quality against ground truth. Their data, however, is limited to fluid datasets (Holynski et al., 2021) or object-level synthetic scenes (Li et al., 2024a), and does not generalize to the diverse scenes. We therefore propose a ground-truth-free evaluation framework based properties a good 3D cinemagraph should have: *vivid*, *natural*, *seamless loop*, *high visual quality*. These metrics are perceptually aligned, as supported by the user study (Section 4.8).

Vividness. This metric measures how rich the displayed dynamics are. We temporally downsample each video to 3 FPS and define the Vividness Degree as the average of three components:

$$\text{Vividness} = \tfrac{1}{3}\big(\text{MV} + \text{IV} + \text{VV}\big). \tag{3}$$

Each component is computed between adjacent frames and averaged over the video, with thresholds set to the level of change perceptible to human vision: *Motion Variation* (MV) is the fraction

Table 2: **Quantitative result.** *Vividness Degree* measures motion, illumination, and appearance change. *KVD* measures the distance to the natural video distribution. *MALF* measures whether the scene loops while it actually moves, and *Seam SSIM* measures the similarity at the seam of consecutive cycles; *Aesthetic Quality* and *Overall Consistency* follow VBench to assess scene quality. **Bold** and underline denote the best and second-best results. Ours achieves the best Vividness, Naturalness, and Aesthetic Quality. Near-static methods (LoopGaussian, 3D Cinemagraphy) trivially reach a high Seam SSIM but near-zero MALF, whereas ours achieves the best MALF with a high Seam SSIM.

		Vividness	Naturalness	Loop Seam Coherence		Scene Quality	
	Method	Vividness Degree ↑	KVD ↓	Motion-Aware Loop Fidelity ↑	Seam SSIM ↑	Aesthetic Quality ↑	Overall Consist. ↑
Static	Gaussians-to-Life	0.0374	127.47	0.0265	0.9388	0.6602	0.2109
	3D Cinemagraphy	0.0066	142.42	0.0134	0.9995	0.6374	0.2110
	LoopGaussian	0.0024	154.65	0.0025	**0.9998**	0.6718	0.2100
	3D-MOM	0.0247	155.23	0.0317	0.9158	0.6245	0.2069
	Ours	**0.0423**	**108.23**	**0.0809**	0.9965	**0.6748**	**0.2111**
Orbit	Gaussians-to-Life	0.5097	51.86	–	–	0.6482	0.2163
	3D Cinemagraphy	0.4945	102.58	–	–	0.5539	0.2156
	LoopGaussian	0.5087	53.38	–	–	0.6470	0.2161
	3D-MOM	0.4724	77.83	–	–	0.5501	**0.2191**
	Ours	**0.5117**	**50.11**	–	–	**0.6659**	0.2185

of pixels whose optical-flow magnitude, estimated by SEA-RAFT (Wang et al., 2024d), exceeds $0.3\% \times \min(H, W)$; *Illumination Variation* (IV) is the fraction of pixels whose grayscale luminance changes by more than 20%; and *Visual Variation* (VV) measures global appearance change as $1 - \mathrm{SSIM}$ (Wang et al., 2004), following VideoScore (He et al., 2024).

Naturalness. We treat a large video generation model as an approximation of the natural video distribution and measure KVD (Unterthiner et al., 2018) against its outputs. Using the open-source MiniMax H3 (MiniMax, 2026), distinct from our Seedance 2.0 to avoid bias, we generate 20 videos per scene (780 in total), conditioned on the reference view and the GPT-5.5 dynamics prompt. For each method, we compute KVD between its renders across all scenes and both trajectories.

Loop Seam Coherence. We evaluate whether each cycle returns to its starting state and whether the transition across cycles is continuous, using two metrics:

- *Motion-Aware Loop Fidelity (MALF).* Since a static scene trivially loops, MALF rewards matching the same phase one period later relative to other phases, and is zero for a static video. With $M{=}10$ time points $\{t_i\}_{i=1}^{M}$ sampled in one period,

$$\mathrm{MALF} = \frac{1}{M}\sum_{i=1}^{M}\Big[\mathrm{SSIM}\big(I_{t_i}, I_{t_i+T}\big) - \frac{1}{M-1}\sum_{j\neq i}\mathrm{SSIM}\big(I_{t_i}, I_{t_j+T}\big)\Big]. \tag{4}$$

- *Seam SSIM.* The SSIM between the last frame of one cycle and the first frame of the next.

Both are computed on a single seam, making them independent of the number of cycles, and under a static camera to exclude viewpoint changes.

Scene Quality. We adopt VBench (Huang et al., 2024c) *Aesthetic Quality* (Schuhmann et al., 2022) and *Overall Consistency* (Wang et al., 2024c).

4.5 Implementation Details

The hyperparameters of ours are listed in Appendix B. For baselines, we use the official implementations (details in Appendix D). All experiments are run on a single NVIDIA RTX 4090 GPU.

4.6 Quantitative Results

As shown in Table 2, our method achieves the best Vividness, Naturalness, MALF, and Aesthetic Quality under both camera settings. LoopGaussian and 3D Cinemagraphy attain slightly higher Seam SSIM, but they are nearly static, as their low Vividness and MALF indicate, so their seams are trivially continuous. Our method, in contrast, combines high Seam SSIM with the highest MALF, showing that its motion is both substantial and seamlessly looping.

4.7 Qualitative Results

Fig. 5 shows results under the orbit camera. 3D Cinemagraphy and 3D-MOM produce disoccluded holes at novel views, while Gaussians-to-Life and LoopGaussian remain nearly static. In contrast, our method produces vivid dynamics and stays consistent across viewpoints.

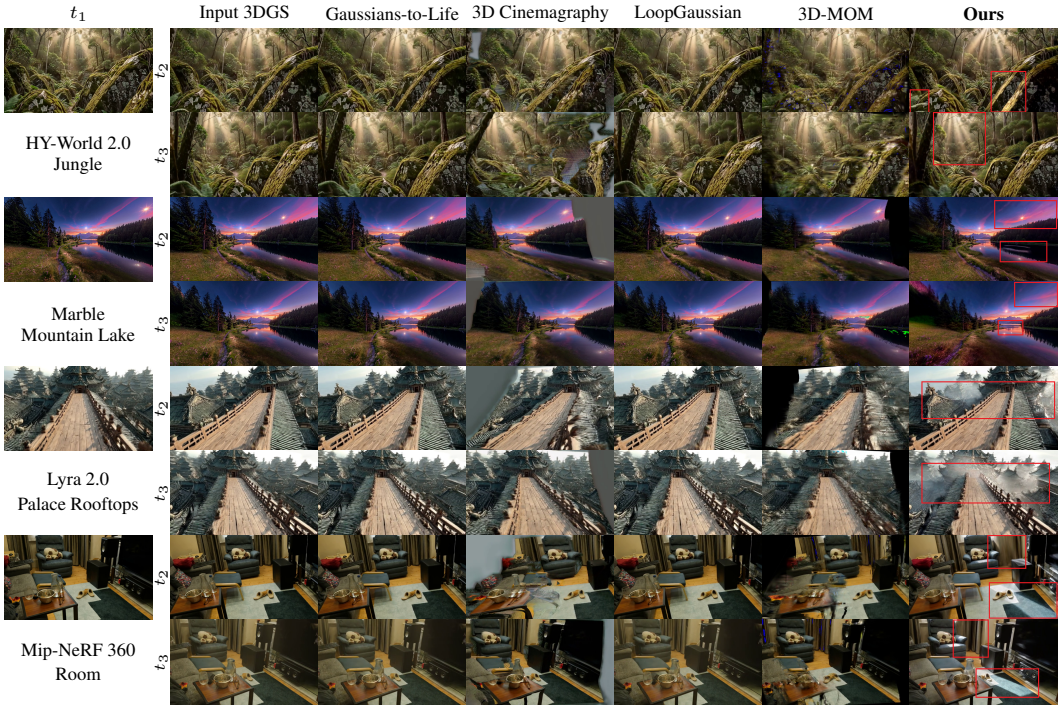


Figure 5: **Qualitative comparison under the orbit camera**, one scene per source. The leftmost column shows the input 3DGS at t_1; the others show each method at later time steps t_2 and t_3. 3D Cinemagraphy and 3D-MOM produce disocclusion holes and color artifacts, while Gaussians-to-Life and LoopGaussian remain nearly static. OuroWorld produces vivid, view-consistent dynamics such as changing illumination, rippling water, drifting fog, and curtains swaying in the wind. See the supplementary videos for full loops.

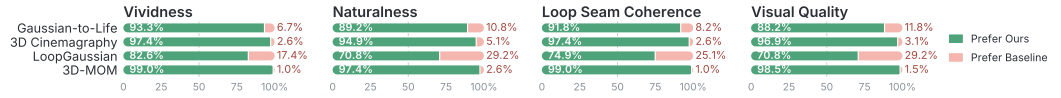


Figure 6: **User study.** In a two-alternative forced-choice study, participants compare OuroWorld with each baseline under four criteria. Bars show the percentage of votes preferring ours (left) versus the baseline (right). Ours is preferred in 70.8%–99.0% of comparisons across all baselines and criteria, with the largest margins in Vividness. 30 participants took part; see Appendix F for details.

4.8 User Study

Since automatic metrics only approximate perceptual qualities, we further conduct a two-alternative forced-choice study in which 30 participants compare our result with each baseline under four criteria: Vividness, Naturalness, Loop Seam Coherence, and Visual Quality (details in Appendix F). As shown in Fig. 6, our method is consistently preferred across all baselines and criteria, with the largest margins in Vividness, highlighting the diverse dynamics enabled by our video generation prior. These preferences are consistent with the results in Table 2 and support the validity of our designed metrics.

4.9 Ablation Study

We ablate three components in Table 3 and one in the periodic deformation paragraph. Besides Vividness, we report VBench (Huang et al., 2024c) Subject and Background Consistency (Caron et al., 2021; Radford et al., 2021). We also report the variance of the Laplacian (VoL) (Pech-Pacheco et al., 2000) as a measure of sharpness.

Drift field. We remove the Grounded Drift Field and fit all views with the Periodic Deformation Field $\mathcal{P}$ alone (Fig. 4a). $\mathcal{P}$ is then forced to average inconsistent multi-view observations. As shown in Table 3, this approach lowers VoL and consistency across both cameras, and causes fine details to fade over time (Fig. 7). This confirms the benefit of modeling per-view variation separately.

Table 3: **Ablation on consistency components.** As illustrated in Fig. 4, removing the *drift field* blurs the scene (low VoL) and lowers both subject and background consistency; removing *grounding* lets drift absorb the dynamics, reducing the scene motion (lowest Vividness); removing *scene-view consistent optimization* breaks scene-view consistency and blurs the scene most severely (lowest subject consistency, VoL -22% under orbit). Our full model is best on all orbit-camera metrics.

Camera	Method	Subject Consistency ↑	Background Consistency ↑	VoL ↑	Vividness Degree ↑
Static camera	w/o Drift	0.9919	0.9818	883.55	0.0424
	w/o Grounding	0.9918	**0.9833**	882.44	0.0392
	w/o Scene-View Consistent Optimization	0.9881	0.9817	876.57	**0.0450**
	Ours	**0.9924**	0.9831	**892.43**	0.0423
Orbit camera	w/o Drift	0.9666	0.9615	704.34	0.5113
	w/o Grounding	0.9649	0.9604	681.16	0.5080
	w/o Scene-View Consistent Optimization	0.9642	0.9611	551.33	0.5081
	Ours	**0.9673**	**0.9622**	**709.75**	**0.5117**

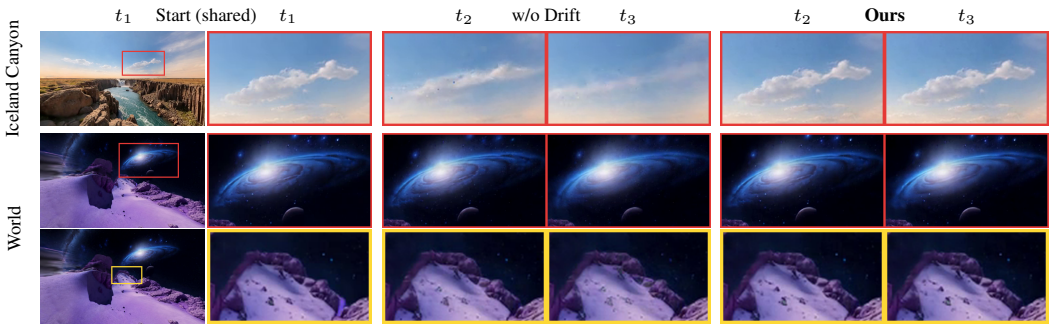


Figure 7: **Effect of drift.** Without drift, fine details degrade over time: the clouds in *Iceland Canyon* blur and fade away (top row), the galaxy in *World* becomes blurry (middle row), and the rocks on the ground vanish (bottom row). In contrast, our full model preserves these details while maintaining vivid and coherent motion. The first column shows the full frame at t_1 with the zoomed region outlined; the remaining columns zoom into that region at t_1, and at t_2 and t_3 for each method.

Grounding. We also apply Drift Field Δ_v to the reference view (Fig. 4b), removing the case split in Eq. (2). Motion can then be explained by either $\mathcal{P}$ or Δ_v, and the more flexible, view-conditioned Δ_v absorbs it. As shown in Table 3, since Δ is discarded at inference, Vividness drops to the lowest among all variants ($0.0423 \rightarrow 0.0392$ under the static camera). Grounding prevents this by requiring $\mathcal{P}$ alone to reproduce the reference dynamics.

Scene-view consistent optimization. We replace the objective of Section 3.4 with a naive ℓ_1 loss on all generated views and timesteps, without diffusion-based LPIPS loss refinement (Liu et al., 2025; Meng et al., 2021; Rombach et al., 2022). Inpainting inconsistencies (Yu et al., 2025b) are baked into the scene, giving the lowest Subject Consistency and a 22% VoL drop under the orbit camera.

Periodic deformation. We double the period of the Fourier basis in Eq. (1) to $2T$, which keeps the capacity unchanged but makes the deformation no longer T-periodic, so the loop is no longer guaranteed. Under the static camera, Seam SSIM drops from 0.9965 to 0.9783: the last frame of each cycle no longer matches its first, causing a visible jump at the seam. MALF, in contrast, barely changes ($0.0809 \rightarrow 0.0788$), showing that the degradation concentrates at the loop boundary rather than in the overall motion.

5 Conclusion and Limitation

We presented OuroWorld, a mask-free framework that brings any static 3DGS scene alive as a seamlessly looping 3D cinemagraph. It distills dynamics from a VLM-guided video generation prior into an Inconsistency-Robust Periodic 4DGS. The resulting scenes exhibit diverse motion and illumination change beyond Eulerian flow, loop exactly by construction, and remain sharp despite the inconsistent generated views. With our ground-truth-free evaluation framework, experiments on both reconstructed and generated scenes show consistent gains over prior 3D cinemagraph methods.

Limitations. Our dynamics are bounded by the video generation prior, so implausible generations propagate to the final scene, although VLM filtering mitigates this. Moreover, completing all views from a single reference video leads to greater disocclusions and inconsistencies at wider viewpoint ranges. Generating multiple reference videos from distinct viewpoints is a promising remedy.

AI USE STATEMENT

In this work, generative AI models are integral components of the proposed method: a vision-language model (GPT-5.5) infers scene dynamics and writes motion prompts, a video generation model (Seedance 2.0) synthesizes the reference video, TrajectoryCrafter and Stable Diffusion v1.5 are used for view completion and refinement, and MiniMax H3 generates reference videos for the KVD metric. Their roles are fully described in Sections 3 and 4. Beyond the method itself, we used LLM-based assistants for polishing the writing and grammar of the manuscript and assisting with code implementation, e.g., evaluation and plotting scripts. We did not use generative AI tools for generating research ideas, designing experiments, or analyzing results. We have reviewed all AI-assisted work: all text edited with AI assistance was checked and revised by the authors, and AI-assisted code was verified and tested for correctness by at least two authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our method animates static 3D scenes and could, like other generative techniques, be misused to create misleading dynamic content. Our scenes depict landscapes, architecture, and indoor environments rather than people, and we do not target human animation; nevertheless, we encourage clearly labeling outputs as synthetic. Motion is inherited from pretrained video and vision-language models, so any biases in these models may propagate to the generated dynamics. All scenes come from the public Mip-NeRF 360 dataset or were generated with publicly available 3D world models (HY-World 2.0, Marble, Lyra 2.0) under their respective terms of use. Our user study involved 30 adult volunteers who compared rendered videos; participation was voluntary and anonymous, no personally identifiable information was collected, and the study was exempted by institution's review board. We are not aware of conflicts of interest beyond those declared in the submission system.

REPRODUCIBILITY STATEMENT

We describe the full pipeline in Section 3, including the looping video generation (Section 3.1), multi-view video generation (Section 3.2), the Inconsistency-Robust Periodic 4DGS formulation (Section 3.3, Eqs. (1) to (2)), and the optimization objective (Section 3.4). Further optimization details and all hyperparameters are given in Appendices B and C. The evaluation protocol, including camera trajectories, frame rate, resolution, and loop handling, is specified in Section 4.3, and all metrics are defined in Section 4.4 (Eqs. (3) to (4)). Baseline configurations and the procedure used to align their inputs are described in Section 4.2 and Appendix D, and user-study details are in Appendix F. All experiments run on a single NVIDIA RTX 4090 GPU. Since several components (GPT-5.5, Seedance 2.0) are proprietary and stochastic, exact outputs may vary across runs; to mitigate this, we will release the generated reference videos, multi-view videos, motion prompts, input 3DGS scenes and camera poses. The pipeline is model-agnostic and accepts any video generator capable of producing approximately looping videos (Section 3.1).

REFERENCES

Sherwin Bahmani, Xian Liu, Wang Yifan, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. In *European Conference on Computer Vision*, pp. 53–72. Springer, 2024a.

Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7996–8006. IEEE, 2024b.

Sherwin Bahmani, Tianchang Shen, Jiawei Ren, Jiahui Huang, Yifeng Jiang, Haithem Turki, Andrea Tagliasacchi, David Lindell, Zan Gojcic, Sanja Fidler, et al. Lyra: Generative 3d scene reconstruction via video diffusion model self-distillation. In *International Conference on Learning Representations*, volume 2026, pp. 82850–82880, 2026.

Jiamin Bai, Aseem Agarwala, Maneesh Agrawala, and Ravi Ramamoorthi. Selectively de-animating video. *ACM Trans. Graph.*, 31(4):66–1, 2012.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5460–5469. IEEE, 2022.

Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pp. 586–606. Spie, 1992.

Xiuli Bi, Jianfei Yuan, Bo Liu, Yong Zhang, Xiaodong Cun, Chi-Man Pun, and Bin Xiao. Mobius: Text to seamless looping video generation via latent shift. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–10, 2025.

Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018.

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

Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 130–141. IEEE, 2023.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International conference on learning representations*, volume 2026, pp. 138846–138923, 2026.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9630–9640. IEEE, 2021.

Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16123–16133, 2022.

Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *International Journal of Computer Vision*, 134(3):83, 2026.

Yen-Chi Cheng, Chen Gao, Chuhan Chen, Tuotuo Li, Rajvi Shah, Ayush Saraf, Changil Kim, Liangyan Gui, Alexander Schwing, Johannes Kopf, et al. Anigs: Bridging rendering and diffusion prior for 3d scene animation. *arXiv preprint arXiv:2607.18539*, 2026.

Jongwoo Choi, Kwanggyoon Seo, Amirsaman Ashtari, and Junyong Noh. Stylecinegan: landscape cinemagraph generation using a pre-trained stylegan. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7872–7881. IEEE, 2024.

Yung-Yu Chuang, Dan B Goldman, Ke Colin Zheng, Brian Curless, David H Salesin, and Richard Szeliski. Animating pictures with stochastic motion textures. In *ACM SIGGRAPH 2005 Papers*, pp. 853–860. 2005.

Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384*, 2023.

Haotian Dong, Wenjing Wang, Chen Li, Jing Lyu, Xin Wang, and Di Lin. Loopy: Seamless video loop generation via anchored looping shift of positional embedding. *arXiv preprint arXiv:2608.23090*, 2026.

Yuanxing Duan, Fangyin Wei, Qiyu Dai, Yuhang He, Wenzheng Chen, and Baoquan Chen. 4d-rotor gaussian splatting: towards efficient novel view synthesis for dynamic scenes. In *ACM SIGGRAPH 2024 conference papers*, pp. 1–11, 2024.

Yuki Endo, Yoshihiro Kanamori, and Shigeru Kuriyama. Animating landscape: self-supervised learning of decoupled motion and appearance for single-image video synthesis. *arXiv preprint arXiv:1910.07192*, 2019.

Siming Fan, Jingtan Piao, Chen Qian, Hongsheng Li, and Kwan-Yee Lin. Simulating fluids in real-world still images. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15876–15885. IEEE, 2023.

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12479–12488. IEEE, 2023.

Tavi Halperin, Hanit Hakim, Orestis Vantzos, Gershon Hochman, Netai Benaim, Lior Sassy, Michael Kupchik, Ofir Bibi, and Ohad Fried. Endless loops: detecting and animating periodic patterns in still images. *ACM Transactions on graphics (TOG)*, 40(4):1–12, 2021.

Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, et al. Videoscore: Building automatic metrics to simulate fine-grained human feedback for video generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 2105–2123, 2024.

Lukas Höllein and Matthias Nießner. World reconstruction from inconsistent views. *arXiv preprint arXiv:2603.16736*, 2026.

Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7875–7886. IEEE, 2023.

Aleksander Holynski, Brian L Curless, Steven M Seitz, and Richard Szeliski. Animating pictures with eulerian motion fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5810–5819, 2021.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pp. 1–11, 2024a.

Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, et al. Vipe: Video pose engine for 3d geometric perception. *arXiv preprint arXiv:2508.10934*, 2025a.

Tianyu Huang, Haoze Zhang, Yihan Zeng, Zhilu Zhang, Hui Li, Wangmeng Zuo, and Rynson WH Lau. Dreamphysics: Learning physics-based 3d dynamics with video diffusion priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3733–3741, 2025b.

Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson Lau, Wangmeng Zuo, et al. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *ACM Transactions on Graphics (TOG)*, 44(6): 1–15, 2025c.

Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4220–4230. IEEE, 2024b.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21807–21818. IEEE, 2024c.

Team HY-World, Chenjie Cao, Xuhui Zuo, Zhenwei Wang, Yisu Zhang, Junta Wu, Zhenyang Liu, Yuning Gong, Yang Liu, Bo Yuan, et al. Hy-world 2.0: A multi-modal world model for reconstructing, generating, and simulating 3d worlds. *arXiv preprint arXiv:2604.14268*, 2026.

Wei-Cih Jhou and Wen-Huang Cheng. Animating still landscape photographs through cloud motion creation. *IEEE Transactions on Multimedia*, 18(1):4–13, 2015.

Yanqin Jiang, Chaohui Yu, Chenjie Cao, Fan Wang, Weiming Hu, and Jin Gao. Animate3d: Animating any 3d model with multi-view video diffusion. *Advances in Neural Information Processing Systems*, 37:125879–125906, 2024.

In-Hwan Jin, Haesoo Choo, Seong-Hun Jeong, Heemoon Park, Junghwan Kim, Oh-joon Kwon, and Kyeongbo Kong. Optimizing 4d gaussians for dynamic scene video from single landscape images. In *International Conference on Learning Representations*, volume 2025, pp. 80517–80537, 2025.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *European conference on computer vision*, pp. 18–35. Springer, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Mert Kiray, Paul Uhlenbruck, Nassir Navab, and Benjamin Busam. Promptvfx: Text-driven fields for open-world 3d gaussian animation. In *2026 International Conference on 3D Vision (3DV)*, pp. 1556–1566. IEEE, 2026.

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

Agelos Kratimenos, Jiahui Lei, and Kostas Daniilidis. Dynmf: Neural motion factorization for real-time dynamic view synthesis with 3d gaussian splatting. In *European Conference on Computer Vision*, pp. 252–269. Springer, 2024.

Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024.

Yao-Chih Lee, Yi-Ting Chen, Andrew Wang, Ting-Hsuan Liao, Brandon Y Feng, and Jia-Bin Huang. Vividdream: Generating 3d scene with ambient dynamics. *arXiv preprint arXiv:2405.20334*, 2024.

Jiahui Lei, Yijia Weng, Adam W Harley, Leonidas Guibas, and Kostas Daniilidis. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6165–6177. IEEE, 2025.

Jiyang Li, Lechao Cheng, Zhangye Wang, Tingting Mu, and Jingxuan He. Loopgaussian: creating 3d cinemagraph with multi-view images via eulerian motion field. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 476–485, 2024a.

Xingyi Li, Zhiguo Cao, Huiqiang Sun, Jianming Zhang, Ke Xian, and Guosheng Lin. 3d cinemagraphy from a single image. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4595–4605. IEEE, 2023.

Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8508–8520. IEEE, 2024b.

Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6498–6508, 2021.

Zhengqi Li, Richard Tucker, Noah Snavely, and Aleksander Holynski. Generative image dynamics. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24142–24153. IEEE, 2024c.

Hanwen Liang, Junli Cao, Vidit Goel, Guocheng Qian, Sergei Korolev, Demetri Terzopoulos, Konstantinos N Plataniotis, Sergey Tulyakov, and Jian Ren. Wonderland: Navigating 3d scenes from a single image. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 798–810. IEEE, 2025.

Jing Liao, Mark Finch, and Hugues Hoppe. Fast computation of seamless video loops. *ACM Transactions on Graphics (TOG)*, 34(6):1–10, 2015.

Zicheng Liao, Neel Joshi, and Hugues Hoppe. Automated video looping with progressive dynamism. *ACM Transactions on Graphics (TOG)*, 32(4):1–10, 2013.

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. *arXiv preprint arXiv:2511.10647*, 2025.

Youtian Lin, Zuozhuo Dai, Siyu Zhu, and Yao Yao. Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21136–21145. IEEE, 2024.

Huan Ling, Seung Wook Kim, Antonio Torralba, Sanja Fidler, and Karsten Kreis. Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8576–8588. IEEE, 2024.

Fangfu Liu, Hanyang Wang, Shunyu Yao, Shengjun Zhang, Jie Zhou, and Yueqi Duan. Physics3d: Learning physical properties of 3d gaussians via video diffusion. *arXiv preprint arXiv:2406.04338*, 2024.

Tianqi Liu, Zihao Huang, Zhaoxi Chen, Guangcong Wang, Shoukang Hu, Liao Shen, Huiqiang Sun, Zhiguo Cao, Wei Li, and Ziwei Liu. Free4d: Tuning-free 4d scene generation with spatial-temporal consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 25571–25582, 2025.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pp. 800–809. IEEE, 2024.

Li Ma, Xiaoyu Li, Jing Liao, and Pedro V Sander. 3d video loops from asynchronous input. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 310–320. IEEE, 2023.

Aniruddha Mahapatra and Kuldeep Kulkarni. Controllable animation of fluid elements in still images. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3657–3666. IEEE, 2022.

Aniruddha Mahapatra, Aliaksandr Siarohin, Hsin-Ying Lee, Sergey Tulyakov, and Jun-Yan Zhu. Text-guided synthesis of eulerian cinemagraphs. *ACM Transactions on Graphics (TOG)*, 42(6): 1–13, 2023.

Aniruddha Mahapatra, Long Mai, Cusuh Ham, and Feng Liu. Dreamloop: Controllable cinemagraph generation from a single photograph. *arXiv preprint arXiv:2601.02646*, 2026.

Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 7206–7215. IEEE, 2021.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

MiniMax. Minimax h3: An open model breaking the boundaries between tasks and modalities. `https://www.minimax.io/blog/minimax-h3`, July 2026. Accessed: 2026-09-26.

Simon Niklaus, Long Mai, Jimei Yang, and Feng Liu. 3d ken burns effect from a single image. *ACM Transactions on Graphics (ToG)*, 38(6):1–15, 2019.

Tae-Hyun Oh, Kyungdon Joo, Neel Joshi, Baoyuan Wang, In So Kweon, and Sing Bing Kang. Personalized cinemagraphs using semantic understanding and collaborative learning. In *Proceedings of the IEEE international conference on computer vision*, pp. 5160–5169, 2017.

Panwang Pan, Chenguo Lin, Jingjing Zhao, Chenxin Li, Yuchen Lin, Haopeng Li, Honglei Yan, Kairun Wen, Yunlong Lin, Yixuan Yuan, et al. Diff4splat: Controllable 4d scene generation with latent dynamic reconstruction models. *arXiv preprint arXiv:2511.00503*, 2025.

Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5845–5854. IEEE, 2021a.

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021b.

José Luis Pech-Pacheco, Gabriel Cristóbal, Jesús Chamorro-Martinez, and Joaquín Fernández-Valdivia. Diatom autofocusing in brightfield microscopy: a comparative study. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 3, pp. 314–317. IEEE, 2000.

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10313–10322. IEEE, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dreamgaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023.

Jiawei Ren, Kevin Xie, Ashkan Mirzaei, Hanxue Liang, Xiaohui Zeng, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung W Kim, et al. L4gm: Large 4d gaussian reconstruction model. *Advances in Neural Information Processing Systems*, 37:56828–56858, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Arno Schodl, Richard Szeliski, David H Salesin, and Irfan Essa. Video textures. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pp. 557–570. 2023.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026.

Jonathan Shade, Steven Gortler, Li-wei He, and Richard Szeliski. Layered depth images. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pp. 231–242, 1998.

Liao Shen, Xingyi Li, Huiqiang Sun, Juewen Peng, Ke Xian, Zhiguo Cao, and Guosheng Lin. Make-it-4d: Synthesizing a consistent long-term dynamic scene video from a single image. In *Proceedings of the 31st ACM international conference on multimedia*, pp. 8167–8175, 2023.

Tianchang Shen, Sherwin Bahmani, Kai He, Sangeetha Grama Srinivasan, Tianshi Cao, Jiawei Ren, Ruilong Li, Zian Wang, Nicholas Sharp, Zan Gojcic, et al. Lyra 2.0: Explorable generative 3d worlds. *arXiv preprint arXiv:2604.13036*, 2026.

Meng-Li Shih, Shih-Yang Su, Johannes Kopf, and Jia-Bin Huang. 3d photography using context-aware layered depth inpainting. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8025–8035. IEEE, 2020.

Meng-Li Shih, Jia-Bin Huang, Changil Kim, Rajvi Shah, Johannes Kopf, and Chen Gao. Modeling ambient scene dynamics for free-view synthesis. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.

Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, et al. Text-to-4d dynamic scene generation. *arXiv preprint arXiv:2301.11280*, 2023.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

Wenqiang Sun, Shuo Chen, Fangfu Liu, Zilong Chen, Yueqi Duan, Jun Zhang, and Yikai Wang. Dimensionx: Create any 3d and 4d scenes from a single image with controllable video diffusion. *arXiv preprint arXiv:2411.04928*, 2024.

HunyuanWorld Team, Zhenwei Wang, Yuhao Liu, Junta Wu, Zixiao Gu, Haoyuan Wang, Xuhui Zuo, Tianyu Huang, Wenhuan Li, Sheng Zhang, et al. Hunyuanworld 1.0: Generating immersive, explorable, and interactive 3d worlds from words or pixels. *arXiv preprint arXiv:2507.21809*, 2025.

James Tompkin, Fabrizio Pece, Kartic Subr, and Jan Kautz. Towards moment imagery: Automatic cinemagraphs. In *2011 conference for visual media production*, pp. 87–93. IEEE, 2011.

Richard Tucker and Noah Snavely. Single-view view synthesis with multiplane images. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 548–557. IEEE, 2020.

Shubham Tulsiani, Richard Tucker, and Noah Snavely. Layer-structured 3d scene inference via view synthesis. In *European Conference on Computer Vision*, pp. 311–327. Springer, 2018.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

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

Chaoyang Wang, Peiye Zhuang, Tuan Duc Ngo, Willi Menapace, Aliaksandr Siarohin, Michael Vasilkovsky, Ivan Skorokhodov, Sergey Tulyakov, Peter Wonka, and Hsin-Ying Lee. 4real-video: Learning generalizable photo-realistic 4d video diffusion. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17723–17732. IEEE, 2025a.

Fanyi Wang, Peng Liu, Haotian Hu, Dan Meng, Jingwen Su, Jinjin Xu, Yanhao Zhang, Xiaoming Ren, and Zhiwang Zhang. Loopanimate: Loopable salient object animation. In *Proceedings of the 6th ACM International Conference on Multimedia in Asia*, pp. 1–8, 2024a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306. IEEE, 2025b.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-ω. *arXiv preprint arXiv:2605.15195*, 2026.

Liao Wang, Jiakai Zhang, Xinhang Liu, Fuqiang Zhao, Yanshun Zhang, Yingliang Zhang, Minye Wu, Jingyi Yu, and Lan Xu. Fourier plenoctrees for dynamic radiance field rendering in real-time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13524–13534, 2022.

Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9660–9672. IEEE, 2025c.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20697–20709. IEEE, 2024b.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. In *International Conference on Learning Representations*, volume 2024, pp. 42055–42079, 2024c.

Yihan Wang, Lahav Lipson, and Jia Deng. Sea-raft: Simple, efficient, accurate raft for optical flow. In *European Conference on Computer Vision*, pp. 36–54. Springer, 2024d.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7465–7475. IEEE, 2020.

Thomas Wimmer, Michael Oechsle, Michael Niemeyer, and Federico Tombari. Gaussians-to-life: Text-driven animation of 3d gaussian splatting scenes. In *2025 International Conference on 3D Vision (3DV)*, pp. 958–968. IEEE, 2025.

World Labs. Marble: A multimodal world model. `https://www.worldlabs.ai/blog/marble-world-model`, November 2025. Official blog post.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20310–20320. IEEE, 2024.

Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26057–26068, 2025.

Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4389–4398. IEEE, 2024.

Yiming Xie, Chun-Han Yao, Vikram Voleti, Huaizu Jiang, and Varun Jampani. Sv4d: Dynamic 3d content generation with multi-frame and multi-view consistency. In *International Conference on Learning Representations*, volume 2025, pp. 33421–33441, 2025.

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pp. 399–417. Springer, 2024.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations*, volume 2024, pp. 9142–9159, 2024a.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20331–20341. IEEE, 2024b.

Vickie Ye, Ruilong Li, Justin Kerr, Matias Turkulainen, Brent Yi, Zhuoyang Pan, Otto Seiskari, Jianbo Ye, Jeffrey Hu, Matthew Tancik, et al. gsplat: An open-source library for gaussian splatting. *Journal of Machine Learning Research*, 26(34):1–17, 2025.

Mei-Chen Yeh and Po-Yi Li. An approach to automatic creation of cinemagraphs. In *Proceedings of the 20th ACM international conference on Multimedia*, pp. 1153–1156, 2012.

Heng Yu, Chaoyang Wang, Peiye Zhuang, Willi Menapace, Aliaksandr Siarohin, Junli Cao, Laszlo A Jeni, Sergey Tulyakov, and Hsin-Ying Lee. 4real: Towards photorealistic 4d scene generation via video diffusion models. *Advances in Neural Information Processing Systems*, 37:45256–45280, 2024a.

Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6658–6667. IEEE, 2024b.

Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5916–5926. IEEE, 2025a.

Mark Yu, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 100–111. IEEE, 2025b.

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19447–19456. IEEE, 2024c.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Forrester Cole, Deqing Sun, Ming-Hsuan Yang, et al. Monst3r: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations*, volume 2025, pp. 82863–82886, 2025.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Tianyuan Zhang, Hong-Xing Yu, Rundi Wu, Brandon Y Feng, Changxi Zheng, Noah Snavely, Jiajun Wu, and William T Freeman. Physdreamer: Physics-based interaction with 3d objects via video generation. In *European Conference on Computer Vision*, pp. 388–406. Springer, 2024.

Zhengyang Zhang, Ziyu Lu, PengCheng Li, Hongbo Duan, Yi Liu, Pengting Luo, Peiyu Zhuang, Xinghui Li, and Shaohua Ma. Fast-gs: Frequency aware space-time gaussian splatting for photorealistic dynamic novel view synthesis. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 9297–9301. IEEE, 2026.

Yuyang Zhao, Chung-Ching Lin, Kevin Lin, Zhiwen Yan, Linjie Li, Zhengyuan Yang, Jianfeng Wang, Gim H Lee, and Lijuan Wang. Genxd: Generating any 3d and 4d scenes. In *International Conference on Learning Representations*, volume 2025, pp. 95479–95499, 2025.

Shijie Zhou, Zhiwen Fan, Dejia Xu, Haoran Chang, Pradyumna Chari, Tejas Bharadwaj, Suya You, Zhangyang Wang, and Achuta Kadambi. Dreamscene360: Unconstrained text-to-3d scene generation with panoramic gaussian splatting. In *European Conference on Computer Vision*, pp. 324–342. Springer, 2024.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

APPENDIX OVERVIEW

This appendix complements the main paper with the following details:

- Appendix A describes how the 39 evaluation scenes are collected from Mip-NeRF 360, HY-World 2.0, Marble, and Lyra 2.0.
- Appendix B details each stage of our pipeline, including the VLM prompt, the video generation and lifting settings, and all hyperparameters of the Inconsistency-Robust Periodic 4DGS.
- Appendix C formulates the scene-view consistent optimization with diffusion-based refinement.
- Appendix D describes the baselines, how their inputs are derived from our pipeline, and their native loop periods.
- Appendix E provides the full Vividness breakdown and further details on the proposed ground-truth-free metrics.
- Appendix F describes the protocol and interface of our user study.

Rendered videos of all methods are provided in the supplementary material.

A DETAILS ON DATASET COLLECTION

Our benchmark contains 39 static 3DGS scenes: 9 reconstructed from Mip-NeRF 360 (Barron et al., 2022) and 10 generated by each of HY-World 2.0 (HY-World et al., 2026), Marble (World Labs, 2025), and Lyra 2.0 (Shen et al., 2026). Each scene is paired with a reference camera pose, from which the reference image I_{ref} is rendered and around which the evaluation trajectories (Section 4.3) are defined.

Scene selection. For the generated sources, we select scenes that (a) contain content with an obvious natural dynamic, such as water, foliage, cloth, fire, or light; (b) jointly span indoor, outdoor-natural, architectural, and stylized environments; and (c) include a clean, well-covered region large enough to support the $\pm 20^\circ$ orbit camera used in evaluation. The reference pose for each scene is manually selected within this region.

Mip-NeRF 360. We use all nine scenes from Mip-NeRF 360 and reconstruct each using the official 3DGS implementation (Kerbl et al., 2023) for 30,000 iterations with its default settings.

HY-World 2.0 and Marble. Both services host public galleries of generated worlds. We download ready-made scenes as Gaussian-splat files from the Marble platform and the Hunyuan 3D "Scene to 3D" platform and use them as-is without any regeneration or re-optimization.

Lyra 2.0. Lyra 2.0 does not host downloadable scenes; it is an image-to-video-to-3DGS pipeline that must be run locally. We therefore generate its scenes with the official code release, taking as input the 15 sample images and captions shipped with the repository, and keep 10 of the resulting scenes. Each sample is a 16:9 image paired with a handwritten caption describing a camera move through a static scene; we use the captions unchanged. Lyra 2.0 operates in two stages: a camera-controlled image-to-video model, fine-tuned from Wan 2.1 I2V-14B (Wan et al., 2025), synthesizes an exploration video at 832×480 and 16 FPS along a prescribed trajectory, and a feed-forward reconstructor, combining ViPE pose estimation (Huang et al., 2025a), Depth Anything 3 (Lin et al., 2025), and Gaussian prediction, lifts the video into 3DGS. We modify both stages as follows.

Exploration trajectory. The default trajectory, a short zoom-in and zoom-out along the optical axis, produces almost no lateral parallax. The reconstructed scene is then well covered only within a narrow cone around the input view, leaving holes under our $\pm 20^\circ$ orbit. We replace it with a dolly-then-orbit trajectory, implemented as an additional preset in the official inference code. The camera first pushes forward into the scene for about 10 s, and then orbits for about 20 s around a pivot placed at the depth center of the input view, sweeping to one side, back through the center, and to the other side while

keeping the pivot centered in the frame. The forward push places the camera well inside the scene, and the orbit provides a wide lateral baseline on both sides of the original optical axis. Each scene thus yields a 480-frame exploration video (30 s at 16 FPS).

Reconstruction. On these long, wide-baseline videos, the released feed-forward reconstructor produces blurry scenes with many floaters. We instead combine a 3D foundation model for poses and initial geometry with a conventional per-scene 3DGS optimization:

1. *Keyframe selection.* We select 64 keyframes per video. We first pick 48 frames at equal arc length in optical-flow space and move each to the sharpest frame within ± 3 frames. Since 2D flow under-weights forward dolly motion, we run VGGT-Ω (Wang et al., 2026) once on these frames, interpolate the recovered camera path over all frames, and re-select 64 keyframes at equal 3D arc length, where rotation is converted into an equivalent translation at the median scene depth.
2. *Initialization.* We run VGGT-Ω (1B parameters, 512 px input) on the 64 keyframes to obtain camera intrinsics, extrinsics, and per-pixel depth with confidence. We unproject the depth into a colored point cloud after discarding the 30% least-confident pixels and pixels straddling depth discontinuities, voxel-downsample it with a voxel size of 0.006 in the normalized scene units of VGGT, and export the poses and points as a COLMAP model.
3. *Optimization.* From this initialization, we optimize a standard 3DGS with gsplat (Ye et al., 2025) for 30k iterations, using an ℓ_1 + 0.2·D-SSIM photometric loss, SH degree 3, and adaptive density control stopped at 2,000 iterations. All 64 keyframes are used for training. When every 8th keyframe is held out, the reconstructions reach a mean novel-view PSNR of 22.1 dB and SSIM of 0.71.

B Details on Our Pipeline

Input preprocessing. The input scenes differ in their spherical-harmonics (SH) degree: HY-World 2.0 and Marble scenes are exported with SH degree 0, whereas Lyra 2.0 (reconstructed as in Appendix A) and Mip-NeRF 360 scenes use SH degree 3. We convert all scenes to SH degree 3 by zero-initializing the missing higher-order coefficients, so that the canonical color $\mathbf{c}$ shares one parameterization across sources.

B.1 Looping Video Generation

Dynamics prompt. GPT-5.5 (Singh et al., 2025) receives the reference image I_{ref} together with the instruction in Fig. 8. The instruction asks for clearly visible motion that may involve multiple elements rather than a single phenomenon, forbids introducing new content, or letting objects cross the camera. The returned prompt conditions the video generator and is reused as the text input of the baselines (Appendix D). We generate another video-generation prompt using the same GPT prompt to produce KVD reference videos (Appendix E).

Video generation. We query Seedance 2.0 (Seedance et al., 2026) with I_{ref} rendered at 1280×720 as both the first and last frames, thereby encouraging the video to return to its initial state. We generate 10-second videos at 720p and 24 FPS without audio, and set the aspect ratio to adaptive so that the output follows the conditioning image and requires no further cropping or reframing.

B.2 Dynamic Point Cloud Lifting

Inputs. We reconstruct the dynamic point cloud with a single forward pass of VGGT-Ω (Wang et al., 2026) on 45 images: 21 static multi-view renders at $t{=}0$ and 24 temporal frames of the reference video. The static renders are produced by the input 3DGS along a purely horizontal orbit, with azimuth ϕ sampled uniformly in $[-20^\circ, 20^\circ]$ at 2° intervals and zero elevation and radius offset, so the views are ordered monotonically. The central view ($\phi{=}0^\circ$) is the reference view, and the remaining 20 views coincide with the training viewpoints of Section B.3. For the temporal frames, we sample 25 sparse frames from V_{ref}. Since its first frame is I_{ref}, which is identical to the static render at $\phi{=}0^\circ$, we drop it and feed only the remaining 24 frames, all assigned to the reference view.

> You write the final English prompt for an image-to-video model.
>
> Look at the reference image and imagine the whole scene naturally coming alive. Describe plausible and obviously visible motion across the scene wherever it would make the image feel vivid, atmospheric, and lived-in. Multiple elements may move at the same time when that feels natural, and their motion may differ in direction, timing, and intensity. Do not limit the animation to a single object or a single motion phenomenon.
>
> The primary motion must remain unmistakably visible at normal playback speed, without requiring frame-by-frame comparison, while preserving the image hierarchy. Being subordinate to the composition does not mean being barely perceptible. Give the primary motion a strong, scene-appropriate amplitude reached gradually. Do not weaken the primary motion with stacked terms such as very light, barely moving, almost imperceptible, minimal, slight, and subtle.
>
> If nothing in the scene would naturally move, you can add lighting and shadow changes to make the scene alive.
>
> Do not add new things to the scene. Only move the existing things in the scene. Do not let anything pass into the foreground from behind the camera. Do not let anything pass the camera, move behind it, and disappear from the video.
>
> Keep the subjects, environment, visual identity, and overall composition recognizable. The result should feel as if the photographed moment has come to life, not like an unrelated scene or a new narrative event. Keep the camera fixed and make the motion flow naturally as a seamless loop.
>
> Write one concise, concrete prompt for the video-generation model. Describe what moves and how the scene feels in motion. Output only the final prompt in English, without analysis, headings, bullet points, JSON, Markdown, or quotation marks.

Figure 8: **Instruction given to GPT-5.5** together with the reference image to write the dynamics prompt.

Preprocessing. Each image is center-cropped to an aspect ratio within $[0.5, 2.0]$, resized so that its longer side is 512 pixels, and aligned to the patch size of 16, e.g., 512×288 for a 16:9 input. All 45 images share the same crop and resize, so the reference intrinsics are transformed only once and apply to every input.

Outputs. We do not use the camera poses predicted by VGGT-Ω; all cameras are taken from the input 3DGS, and the predicted poses serve only as a sanity check. We remove the 10% least-confident points of each frame and align the resulting dynamic point cloud to the coordinate frame of the input 3DGS with an affine transformation.

B.3 Multi-View Video Generation

Cameras. For each scene, we manually select a pivot $\mathbf{p}$ by back-projecting a pixel of the reference view. The $N{=}20$ training cameras face $\mathbf{p}$ and rotate within $\pm 20^\circ$ about the axis through $\mathbf{p}$ parallel to the yaw axis of the reference camera, at radius $r = \|\mathbf{o}_{\text{ref}} - \mathbf{p}\|$, where $\mathbf{o}_{\text{ref}}$ is the reference camera center. The orbit camera used for evaluation (Section 4.3) shares the same pivot but uses a radius of $0.95r$, thereby probing viewpoints off the training trajectory.

Video inpainting. We render the aligned dynamic point cloud from each training camera to obtain incomplete videos, which TrajectoryCrafter (Yu et al., 2025b) completes at 672×384 with 50 sampling steps and a guidance scale of 6.0 (seed 43). All target views share the same initial noise, which reduces cross-view appearance drift.

B.4 Inconsistency-Robust Periodic 4DGS

Architecture. The Periodic Deformation Field $\mathcal{P}$ uses a single-scale triplane with a resolution of 64 per axis. The triplane feature is split into Fourier coefficients with $K{=}4$ frequencies (Eq. (1)), and the time-dependent feature is decoded by an MLP of width 128 with separate heads for the offsets of position, rotation, scale, and SH coefficients. The Grounded Drift Field Δ uses the same architecture and heads, and conditions on the time embedding $\gamma(t)$ and the per-view embedding $\mathbf{e}_v$ instead of the periodic feature. A single triplane $\mathbf{F}_\Delta$ is shared across all non-reference views, and views are distinguished only by $\mathbf{e}_v$. We share it because we observe that the inconsistencies

Table 4: **Hyperparameters of OuroWorld.**

Stage	Hyperparameter	Value
Looping video	VLM / video generator	GPT-5.5 / Seedance 2.0
	Conditioning image resolution	1280×720
	Duration / frame rate	10 s / 24 FPS
Lifting	Static renders / temporal frames	21 / 24
	Input resolution	512×288
	Discarded low-confidence points	10% per frame
Multi-view video	Number of views N	20
	Yaw range	$\pm 20^\circ$
	Inpainting resolution	672×384
	Sampling steps / guidance scale	50 / 6.0
IRP-4DGS	Triplane resolution ($\mathcal{P}$ and Δ)	64
	Temporal bandwidth K	4
	MLP width	128
	Loop period T	10s
	Loss weights λ_1 / λ_p	1 / 0.2
	Iterations	per scene

produced by TrajectoryCrafter are correlated across views, i.e., independently inpainted views tend to exhibit similar artifacts, so a shared spatial feature explains them with fewer parameters than per-view features.

Hyperparameters. Table 4 summarizes the hyperparameters of the full pipeline. The number of optimization iterations is adjusted per scene.

C Details on Scene-View Consistent Optimization

We detail the objective of Section 3.4, which follows the consistent 4D-GS optimization of Free4D (Liu et al., 2025). Let $\hat{I}_{t,v}$ denote the rendering of $\mathcal{G}_{t,v}$ (Eq. (2)) at time t and view v, and let $I_{t,v}$ denote the corresponding supervision frame.

Consistent supervision. Two sets of frames are highly consistent with the input scene: the reference video V_{ref}, which comes directly from the video model, and the $t{=}0$ renders of the input 3DGS at all training views, which coincide with the scene at the start of the loop since V_{ref} begins at I_{ref}. We denote their union by $\mathcal{S}$ and supervise it at the pixel level:

$$\mathcal{L}_1 = \sum_{(t,v)\in\mathcal{S}} \big\|\hat{I}_{t,v} - I_{t,v}\big\|_1. \tag{5}$$

Diffusion-based refinement. The remaining generated frames, $(t, v) \notin \mathcal{S}$, are less reliable, so we do not use them as pixel-level targets. Instead, we use them to guide the refinement of the current rendering. We encode $\hat{I}_{t,v}$ into a latent with the encoder $\mathcal{E}$ of Stable Diffusion v1.5 (Rombach et al., 2022), perturb it with noise and denoise it, similar to SDEdit (Meng et al., 2021). At each denoising step i, the model predicts a clean latent $\hat{\mathbf{z}}_{0|i}$, which we modulate toward the generated frame $\mathbf{z}^g = \mathcal{E}(I_{t,v})$:

$$\tilde{\mathbf{z}}_{0|i} = w_i\,\gamma_i\,\mathbf{z}^g + (1 - w_i)\,\hat{\mathbf{z}}_{0|i}, \qquad \gamma_i = \frac{\mathrm{std}(\hat{\mathbf{z}}_{0|i})}{\mathrm{std}(\mathbf{z}^g)}, \tag{6}$$

where $w_i \in \{0.5, 0.4, 0.3, 0.2, 0.1\}$ controls the influence of the generated frame, linearly decreasing over the $i = 1, \ldots, 5$ denoising steps, and γ_i matches the latent statistics to avoid over-exposure. The modulated $\tilde{\mathbf{z}}_{0|i}$ replaces $\hat{\mathbf{z}}_{0|i}$ in the subsequent denoising step, and the final latent is decoded into the refined image $\tilde{I}_{t,v}$. The refined image inherits the dynamics of the generated frame while staying close to the current scene, so the inconsistencies of the generated frame are not transferred at the pixel level. We supervise the rendering with a perceptual loss:

$$\mathcal{L}_p = \sum_{(t,v)\notin\mathcal{S}} \mathrm{LPIPS}\left(\hat{I}_{t,v}, \tilde{I}_{t,v}\right), \tag{7}$$

where $\tilde{I}_{t,v}$ is treated as a constant. For non-reference views, both losses are computed on renderings that include the drift Δ_v, following Eq. (2).

Schedule. Since the refinement relies on a reasonable rendering, we first optimize with $\mathcal{L}_1$ alone, then with the full objective $\mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_p \mathcal{L}_p$. The hyperparameters are listed in Table 4.

D Details on Baselines

D.1 Methods

Gaussians-to-Life (Wimmer et al., 2025) takes as input a 3DGS, a 3D mask, and a text prompt. It uses a video diffusion model (Blattmann et al., 2023) as the motion prior, lifts the generated 2D motion to 3D point flow via point tracking (Karaev et al., 2024) and depth estimation, and uses this flow to drive the Gaussians. It does not aim to produce looping motion; for all period-dependent evaluation, we treat the length of its generated motion as its period.

3D Cinemagraphy (Li et al., 2023) takes as input a single image and a 2D mask. It predicts motion with an Eulerian motion model trained on fluid videos (Holynski et al., 2021), lifts the image to a layered depth image (LDI) (Shade et al., 1998; Shih et al., 2020), and produces a loop by symmetrically splatting the LDI features forward and backward in time.

3D-MOM (Jin et al., 2025) takes as input a single image and a 2D mask. It back-projects the 2D Eulerian motion into a 4D Gaussian representation (Wu et al., 2024). Its 2D motion prior loops through symmetric splatting, but the optimized 4D representation is not guaranteed to loop.

LoopGaussian (Li et al., 2024a) takes as input a 3DGS and a motion mask. It heuristically moves Gaussians toward positions with similar features and forms a loop by bidirectionally blending an Eulerian motion field. It updates only the Gaussian positions.

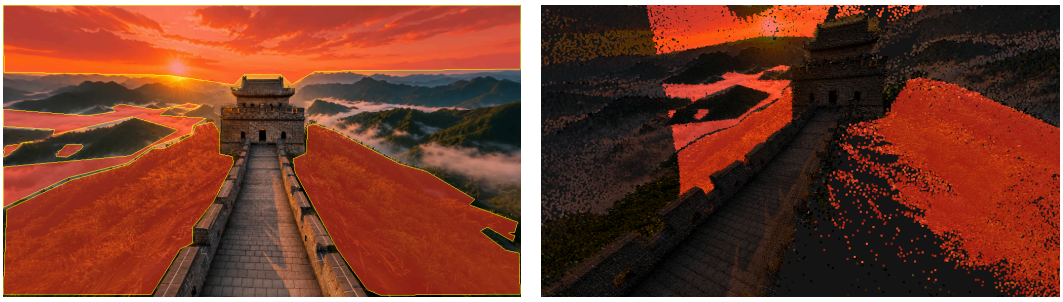
For fairness, we also apply the same GPT-5.5 filtering for our method to all baselines.

D.2 Input Alignment

All baselines require inputs beyond a static 3DGS. We derive them from our pipeline so that the same scene, view, and motion description drives every method. Input images are rendered from the input 3DGS at the reference pose, and text prompts are the GPT-5.5 dynamics prompts of Section B.1.

2D motion mask. We derive the 2D motion mask from the dynamics prompt rather than from the generated videos, so that it reflects the intended motion instead of any single method's output (Fig. 9a). A deterministic, rule-based parser first extracts the moving entities from the prompt, i.e., nouns paired with a motion verb such as *river*, *trees*, or *grass*, together with any entities the prompt explicitly marks as static. Because the parser is rule-based rather than LLM-based, the extracted entities are fully reproducible. Each entity phrase is then segmented separately on the reference image with SAM 3 (Carion et al., 2026) using text-concept segmentation, since short noun phrases give more reliable concept masks than the full prompt. We filter the predictions with the following rules: (i) we keep instances with a confidence score of at least 0.45 to suppress false positives; (ii) we discard any instance covering more than 85% of the frame, which typically indicates a degenerate prediction that selects the whole image; (iii) we discard any entity whose union covers more than 90% of the frame, as such a mask cannot separate moving from static regions; and (iv) we remove connected components smaller than 0.08% of the image to eliminate speckle noise. We then subtract the masks of the static entities, which handles cases where a static object overlaps a moving region, such as a bridge spanning a river. The union of the remaining masks forms the 2D motion mask. Finally, every mask is verified by a human before it is used for evaluation.

3D motion mask. Among the baselines, only Gaussians-to-Life requires a 3D motion mask; all others use the 2D mask directly. For it, we lift the 2D motion mask to 3D (Fig. 9b). We project every Gaussian center into the reference camera and select the Gaussians that fall inside the 2D mask, lie in front of the camera, and have depth within $[d_{\min}, d_{\max}]$. This interval is set automatically from



(a) 2D motion mask (b) 3D motion mask

Figure 9: **Motion masks.** (a) The 2D motion mask obtained by segmenting the moving entities in the reference image with SAM 3. (b) The corresponding 3D motion mask.

Table 5: **Inputs and native loop periods of all methods.** For Gaussians-to-Life, which does not loop, the period is the length of its generated motion.

Method	Scene input	Motion input	Native period
Gaussians-to-Life	3DGS	3D mask + text	0.5 s
3D Cinemagraphy	Single image	2D mask	2.4 s
3D-MOM	Single image	2D mask	2.0 s
LoopGaussian	3DGS	2D mask	2.0 s
Ours	3DGS	None	10 s

the 5th and 95th percentiles of the in-mask Gaussian depths, padded by 10% on each side, which excludes background Gaussians that project into the mask from behind the moving entity. As with the 2D mask, the selection is verified by a human; in a typical scene, it contains 14% of the Gaussians.

D.3 Native Periods

As described in Section 4.3, we keep the native period of each method and repeat it to fill the 20-second evaluation video. Table 5 summarizes the inputs and native periods of all methods.

E Details on Evaluation

E.1 Vividness

Thresholds. Each rendered video is temporally downsampled from 30 to 3 FPS, and the three components are computed between adjacent frames. The thresholds of Motion Variation and Illumination Variation, i.e., a flow magnitude of $0.3\% \times \min(H, W)$ (about 1.2 pixels at 672×384) and a luminance change of 20%, were chosen empirically. We visually inspected which regions appear to move or change in illumination, tested several candidate thresholds, and selected the pair whose masks best agreed with human perception.

Full results. Tables 6 and 7 report each component of the Vividness Degree for the comparison and the ablation, respectively. Under the static camera, Gaussians-to-Life attains the highest Motion and Visual Variation, because it loops over only 0.5 s, and its low MALF (Table 2) shows that this motion does not form a coherent loop. Our method obtains by far the highest Illumination Variation, which reflects the illumination changes that Eulerian baselines cannot express. Under the orbit camera, Motion Variation is dominated by the camera motion. It is nearly identical across methods, so the differences mainly come from Illumination Variation, where our method again leads. In the ablation, removing grounding primarily reduces Motion Variation (0.0413 $\rightarrow$ 0.0324 under the static camera), confirming that ungrounded drift absorbs object motion. Removing scene-view consistent optimization slightly increases the static-camera components, which we attribute to inpainting inconsistencies baked into the scene that flicker over time rather than to meaningful dynamics; this variant also shows the lowest consistency and sharpness in Table 3.

Table 6: **Vividness breakdown against baselines.** MV, IV, and VV denote Motion, Illumination, and Visual Variation; Vividness is their average. Bold denotes the best result.

	Static camera				Orbit camera			
Method	MV ↑	IV ↑	VV ↑	Vividness ↑	MV ↑	IV ↑	VV ↑	Vividness ↑
Gaussians-to-Life	**0.0588**	0.0170	**0.0364**	0.0374	0.9412	0.0837	0.5044	0.5097
3D Cinemagraphy	0.0045	0.0069	0.0086	0.0066	0.9348	0.0765	0.4722	0.4945
LoopGaussian	0.0021	0.0025	0.0027	0.0024	0.9390	0.0823	**0.5047**	0.5087
3D-MOM	0.0234	0.0248	0.0259	0.0247	0.9114	0.0970	0.4089	0.4724
Ours	0.0413	**0.0518**	0.0338	**0.0423**	**0.9421**	**0.1146**	0.4784	**0.5117**

Table 7: **Vividness breakdown of the ablation.** SVCO denotes scene-view consistent optimization. Bold denotes the best result.

	Static camera				Orbit camera			
Method	MV ↑	IV ↑	VV ↑	Vividness ↑	MV ↑	IV ↑	VV ↑	Vividness ↑
w/o Drift	0.0414	0.0519	0.0340	0.0424	0.9420	**0.1151**	0.4768	0.5113
w/o Grounding	0.0324	**0.0528**	0.0323	0.0392	**0.9421**	0.1144	0.4676	0.5080
w/o SVCO	**0.0442**	0.0450	**0.0454**	**0.0450**	0.9409	0.1039	**0.4795**	0.5081
Ours	0.0413	0.0518	0.0338	0.0423	**0.9421**	0.1146	0.4784	**0.5117**

E.2 Naturalness

For each scene, we generate 20 videos with MiniMax H3 (MiniMax, 2026) using its default settings, conditioned on the reference image and the GPT-5.5 dynamics prompt, for a total of 780 videos. For each method, we compute KVD (Unterthiner et al., 2018) between these videos and the method's 78 rendered videos (39 scenes $\times$ 2 trajectories). Videos are split into non-overlapping 10 s clips, each sampled to 60 frames at 6 FPS and 224×224, and then encoded into 400-d features using the Kinetics-400 I3D network for FVD. Following KID (Bińkowski et al., 2018), we use the unbiased MMD^2 estimator with kernel $k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^\top \mathbf{y}/D + 1)^3$, averaged over 100 random subsets of up to 100 clips. Since static camera videos are rendered from a fixed camera, whereas orbit videos include camera motion, KVD is comparable across methods within each camera setting but not across settings.

E.3 Loop Seam Coherence

Motion-Aware Loop Fidelity. Let I_t be the frame at time t under the static camera and T the native period of the method. We sample $M{=}10$ time points $\{t_i\}_{i=1}^{M}$ uniformly within one period and evaluate Eq. (4). The first term, $\mathrm{SSIM}(I_{t_i}, I_{t_i+T})$, measures whether the scene returns to the same state one period later. The second term averages the similarity between I_{t_i} and the other phases in the next cycle, thereby measuring how similar the frames are regardless of phase. Their difference is therefore high only when the motion is substantial and phase-aligned across cycles. A static video has all similarities equal to one and receives MALF$=0$. A video that moves but does not return to its initial state has a first term no larger than the second and also receives a near-zero score. Unlike Seam SSIM, MALF therefore cannot be improved by suppressing motion.

Seam SSIM. Seam SSIM is the SSIM between the last frame of one cycle and the first frame of the next, i.e., between $I_{T-1/30}$ and I_T at 30 FPS. It directly measures the visible jump at the loop boundary, but is trivially high for near-static videos, which is why we report it together with MALF.

F Details on User Study

Protocol. We conduct a two-alternative forced-choice study. Each trial shows two rendered videos of the same scene side by side, one from our method and one from a baseline, with the left-right order randomized. Participants answer four questions per trial, one for each criterion:

- *Vividness*: Which scene looks the most vivid?
- *Naturalness*: Which scene looks the most natural?

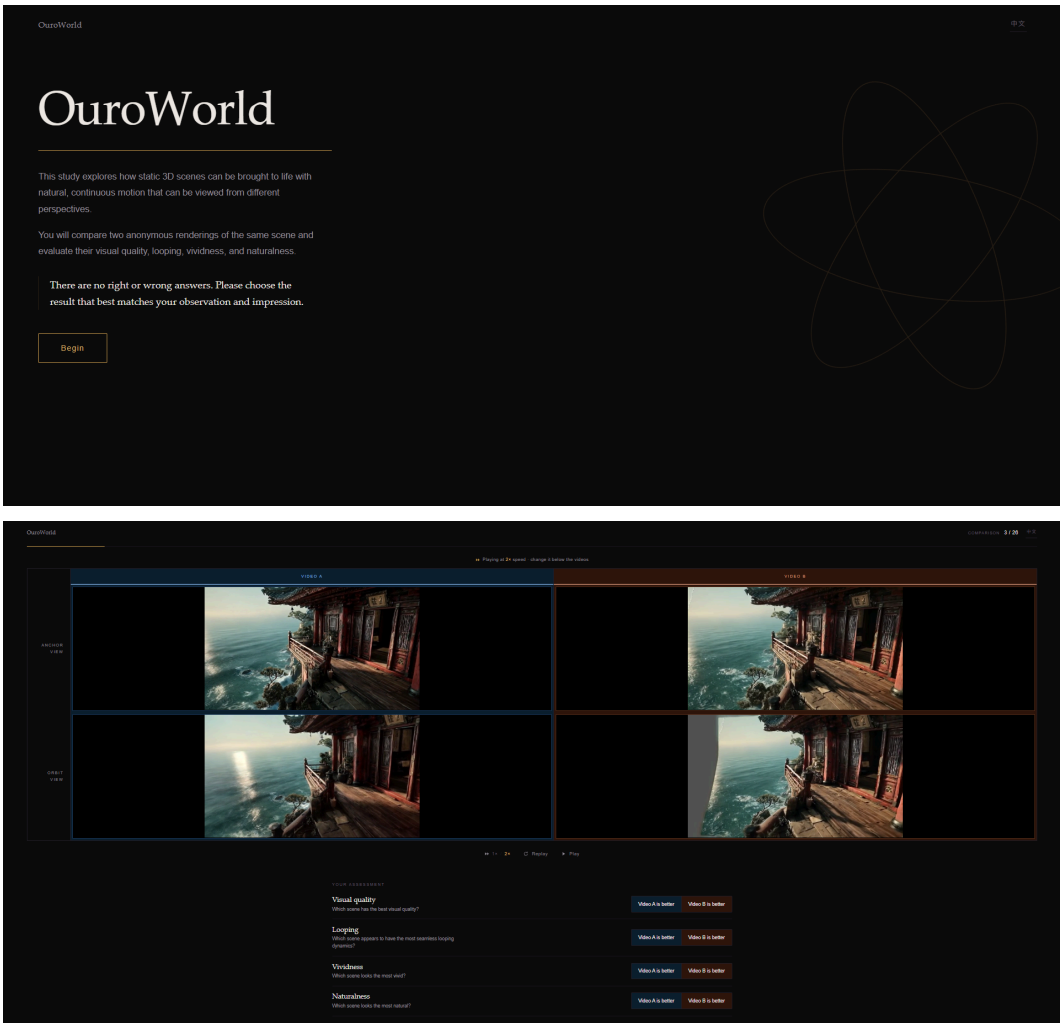


Figure 10: **User study interface.** Two videos of the same scene are shown side by side in random order (top), and participants choose one for each of the four criteria (bottom).

- *Loop Seam Coherence*: Which scene appears to have the most seamless looping dynamics?
- *Visual Quality*: Which scene has the best visual quality?

The interface is shown in Fig. 10.

Statistics. The study covers all 39 scenes and 4 baselines, i.e., 156 video pairs. Each pair is judged by 5 different participants, and each of the 30 participants judges 26 pairs, resulting in 780 trials in total. Each criterion thus receives 780 votes, 195 per baseline, from which the preference rates in Fig. 6 are computed. All participants are adult volunteers; participation was voluntary and anonymous, and no personally identifiable information was collected.