# PanoWorld: Geometry-Consistent Panoramic Video World Modeling

**Le Jiang** **Xiangyu Bai** **Bishoy Galoaa** **Shayda Moezzi** **Caleb James Lee**

**Tooba Imtiaz** **Edmund Yeh** **Jennifer Dy** **Yanzhi Wang**

**Sarah Ostadabbas**[†]

{jiang.l, bai.xiang, galoaa.b, moezzi.s, lee.calebj,
imtiaz.t, e.yeh, j.dy, yanzhiwang, s.ostadabbas}@northeastern.edu

*Northeastern University*




(a) Input (b) Panoramic Video Generation (c) 3D Reconstruction (d) Novel View

Figure 1: **PanoWorld** generates geometry-consistent 360° panoramic video from a single perspective image and text prompt. Unlike prior panoramic video methods that optimize for visual realism alone, PanoWorld enforces depth and trajectory consistency in the latent world state, so that its output can also be lifted to 3D (qualitative illustration): (a) input perspective image, (b) generated 360° panoramic video, (c) point cloud lifted from the generated video, and (d) rendering from a nearby camera pose unseen during generation.

## Abstract

We present **PanoWorld**, a panoramic video world model that generates geometry-consistent 360° video from a single image and a caption. Existing panoramic video methods optimize primarily for visual realism and do not explicitly constrain the underlying 3D scene state, producing outputs that appear plausible yet exhibit inconsistent depth, broken correspondences, and implausible motion across the spherical surface. We address this gap by framing panoramic video generation as a geometry- and dynamics-consistent latent state modeling problem rather than pure visual synthesis. Building on a pre-trained perspective video world model, we introduce two lightweight regularizers: a depth consistency loss against pseudo ground-truth panoramic depth, and a trajectory consistency loss that supervises the 3D world-frame positions of tracked points across time. We further apply spherical-geometry-aware adaptation to the conditioning and positional encoding. We additionally introduce **PanoGeo**, a unified geometry-aware

[†]Corresponding author: s.ostadabbas@northeastern.edu

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

panoramic video dataset with consistent depth, trajectory, and prompt annotations across diverse real and synthetic sources, used for both training and stratified evaluation. Experiments show that **PanoWorld** improves geometric consistency over prior panoramic generation methods while maintaining competitive visual realism, suggesting that explicit geometric supervision is a practical step toward panoramic world models for embodied AI applications. Code is available at `https://github.com/ostadabbas/PanoWorld`.

## 1 Introduction

Video world models [Agarwal et al., 2025, Dawid and LeCun, 2024] have recently emerged as a promising paradigm for learning structured representations of dynamic physical environments. Rather than modeling video purely as appearance sequences, these systems seek internal state representations that support prediction, simulation, and planning [Hafner et al., 2019a, Bruce et al., 2024, Russell et al., 2025, Zhu et al., 2025a]. By training on large-scale video corpora, such models aim to capture scene dynamics, geometry, and semantics, enabling applications in autonomous driving, robotic control, and video understanding [Huang et al., 2025a, Zhu et al., 2025b, Hong et al., 2023, Chen et al., 2025b, Chi et al., 2025]. A central question, however, is what meaningfully distinguishes a video world model from a high-capacity video generator.

Conventional video generation models typically learn conditional distributions over pixel sequences, either as joint denoising $p(x_{1:T})$ [Blattmann et al., 2023, Bar-Tal et al., 2024, Zheng et al., 2024, Brooks et al., 2024] or as autoregressive prediction $p(x_{t+1} \mid x_{\leq t})$ [Hong et al., 2023, Kondratyuk et al., 2023]. By contrast, video world models can be viewed as learning latent dynamical systems:

$$s_{t+1} = f_\theta(s_t, a_t), \quad x_t = g_\phi(s_t), \tag{1}$$

where $s_t \in \mathbb{R}^d$ denotes a latent scene state encoding geometry, object configurations, and physical factors, $a_t$ denotes the conditioning input (the text prompt in PanoWorld), $f_\theta$ is a learned transition model, and $g_\phi$ maps latent states to observations. Video is then generated through state evolution and rendering rather than direct frame-wise synthesis, and structured latent transitions are particularly important when downstream tasks require persistent spatial understanding rather than short-horizon realism [Wu et al., 2024].

Despite recent progress, many current video world models [Agarwal et al., 2025, Blattmann et al., 2023, Hong et al., 2023, World Labs, 2025] still operate in the limited field of view of standard perspective cameras. As a result, the latent state at time $t$ is learned from only a partial observation of the surrounding world:

$$x_t = g_\phi(s_t, \Omega_t), \quad \Omega_t \subset \mathbb{S}^2, \tag{2}$$

where $\Omega_t$ denotes the visible camera frustum on the viewing sphere $\mathbb{S}^2$ of all directions. This induces a partially observable setting in which critical scene content may lie outside the current view, complicating tasks that require holistic environmental awareness such as immersive telepresence [Huang et al., 2025b, Li et al., 2026b], navigation [Zeng et al., 2025, Li et al., 2026a], and robotic manipulation [Bu et al., 2025, Routray et al., 2025]. Multi-view fusion across narrow-view observations is one workaround [Mildenhall et al., 2021, Chang et al., 2025], but persistent identity, geometry, and motion consistency are difficult to maintain across long-range motion or in a shared coordinate frame [Macario Barros et al., 2022, Qin and Yu, 2023, Gao et al., 2022].

Panoramic (equirectangular) video offers an attractive alternative by directly modeling the full observation domain ($\Omega_t = \mathbb{S}^2$). A single panoramic frame captures the surrounding 360° scene in one global coordinate system, enabling unified reasoning over spatial relationships and motion across all directions. Although panoramic representations introduce projection distortions, they substantially reduce viewpoint truncation and fragmented scene state estimation.

Existing panoramic video methods [Luo et al., 2025, Tan et al., 2025, Fang et al., 2025, Wang et al., 2024] remain primarily optimized for visual realism, minimizing appearance-level losses between generated and ground-truth panoramic sequences without explicitly constraining latent state consistency across time or viewpoints. Generated panoramas thus appear realistic while exhibiting inconsistent depth, broken correspondences across adjacent regions, or implausible motion trajectories on the sphere, failure modes that block full-sphere world modeling for embodied applications.

In this work, we propose **PanoWorld**, a panoramic video world model that formulates 360° video generation as a geometry- and dynamics-consistent state modeling problem rather than pure visual synthesis. Starting from a pre-trained video world model, we explicitly regularize latent state evolution across time and viewpoints using complementary supervisory signals: appearance fidelity, 3D geometric consistency via depth supervision [Lin et al., 2025], and motion consistency via trajectory alignment [Karaev et al., 2024]. This shifts panoramic generation from frame matching toward structured world-state prediction in a unified spherical coordinate space. Because depth is supervised jointly with appearance, the generated video can also be lifted to a dense 4D point cloud and rendered from nearby viewpoints, which we show qualitatively in Figure 1 and Appendix F.

Our contributions are three-fold:

- We introduce PanoWorld, a panoramic world-modeling framework that adapts a pre-trained perspective video world model to full $360^\circ$ spherical observations, recovering the panoramic state outside the initial camera frustum from a single partial observation while enforcing geometric and temporal consistency over the generated clip.
- We propose two lightweight latent regularizers beyond visual similarity: (i) a *depth consistency loss* that promotes multi-view geometric coherence, and (ii) a *trajectory consistency loss* that encourages temporally plausible motion. Together, these objectives induce a more structured latent space with minimal added computational overhead.
- We introduce PanoGeo, a unified geometry-aware panoramic video dataset that mixes self-captured 4K 360° video, public panorama datasets [Wang et al., 2024, Chen et al., 2024], and Habitat-rendered simulations [Puig et al., 2023] across diverse indoor and outdoor scenes, all processed by a single annotation pipeline that produces consistent depth, dense 2D trajectories with camera-motion compensation, and prompts. The same pipeline also yields a stratified evaluation split spanning in-domain held-out, real out-of-domain, and synthetic out-of-domain regimes (Sec. 4.1, full construction details in the Appendix A).

## 2 Related Work

**Video generation and world models.** Diffusion transformers [Peebles and Xie, 2023] trained on large-scale video corpora produce realistic motion and appearance [Blattmann et al., 2023, Hong et al., 2023, Kondratyuk et al., 2023, Bar-Tal et al., 2024, Brooks et al., 2024, Zheng et al., 2024], but high perceptual realism does not imply persistent state modeling and these systems often struggle with long-horizon consistency and physically grounded reasoning [Bai et al., 2026, Bansal et al., 2024]. Video world models [Ha and Schmidhuber, 2018, Dawid and LeCun, 2024, Hafner et al., 2019a,b, Bruce et al., 2024, Russell et al., 2025, Zhu et al., 2025a, Agarwal et al., 2025] unify generation with structured latent dynamics for prediction, simulation, and planning. A complementary direction targets static 3D scenes via NeRF [Mildenhall et al., 2021], 3D Gaussian splatting [Kerbl et al., 2023], or generative scene priors [World Labs, 2025, Huang et al., 2025b, Li et al., 2026b, Team HunyuanWorld, 2025, Wang et al., 2026], capturing space without time. Most existing video world models still operate from perspective frustums $\Omega_t \subset \mathbb{S}^2$ [Zhu et al., 2025a, Li et al., 2025, Bardes et al., 2024], leaving holistic full-sphere reasoning open.

**Panoramic video generation.** Panoramic video exposes the full spherical observation domain ($\Omega_t = \mathbb{S}^2$) in a single coordinate frame and is naturally suited to immersive telepresence, robotics, and scene-centric world modeling. Recent panoramic generators include 360DVD [Wang et al., 2024] (controllable dual-branch diffusion), ARGUS [Luo et al., 2025] (perspective-conditioned SVD with reference latents and CLIP features), Imagine360 [Tan et al., 2025] (perspective-anchor 360° synthesis), and ViewPoint [Fang et al., 2025] (motion control). A wide-field outpainting line includes Follow-Your-Canvas [Chen et al., 2025a] and OmniRoam [Liu et al., 2026], which couples a perspective generator with a long-horizon trajectory module. These methods mainly optimize appearance metrics such as FID [Heusel et al., 2017], FVD [Unterthiner et al., 2019], and LPIPS [Zhang et al., 2018], and the latent geometry and temporal coherence of the full sphere are left unconstrained.

**Geometry-aware generative modeling.** Structural priors have been injected into generative models via conditioning signals (ControlNet [Zhang et al., 2023] and follow-ups [Bai et al., 2025]),

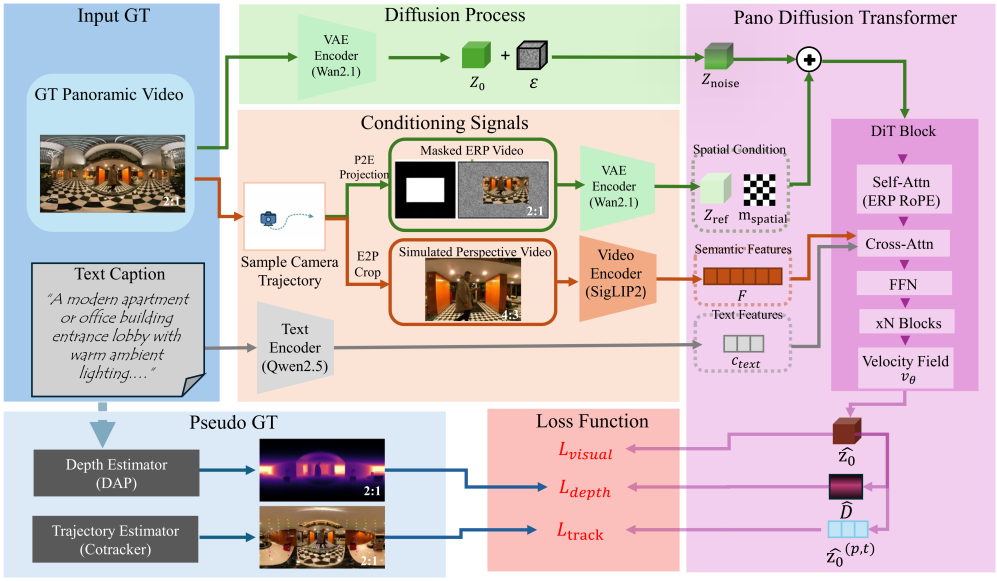



Figure 2: **Training pipeline overview.** A ground-truth panoramic video from PanoGeo (blue, left) is VAE-encoded into a clean latent $\mathbf{z}_0$ and mixed with noise $\boldsymbol{\epsilon}$ to form $\mathbf{z}_{\text{noise}}$ (green, top). Three conditioning signals (orange, center) are derived from a randomly sampled perspective crop: (1) *spatial*: the crop is projected onto the equirectangular canvas (P2E) and VAE-encoded into reference latent $\mathbf{z}_{\text{ref}}$ and mask $\mathbf{m}_{\text{spatial}}$, concatenated with $\mathbf{z}_{\text{noise}}$. (2) *semantic*: frozen SigLIP2 encodes the crop into per-frame features $\mathbf{F}$, injected via cross-attention. (3) *text*: caption features $\mathbf{c}_{\text{text}}$, also via cross-attention. The Pano Diffusion Transformer (purple, right) processes $\mathbf{z}_{\text{input}}=[\mathbf{z}_{\text{noise}};\mathbf{z}_{\text{ref}};\mathbf{m}_{\text{spatial}}]$ through $N$ DiT blocks with ERP-aware RoPE and predicts velocity $v_\theta$, from which $\hat{\mathbf{z}}_0$ is recovered. Three losses supervise $\hat{\mathbf{z}}_0$: $\mathcal{L}_{\text{visual}}$ (rectified-flow target), $\mathcal{L}_{\text{depth}}$ (Eq. 4, depth head vs. Depth Any Panoramas (DAP) pseudo-labels [Lin et al., 2025]), and $\mathcal{L}_{\text{track}}$ (Eq. 6, 3D point consistency of CoTracker3 [Karaev et al., 2024] tracks lifted via the predicted depth). All pseudo-labels (blue, bottom left) are precomputed offline at zero training-time cost. See Sec. 3 for details.

correspondence-aware attention [Tang et al., 2023], epipolar constraints [Watson et al., 2022], and spherical convolutions [Coors et al., 2018]. A parallel direction fits explicit 3D / 4D representations [Wu et al., 2025, Pan et al., 2025] or feed-forward reconstructors [Wang et al., 2025, Tang et al., 2025]. These approaches target image generation, discrete view synthesis, or post-hoc 3D reconstruction, and do not regularize latent state dynamics during panoramic video rollout. PanoWorld bridges this gap by treating panoramic video generation as a state-space modeling problem, combining full-sphere observations, latent temporal dynamics, and geometry-aware regularization in a unified training framework.

## 3 Introducing PanoWorld

We learn a panoramic video world model that predicts full-sphere observations from a partial perspective view while preserving coherent geometry and motion. Building on the formulation in Sec. 1, our approach combines (i) spherical-geometry-aware adaptation of a pretrained perspective video world model, (ii) multi-path conditioning from the partial observation, and (iii) two geometry-aware auxiliary objectives – a depth consistency regularizer $\mathcal{L}_{\text{depth}}$ and a trajectory consistency regularizer $\mathcal{L}_{\text{track}}$ that augment the rectified-flow visual loss $\mathcal{L}_{\text{visual}}$ via pseudo ground-truth depth and trajectories from PanoGeo (full objective: Sec. 3.4).

### 3.1 Training Pipeline Overview

We build on Cosmos Predict 2.5 [Agarwal et al., 2025], a rectified-flow [Liu et al., 2023] DiT video world model with a WAN 2.1 VAE [Wan et al., 2025] (encoder $\mathcal{E}$, $8\times$ spatial / $4\times$ temporal compression) and a frozen text encoder; off-the-shelf component versions are listed in Appendix C.

Rather than training a panoramic model from scratch, our goal is to transfer this perspective prior into the spherical domain with minimal additional supervision.

Figure 2 illustrates the training pipeline. Given a panoramic video from PanoGeo, we sample a random perspective trajectory (FOV 30°–120°, randomized yaw/pitch) that simulates a conventional camera observing a subset of the scene, turning training into a state-completion problem: from partial perspective evidence the model must reconstruct the full-sphere world state and its evolution. The sampled crop conditions the DiT through three complementary pathways (Sec. 3.3): SigLIP2 cross-attention, an equirectangular reference latent with spatial coverage mask, and masked-blending preservation of observed regions. The clean-latent estimate is supervised by geometry-aware losses (Sec. 3.4): a lightweight depth head maps it to a depth map against DAP pseudo-labels [Lin et al., 2025], and tracked points lifted to 3D through that predicted depth are supervised against precomputed GT 3D positions, jointly constraining depth head and denoiser to produce a temporally coherent scene geometry. Training proceeds progressively at $256\times512$ and then $512\times1024$, with geometry losses active throughout.

### 3.2 Equirectangular Adaptation

Equirectangular projection (ERP) maps longitude $\lambda \in [-\pi, \pi]$ and latitude $\phi \in [-\pi/2, \pi/2]$ on the viewing sphere onto a non-uniform 2D image, with horizontal pixel stretching near the poles proportional to $1/\cos\phi$. Standard linear positional embeddings and pixel-uniform losses therefore distort attention and gradients in ways inconsistent with the underlying sphere. We align the model with spherical geometry through three modifications.

**Latitude-aware positional embeddings.** We replace the linear RoPE coordinate $\text{pos}_h = h$ with a latitude-aware position that compresses indices near the poles, where equal pixel increments correspond to smaller angular displacements on the sphere:

$$\text{pos}_h = \frac{H-1}{2}\left(\sin\left(\frac{\pi h}{H-1} - \frac{\pi}{2}\right) + 1\right), \qquad \text{pos}_w = w. \tag{3}$$

Equivalently $\text{pos}_h \propto \sin\phi(h)+1$ where $\phi(h) = \pi h/(H-1) - \pi/2$ is the latitude of row $h$, matching the true angular spacing of ERP rows. Longitude is left linear since ERP preserves uniform angular spacing along the width axis.

**Spherical area weighting.** Because ERP pixels near the poles represent smaller solid angles on the sphere than equatorial pixels, unweighted pixel-space losses over-represent polar content during optimization. We correct this by weighting all pixel-space loss terms by $\cos\phi(h)$, which approximates the spherical area element $\mathrm{d}\Omega = \cos\phi\,\mathrm{d}\phi\,\mathrm{d}\lambda$. This weighting is applied to the ERP-pixel-space geometric objectives (Eq. 4 and Eq. 6) so their gradient contributions are proportional to the surface area each pixel represents. The latent-space rectified-flow objective $\mathcal{L}_{\text{visual}}$ is left unweighted since ERP latitude has no direct counterpart at the compressed latent grid.

**Circular shift augmentation.** The left and right boundaries of an equirectangular image are adjacent in 3D space ($\lambda = -\pi$ and $\lambda = \pi$ are the same meridian), yet a standard spatial transformer treats them as maximally distant. We apply random horizontal circular shifts to each training panorama, uniformly sampling a column offset from $\{0, \ldots, W-1\}$ and rolling the image along the width axis. This augmentation exposes the model to all possible boundary alignments during training, promoting seamless wrap-around generation at inference time without boundary artifacts.

### 3.3 Video-Conditioned Panoramic Generation

The conditioning pathways accept a perspective video; a single input image is treated as a static clip.

For semantic conditioning, each observed frame is encoded by a frozen image-level semantic encoder (SigLIP2 [Tschannen et al., 2025]). The resulting dense spatial tokens are projected into the DiT feature dimension and injected into each block through cross-attention. Compared with conventional CLIP-style encoders [Radford et al., 2021], SigLIP2 yields substantially richer spatial tokenization, which we find improves scene and object grounding across time. During training, we apply $10\%$ feature dropout to enable classifier-free guidance at inference.

For geometric conditioning, each perspective frame is projected onto the equirectangular canvas using the corresponding camera parameters (FOV, yaw, pitch). Unobserved regions are filled with scaled Gaussian noise, and the resulting sequence is encoded into reference latents $\mathbf{z}_{\text{ref}} = \mathcal{E}(\cdot)$. These reference latents are concatenated with the noisy latents $\mathbf{z}_{\text{noise}}$ and a per-frame visibility mask $\mathbf{m}_{\text{spatial}}$ along the channel dimension to form the DiT input (Figure 2). Because the projection is performed per frame rather than from a single image, the reference latent preserves temporal variation consistent with the observed motion.

During training and at the bootstrap inference stage, observed regions are reinserted at every denoising step through blended diffusion [Avrahami et al., 2022]. This preserves evidence consistency while requiring the model to infer unseen panoramic content.

The reference latent and SigLIP2 pathways are co-active at every training step. The base Cosmos temporal conditioning channel (which replaces a small leading block of latent frames with GT during denoising) is kept at a low-probability schedule (Sec. 4.1), so the panoramic pathways dominate the conditioning signal during training.

### 3.4 Geometry-Aware Training Objectives

Visual realism alone is insufficient for world modeling. Generated panoramas should correspond to a stable 3D scene whose objects evolve smoothly over time. We therefore regularize the denoiser along two complementary axes, a per-frame depth objective and a multi-frame 3D trajectory objective, both operating on the clean-latent estimate $\hat{\mathbf{z}}_0 = \mathbf{z}_\sigma - \sigma\, v_\theta(\mathbf{z}_\sigma, \sigma, \mathbf{c})$, where $\sigma \in [0,1]$ is the rectified-flow noise level, $\mathbf{z}_\sigma = (1-\sigma)\mathbf{z}_0 + \sigma\boldsymbol{\epsilon}$ is the noisy latent, $v_\theta$ is the DiT-parameterized velocity predictor, and $\mathbf{c} = (\mathbf{c}_{\text{sem}}, \mathbf{z}_{\text{ref}}, \mathbf{m}_{\text{spatial}})$ bundles the conditioning context of Sec. 3.3. The main training objective is the standard rectified-flow velocity-matching loss [Liu et al., 2023], $\mathcal{L}_{\text{visual}} = \mathbb{E}_{\sigma,\mathbf{z}_0,\boldsymbol{\epsilon}}\big\|v_\theta(\mathbf{z}_\sigma, \sigma, \mathbf{c}) - (\boldsymbol{\epsilon} - \mathbf{z}_0)\big\|_2^2$. The two auxiliary objectives below augment it with geometric supervision on $\hat{\mathbf{z}}_0$ using pseudo-labels precomputed offline on PanoGeo, requiring no external model inference at training time.

Because auxiliary supervision is unreliable when $\hat{\mathbf{z}}_0$ is still noisy, both auxiliary losses are modulated by a noise-adaptive confidence factor $c(\sigma) = \mathbf{1}[\sigma < \sigma_{\max}] \cdot (1 - \sigma/\sigma_{\max})_+^2$ that fades them in quadratically as denoising progresses ($\sigma_{\max}$ in Appendix D). Pixel-space terms are additionally weighted by the spherical area factor $w_h = \cos\phi(h)$ (Sec. 3.2).

**Depth consistency loss.** A lightweight depth head (compact 3D conv network, $\sim$138K parameters) maps $\hat{\mathbf{z}}_0$ to a normalized depth map $\hat{D} \in [0,1]^{1\times T'\times H'\times W'}$, supervised by panoramic depth pseudo-labels $D^{gt}$ [Lin et al., 2025], which are computed natively on equirectangular input. Because both $\hat{D}$ and $D^{gt}$ are normalized to $[0,1]$, scale-invariance is unnecessary, and we adopt an L1 formulation augmented with an edge-preserving gradient term. Let $\mathcal{M} = \{(t,h,w) : 0.01 < D^{gt}_{t,h,w} < 0.95\}$ mask out saturated sky and near-range artefacts, and let $\mathcal{M}_q \subseteq \mathcal{M}$ further exclude the top $2\%$ per-pixel residuals, which empirically correspond to pseudo-label annotation outliers. Writing $\nabla_a$ for finite differences along axis $a \in \{h, w\}$ with neighbour-valid subset $\mathcal{M}_a$, the depth loss is

$$\mathcal{L}_{\text{depth}} = c(\sigma)\left[\frac{1}{|\mathcal{M}_q|}\sum_{(t,h,w)\in\mathcal{M}_q} w_h\,\big|\hat{D}_{t,h,w} - D^{gt}_{t,h,w}\big| \;+\; \frac{1}{2}\sum_{a\in\{h,w\}}\frac{1}{|\mathcal{M}_a|}\sum_{\mathcal{M}_a}\big|\nabla_a\hat{D} - \nabla_a D^{gt}\big|\right], \tag{4}$$

where the first term encodes absolute depth and the second sharpens discontinuities at object boundaries. Compared with scale-invariant log loss [Eigen et al., 2014], we find direct L1 preferable here because the data is already pre-normalized, and because inverse-depth objectives conflict with ERP area weighting by over-emphasizing near-range pixels.

**Trajectory consistency loss.** Per-frame depth alone does not enforce temporal identity: the same physical point may receive inconsistent depth estimates across frames, and dynamic objects may exhibit physically implausible motion. We therefore lift each tracked 2D point at frame $t$ into a world-frame 3D position via the predicted depth and supervise it directly against the precomputed GT 3D track from PanoGeo. For CoTracker3 [Karaev et al., 2024] track $p$ at frame $t$ with normalized ERP coordinate $\mathbf{u}_{p,t} \in [0,1]^2$ and visibility $v_{p,t} \in \{0,1\}$, we bilinearly sample depth $\hat{D}(\mathbf{u}_{p,t})$ at

the corresponding latent frame, apply the spherical unprojection $\pi_{\mathbb{S}^2} : [0,1]^2 \times [0,1] \to \mathbb{R}^3$ (which maps a normalized ERP coordinate and a depth scalar to the corresponding 3D ray endpoint in the camera frame), and rotate to the world frame via the per-frame camera pose $(\mathbf{R}_t, \mathbf{t}_t)$ (identity for static-camera clips), so that ego-motion in hand-held and walking captures is not supervised as object motion:

$$\mathbf{X}_{p,t}^{\text{pred}} = \mathbf{R}_t^\top \big(\pi_{\mathbb{S}^2}(\mathbf{u}_{p,t},\, \hat{D}(\mathbf{u}_{p,t})) - \mathbf{t}_t\big). \tag{5}$$

Beyond absolute position, we additionally supervise the first- and second-order forward differences of the trajectory by stacking position, velocity, and acceleration into an augmented state $\boldsymbol{\xi}_{p,t} = [\mathbf{X}_{p,t};\, \alpha\, \Delta\mathbf{X}_{p,t};\, \beta\, \Delta^2\mathbf{X}_{p,t}] \in \mathbb{R}^9$, where $\Delta, \Delta^2$ denote forward differences along $t$ and $\alpha, \beta$ control the relative weight of velocity and acceleration (Appendix D). Static scene points have zero GT velocity/acceleration, so rigid-scene self-consistency is enforced implicitly while dynamic objects receive non-trivial motion supervision. With visibility- and latitude-weighted gating

$w_{p,t} = v_{p,t} \cdot \cos\phi_{p,t}$ (where $\phi_{p,t}$ is the latitude of $\mathbf{u}_{p,t}$) and top-$2\%$ trimming over per-element residuals $\|\boldsymbol{\xi}_{p,t}^{\text{pred}} - \boldsymbol{\xi}_{p,t}^{\text{gt}}\|_1$, the trajectory loss is

$$\mathcal{L}_{\text{track}} = c(\sigma)\, \frac{1}{\sum_{p,t} w_{p,t}} \sum_{p,t} w_{p,t}\, \big\|\boldsymbol{\xi}_{p,t}^{\text{pred}} - \boldsymbol{\xi}_{p,t}^{\text{gt}}\big\|_1. \tag{6}$$

Lifting to 3D before comparison tightly couples depth and trajectory supervision: drift in either prediction raises the loss, so the depth head and denoiser must jointly produce a temporally coherent scene geometry. The final training objective combines the rectified-flow visual loss with the two geometry-aware regularizers via scalar weights $\lambda_d, \lambda_\tau \in \mathbb{R}_+$,

$$\mathcal{L} = \mathcal{L}_{\text{visual}} + \lambda_d\, \mathcal{L}_{\text{depth}} + \lambda_\tau\, \mathcal{L}_{\text{track}}, \tag{7}$$

with $\lambda_d$=0.3 and $\lambda_\tau$=0.06 throughout (full schedule in Appendix D). These regularizers are especially important in the panoramic setting, where each frame already covers the full sphere and no direct conditioning remains for the $>90\%$ of pixels outside the input frustum: $\mathcal{L}_{\text{depth}}$ stabilizes global 3D structure across the clip while $\mathcal{L}_{\text{track}}$ preserves plausible object motion across previously unseen panoramic regions.

# 4 Experiments

## 4.1 Experimental Setup

**Dataset.**

We use PanoGeo for both training and evaluation (sources, sizes, and captioning in Appendix A). The 150-clip evaluation suite spans three sources that probe distinct generalization axes: PanoGeo (50, held-out from our self-captured corpus), the test split of ARGUS [Luo et al., 2025] (50), and Habitat-Sim [Puig et al., 2023] (50, synthetic); aggregate scores weight the three sources equally. The self-captured corpus is acquired in publicly accessible spaces; all faces are anonymized via Deface [Xu et al., 2020] prior to training and evaluation.

**Implementation.** We train PanoWorld in two progressive stages: Stage 1 at $256 \times 512$ for $6{,}000$ iterations and Stage 2 at $512 \times 1024$ for $2{,}000$ iterations. Full optimization, loss hyperparameters, and the two-stage inference pipeline are detailed in Appendix D and E.

**Evaluation.** We compare PanoWorld against 360DVD [Wang et al., 2024], Imagine360 [Tan et al., 2025], ARGUS [Luo et al., 2025], Follow-Your-Canvas [Chen et al., 2025a], and OmniRoam [Liu et al., 2026] under two input regimes per clip: *Stage 1* (single perspective image + caption) and *Stage 2* (single ERP first frame + caption, frame-0 anchored). Because Stage 1 induces per-method camera paths that misalign with GT, we report quality along two correspondence-free axes that are well-defined regardless of the predicted trajectory: distributional visual realism (FVD [Unterthiner et al., 2019], FAED, FID [Heusel et al., 2017]), caption alignment (CLIP-T [Radford et al., 2021]), and geometric self-consistency computed on the predicted video alone via PaGeR [Bozic et al., 2026] and BootsTAPIR [Doersch et al., 2024], independent of our DAP/CoTracker3 training labels (3D-Smooth, Depth-$\sigma$, Tr-Life; DAP/CoTracker3 scores in Appendix B.3). The input-parity protocol and metric equations are in Appendix B.




Figure 3: **Qualitative comparison on PanoGeo under Stage 1 input (single perspective image + caption).** One representative ERP frame per method on six clips spanning the three evaluation regimes: PanoGeo (rows 1–2), Argus Testset (rows 3–4), and Habitat-Sim (rows 5–6). The red dashed box on each GT frame marks the perspective crop fed to all methods; the rest of the sphere is hallucinated. Compared methods: OmniRoam [Liu et al., 2026], Imagine360 [Tan et al., 2025], Follow-Your-Canvas (FYC) [Chen et al., 2025a], 360DVD [Wang et al., 2024], and ARGUS [Luo et al., 2025], all run under the input-parity protocol of Sec. 4.1. Stage 2 (ERP first-frame input) is omitted from this figure as a still frame cannot reveal its temporal-anchoring advantage; quantitative Stage 2 results are reported in Table 1.

Table 1: **Quantitative comparison across PanoGeo / Argus testset / Habitat-Sim** (150 **clips, balanced** 50/50/50**).** We report distributional visual realism (FVD, FAED, FID), caption alignment (CLIP-T), and three correspondence-free geometric self-consistency scores computed on the predicted video alone via PaGeR depth and BootsTAPIR 3D tracks (3D-Smooth, Depth-$\sigma$, Tr-Life). Formal definitions and equations for all metrics are in Appendix B.2. *Top – Stage 1*: single perspective image + caption input. *Bottom – Stage 2*: single ERP first frame + caption (frame-0 anchored). Base models: 360DVD / Imagine360 / Follow-Your-Canvas use AnimateDiff, ARGUS uses SVD, OmniRoam uses WAN 2.1, and PanoWorld uses Cosmos 2.5. Best in **bold**, second underlined.

| Method | Visual Realism | | | Caption | Geometric Self-Consistency | | |
|---|---|---|---|---|---|---|---|
| | FVD↓ | FAED↓ | FID↓ | CLIP-T↑ | 3D-Smooth↓ | Depth-$\sigma$ ↓ | Tr-Life↑ |
| *Stage 1: single perspective image + caption* | | | | | | | |
| 360DVD [Wang et al., 2024] | 100.7 | 94.6 | 191.8 | 0.262 | 0.763 | 0.065 | 0.881 |
| Imagine360 [Tan et al., 2025] | 67.7 | 96.6 | 142.9 | 0.253 | 0.364 | 0.099 | 0.594 |
| ARGUS [Luo et al., 2025] | 65.0 | **82.7** | 152.5 | 0.212 | **0.100** | 0.053 | **0.991** |
| Follow-Your-Canvas [Chen et al., 2025a] | 82.2 | 108.8 | 182.3 | **0.279** | 0.463 | 0.140 | 0.600 |
| OmniRoam [Liu et al., 2026] | 98.8 | 131.2 | 178.4 | 0.132 | 1.063 | 0.280 | 0.200 |
| **PanoWorld(Ours)** | **56.1** | 83.4 | **136.4** | 0.264 | 0.101 | **0.033** | 0.977 |
| *Stage 2: single ERP first frame + caption (frame-0 anchored)* | | | | | | | |
| OmniRoam [Liu et al., 2026] | 39.7 | 53.6 | 69.7 | 0.233 | 1.197 | 0.180 | 0.608 |
| **PanoWorld(Ours)** | **28.5** | **41.5** | **55.4** | **0.253** | **0.511** | **0.060** | **0.776** |

### 4.2 Comparison with Baselines

Table 1 reports Stage 1 (top) and Stage 2 (bottom) results aggregated over the 150-clip suite; the per-source breakdown is in Table 3.

**Distributional visual realism.** PanoWorld obtains the lowest FVD (56.1, $-13.7\%$ vs. ARGUS) and the lowest FID (136.4 vs. Imagine360 142.9), confirming that geometry conditioning does *not* sacrifice visual quality. On FAED PanoWorld (83.4) trails ARGUS (82.7) by a thin margin: ARGUS's near-static rollouts match GT temporal statistics easily under a transformer 3D encoder. The two

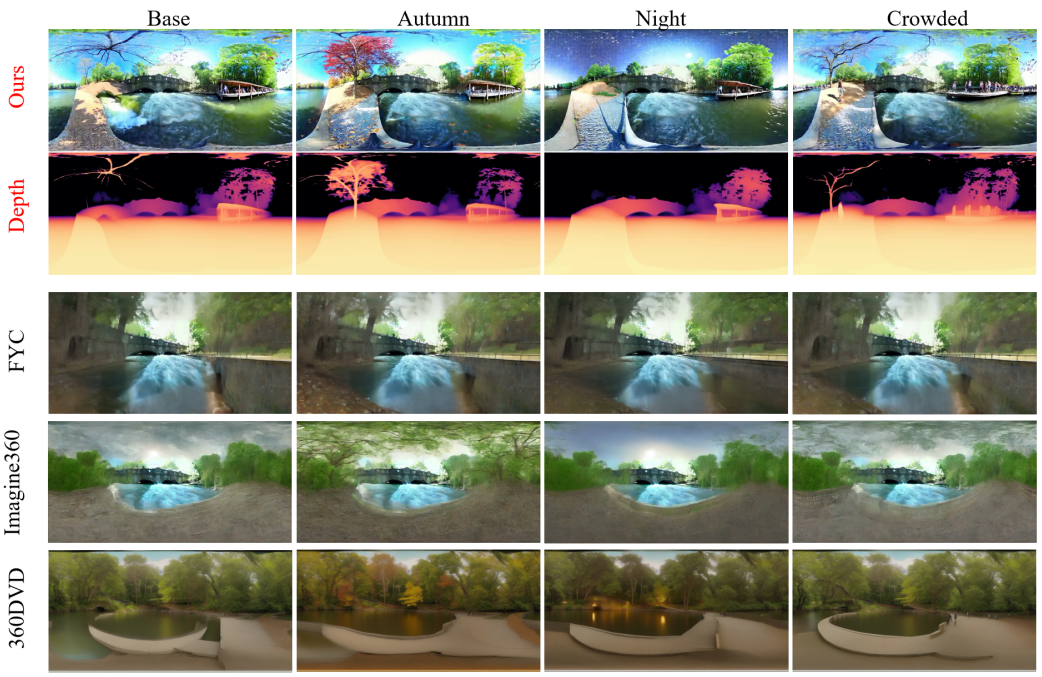



Figure 4: **Prompt controllability from a fixed input image.** Same input image (an Argus testset clip) and seed; only the last clause of the prompt changes per column. The depth row is DAP on our generated frames (bright is near). ARGUS and OmniRoam take no caption and are omitted.

AnimateDiff baselines (360DVD, FYC) and OmniRoam show large gaps on every distributional metric ($+26\sim45$ FVD, $+42\sim55$ FID).

**Caption alignment.** PanoWorld reaches CLIP-T $0.264$, second only to Follow-Your-Canvas ($0.279$). FYC's edge stems from its fixed-FOV outpainting: it never moves the camera away from the prompt-relevant input crop, while full-sphere methods must distribute caption-relevant content across the entire panorama. PanoWorld still substantially outperforms all five-DoF baselines (ARGUS $0.212$, OmniRoam $0.132$).

**Geometric self-consistency.** The geometry losses translate directly into the three self-consistency scores, the central claim of this work. PanoWorld attains the lowest Depth-$\sigma$ ($0.033$, $-37\%$ vs. ARGUS) and is a close second on 3D-Smooth ($0.101$ vs. $0.100$) and Tr-Life. The contrast is starkest against OmniRoam, whose unconstrained backbone yields a 3D-Smooth an order of magnitude worse ($1.06$) and tracks that survive only $20\%$ of frames – evidence that purely visual objectives admit large geometric drift even when frames look plausible. The AnimateDiff baselines (360DVD, Imagine360, FYC) sit $3.6$–$7.6\times$ worse on 3D-Smooth, consistent with per-frame ControlNet-style conditioning. ARGUS needs video input, so it receives a frame-replicated static clip whose quasi-static rollouts favor smoothness; its 3D-Smooth lead stems from its own test split ($0.175$ vs. $0.216$), while PanoWorld is lower on PanoGeo ($0.052$ vs. $0.071$) and Habitat-Sim ($0.035$ vs. $0.054$; Table 3).

**Stage 2 (ERP first frame anchored).** In the Stage2 of Table 1, PanoWorld dominates OmniRoam on all 7 metrics ($-28\%$ FVD, $-23\%$ FAED, $-21\%$ FID, $+9\%$ CLIP-T, $-57\%$ 3D-Smooth, $-67\%$ Depth-$\sigma$, $+28\%$ Tr-Life) with fixed panoramic frame-0. PanoWorld's own Stage 1 $\rightarrow$Stage 2 gap shows the expected behavior: the real ERP anchor sharply lowers FVD ($56.1 \rightarrow 28.5$), while 3D-Smooth and Tr-Life soften because the ERP-conditioned distribution induces larger learned camera motion.

**Qualitative Results.** Figure 3 compares Stage 1 generations on six clips spanning the three sources. Failure modes cluster by input modality: OmniRoam (trajectory-conditioned, no perspective-to-ERP path) generates a sphere disjoint from the GT layout; 360DVD (caption-only) cannot recover the specific details only based on caption; Imagine360 and ARGUS preserve the perspective region but synthesize the rest of the sphere with weak global structure – duplicated objects across the antipode and high-frequency noise on fine details. PanoWorld preserves the input region faithfully and extends it into a globally coherent panorama, matching the GT layout most closely across all three sources.

Table 2: All four variants share the same architecture (ERP RoPE + SigLIP2 + panorama-shaped multi-path conditioning) and the same two-stage training schedule on PanoGeo; only the auxiliary loss weights $\lambda_d, \lambda_\tau$ in Eq. 7 are toggled. We report the same Stage 1 evaluation regime and the same 150-clip split as Table 1. Best in **bold**.

| $\mathcal{L}_{\text{depth}}$ | $\mathcal{L}_{\text{track}}$ | Visual / Caption | | Geometric Self-Consistency | | |
|---|---|---|---|---|---|---|
| | | FVD↓ | CLIP-T↑ | 3D-Smooth↓ | Depth-$\sigma$ ↓ | Tr-Life↑ |
| ✗ | ✗ | 56.72 | 0.263 | 0.177 | 0.055 | 0.899 |
| ✓ | ✗ | 57.47 | **0.267** | 0.149 | 0.041 | 0.933 |
| ✗ | ✓ | 57.05 | 0.263 | 0.115 | 0.039 | 0.940 |
| ✓ | ✓ | **56.12** | 0.264 | **0.101** | **0.033** | **0.977** |

### 4.3 Prompt Controllability

To isolate what the caption controls, we fix the input image and the seed and vary only the text. Each prompt is a fixed *layout prompt*, which describes what lies in the front, right, behind and left of the camera, followed by one *swappable clause* that changes the season, illumination, weather and crowd density (Fig. 4). PanoWorld renders each clause where it belongs: a red-and-gold tree appears for Autumn, a starry, moonlit sky replaces the daytime sky for Night, and onlookers fill the bank and the platform for Crowded, while the bridge, river channel and platform stay fixed. The depth maps show the new tree and people entering as near-field geometry rather than as texture painted over the old scene. Among the baselines, ARGUS and OmniRoam take no caption, FYC barely changes across prompts, Imagine360 picks up only the global mood, and the text-only 360DVD follows the clause but cannot keep the input view.

### 4.4 Ablation Study

Table 2 toggles $\lambda_d$ and $\lambda_\tau$ (Eq. 7) with everything else fixed. The no-loss baseline (row 1) already inherits the Cosmos perspective prior and our panorama-shaped conditioning, so its FVD (56.7) is close to the full model; the ablation thus exposes where the two losses move *geometric* stability rather than appearance. $\mathcal{L}_{\text{depth}}$ alone (row 2) cuts Depth-$\sigma$ by $25\%$ and 3D-Smooth by $16\%$ while leaving FVD essentially unchanged, evidencing that the depth prior acts as geometric "free supervision" rather than a quality regularizer. $\mathcal{L}_{\text{track}}$ alone (row 3) is the single most effective change for the trajectory metrics (3D-Smooth $0.177 \to 0.115$, Tr-Life $0.899 \to 0.940$); it also lowers Depth-$\sigma$ to $0.039$ because lifting 2D tracks through the same depth head back-propagates a depth-style gradient, showing the two losses are mutually reinforcing rather than orthogonal. Stacking both (row 4, full model) is best on $4/5$ metrics and cuts 3D-Smooth by $43\%$ ($0.177 \to 0.101$) and Depth-$\sigma$ by $40\%$ ($0.055 \to 0.033$) versus the no-loss baseline while marginally improving FVD; the ordering is unchanged under DAP/CoTracker3 (Appendix B.3).

## 5 Conclusion

Existing panoramic video generators treat the task as visual synthesis, but downstream embodied applications require a coherent *geometric* state. Our results expose this gap quantitatively: prior baselines remain competitive on distributional realism (FVD/FAED/FID) yet drift on geometric self-consistency (3D-Smooth, Depth-$\sigma$, Tr-Life), indicating visual plausibility without a stable underlying scene. PanoWorld closes this gap by jointly enforcing $\mathcal{L}_{\text{depth}}$ and $\mathcal{L}_{\text{track}}$, which couple per-frame depth to temporal point identity and reduce depth and trajectory drift over the clip. Combined with PanoGeo and a progressive two-stage training schedule, this suggests that geometric supervision is a practical step toward world models with full-sphere spatial understanding for embodied AI.

**Limitations.** Like its Cosmos base model and all compared methods, PanoWorld generates fixed 93-frame clips; arbitrarily long rollouts accumulate drift that we leave to future work. The monocular depth pseudo-labels and the perspective video prior of the base Cosmos model also introduce supervisory bias on equirectangular content.

## References


Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Xiangyu Bai, Yedi Luo, Le Jiang, and Sarah Ostadabbas. Dual-conditioned temporal diffusion modeling for long driving video generation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

Xiangyu Bai, He Liang, Bishoy Galoaa, Utsav Nandi, Shayda Moezzi, Yuhang He, and Sarah Ostadabbas. Moregen: Multi-agent motion-reasoning engine for code-based text-to-video synthesis. In *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2026.

Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, Chenfanfu Jiang, Yizhou Sun, Kai-Wei Chang, and Aditya Grover. Videophy: Evaluating physical commonsense for video generation. *arXiv preprint arXiv:2406.03520*, 2024.

Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, et al. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024.

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.

Vukasin Bozic, Isidora Slavkovic, Dominik Narnhofer, Nando Metzger, Denis Rozumny, Konrad Schindler, and Nikolai Kalischek. Unified panoramic geometry estimation via multi-view foundation models. *arXiv preprint arXiv:2605.26368*, 2026.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators, 2024. URL `https://openai.com/research/video-generation-models-as-world-simulators`. OpenAI Technical Report.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

Jiahao Chang, Chongjie Ye, Yushuang Wu, Yuantao Chen, Yidan Zhang, Zhongjin Luo, Chenghong Li, Yihao Zhi, and Xiaoguang Han. Reconviagen: Towards accurate multi-view 3d object reconstruction via generation. *arXiv preprint arXiv:2510.23306*, 2025.

Hao Chen, Yuqi Hou, Chenyuan Qu, Irene Testini, Xiaohan Hong, and Jianbo Jiao. 360+ x: A panoptic multi-modal scene understanding dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19373–19382, 2024.

Qihua Chen, Yue Ma, Hongfa Wang, Junkun Yuan, Wenzhe Zhao, Qi Tian, Hongmei Wang, Shaobo Min, Qifeng Chen, and Wei Liu. Infinite-canvas: Higher-resolution video outpainting with extensive content generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 2150–2158, 2025a.

Yabo Chen, Yuanzhi Liang, Jiepeng Wang, Tingxi Chen, Junfei Cheng, Zixiao Gu, Yuyang Huang, Zicheng Jiang, Wei Li, Tian Li, et al. Teleworld: Towards dynamic multimodal synthesis with a 4d world model. *arXiv preprint arXiv:2601.00051*, 2025b.

Xiaowei Chi, Peidong Jia, Chun-Kai Fan, Xiaozhu Ju, Weishi Mi, Kevin Zhang, Zhiyuan Qin, Wanxin Tian, Kuangzhi Ge, Hao Li, et al. Wow: Towards a world omniscient world model through embodied interaction. *arXiv preprint arXiv:2509.22642*, 2025.

Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In *Proceedings of the European Conference on Computer Vision*, 2018.

Anna Dawid and Yann LeCun. Introduction to latent variable energy-based models: a path toward autonomous machine intelligence. *Journal of Statistical Mechanics: Theory and Experiment*, 2024 (10):104011, 2024.

Carl Doersch, Pauline Luc, Yi Yang, Dilara Gokay, Skanda Koppula, Ankush Gupta, Joseph Heyward, Ignacio Rocco, Ross Goroshin, Joao Carreira, et al. Bootstap: Bootstrapped training for tracking-any-point. In *Asian Conference on Computer Vision*, pages 483–500. Springer, 2024.

David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*, 2014.

Zixun Fang, Kai Zhu, Zhiheng Liu, Yu Liu, Wei Zhai, Yang Cao, and Zheng-Jun Zha. Panoramic video generation with pretrained diffusion models. In *Advances in Neural Information Processing Systems*, 2025.

Shaohua Gao, Kailun Yang, Hao Shi, Kaiwei Wang, and Jian Bai. Review on panoramic imaging and its applications in scene understanding. *IEEE Transactions on Instrumentation and Measurement*, 71:1–34, 2022.

David Ha and Jürgen Schmidhuber. World models. In *Advances in Neural Information Processing Systems*, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019b.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, 2017.

Wenyi Hong, Ming Ding, Wendi Zheng, Xiao Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. In *International Conference on Learning Representations*, 2023.

Siqiao Huang, Jialong Wu, Qixing Zhou, Shangchen Miao, and Mingsheng Long. Vid2world: Crafting video diffusion models to interactive world models. *arXiv preprint arXiv:2505.14357*, 2025a.

Yuanhui Huang, Weiliang Chen, Wenzhao Zheng, Xin Tao, Pengfei Wan, Jie Zhou, and Jiwen Lu. Terra: Explorable native 3d world model with point latents, 2025b. URL `https://arxiv.org/abs/2510.14977`.

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*, pages 18–35. Springer, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–139:14, 2023.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023.

Bohan Li, Zhuang Ma, Dalong Du, Baorui Peng, Zhujin Liang, Zhenqiang Liu, Xianda Guo, Zheng Zhu, Chao Ma, Yueming Jin, Xin Jin, Hao Zhao, and Wenjun Zeng. Omninwm: Omniscient driving navigation world models. In *European Conference on Computer Vision*, pages 187–208. Springer, 2026a.

Longfei Li, Zhiwen Fan, Wenyan Cong, Xinhang Liu, Yuyang Yin, Matt Foutter, Panwang Pan, Chenyu You, Yue Wang, Zhangyang Wang, et al. Martian world model: Controllable video synthesis with physically accurate 3d reconstructions. *arXiv preprint arXiv:2507.07978*, 2025.

Sikuang Li, Chen Yang, Jiemin Fang, Taoran Yi, Jia Lu, Jiazhong Cen, Lingxi Xie, Wei Shen, and Qi Tian. Worldgrow: Generating infinite 3d world. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 6433–6441, 2026b.

Xin Lin, Meixi Song, Dizhe Zhang, Wenxuan Lu, Haodong Li, Bo Du, Ming-Hsuan Yang, Truong Nguyen, and Lu Qi. Depth any panoramas: A foundation model for panoramic depth estimation. *arXiv preprint arXiv:2512.16913*, 2025.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Yuheng Liu, Xin Lin, Xinke Li, Baihan Yang, Chen Wang, Kalyan Sunkavalli, Yannick Hold-Geoffroy, Hao Tan, Kai Zhang, Xiaohui Xie, et al. Omniroam: World wandering via long-horizon panoramic video generation. *arXiv preprint arXiv:2603.30045*, 2026.

Rundong Luo, Matthew Wallingford, Ali Farhadi, Noah Snavely, and Wei-Chiu Ma. Beyond the frame: Generating 360° panoramic videos from perspective videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Andréa Macario Barros, Maugan Michel, Yoann Moline, Gwenolé Corre, and Frédérick Carrel. A comprehensive survey of visual slam algorithms. *Robotics*, 11(1):24, 2022.

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.

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.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars and robots. *arXiv preprint arXiv:2310.13724*, 2023.

Yong Qin and Haidong Yu. A review of visual slam with dynamic objects. *Industrial Robot: the international journal of robotics research and application*, 50(6):1000–1010, 2023.

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*, 2021.

Sandeep Routray, Hengkai Pan, Unnat Jain, Shikhar Bahl, and Deepak Pathak. Vipra: Video prediction for robot actions. *arXiv preprint arXiv:2511.07732*, 2025.

Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. Gaia-2: A controllable multi-view generative world model for autonomous driving. *arXiv preprint arXiv:2503.20523*, 2025.

Jing Tan, Shuai Yang, Tong Wu, Jingwen He, Yuwei Guo, Ziwei Liu, and Dahua Lin. Imagine360: Immersive 360 video generation from perspective anchor. In *Advances in Neural Information Processing Systems*, 2025.

Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. MVDiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. *Advances in Neural Information Processing Systems*, 36:51202–51233, 2023.

Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5283–5293, 2025.

Team HunyuanWorld. Hunyuanworld 1.0: Generating immersive, explorable, and interactive 3d worlds from words or pixels. *arXiv preprint arXiv:2507.21809*, 2025.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Fvd: A new metric for video generation. In *ICLR 2019 Workshop on Deep Generative Models for Highly Structured Data*, 2019.

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.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025.

Pengfei Wang, Liyi Chen, Zhiyuan Ma, Yanjun Guo, Guowen Zhang, and Lei Zhang. One2scene: Geometric consistent explorable 3d scene generation from a single image. *arXiv preprint arXiv:2602.19766*, 2026.

Qian Wang, Weiqi Li, Chong Mou, Xinhua Cheng, and Jian Zhang. 360dvd: Controllable panorama video generation with 360-degree video diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6913–6923, 2024.

Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022.

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

Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. ivideogpt: Interactive videogpts are scalable world models. In *Advances in Neural Information Processing Systems*, 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*, pages 26057–26068, 2025.

Yuanyuan Xu, Wan Yan, Genke Yang, Jiliang Luo, Tao Li, and Jianan He. Centerface: joint face detection and alignment using face as point. *Scientific Programming*, 2020(1):7845384, 2020.

Shuang Zeng, Xinyuan Chang, Mengwei Xie, Xinran Liu, Yifan Bai, Zheng Pan, Mu Xu, and Xing Wei. Futuresightdrive: Thinking visually with spatio-temporal cot for autonomous driving. *arXiv preprint arXiv:2505.17685*, 2025.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-aware unified world modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8535–8546, 2025a.

Yixuan Zhu, Jiaqi Feng, Wenzhao Zheng, Yuan Gao, Xin Tao, Pengfei Wan, Jie Zhou, and Jiwen Lu. Astra: General interactive world model with autoregressive denoising. *arXiv preprint arXiv:2512.08931*, 2025b.

# A PanoGeo Dataset Construction

PanoGeo is built around a single *unified* panoramic video annotation pipeline applied to three heterogeneous video sources, so that all training and evaluation clips share identical caption / depth / trajectory pseudo-labels regardless of origin.

## A.1 Video Sources

**Self-captured 4K panoramas.** We capture 4K ($3840\times1920$) $360^\circ$ videos with a consumer panoramic camera across diverse environments including hotel lobbies, residential interiors, retail and commercial spaces, museums, urban streets, and natural landscapes. Faces are blurred with Deface [Xu et al., 2020]. From each of the $\sim$200 recordings we extract ten-second clips at varying temporal offsets, yielding $\sim$5,000 clips that capture diverse motion patterns and lighting conditions within each scene.

**Public panorama datasets.** We supplement self-captured data with $\sim$12,000 ERP clips cut from $\sim$230 x360 [Chen et al., 2024] recordings and $\sim$2,000 clips from 360DVD [Wang et al., 2024], providing additional motion categories and outdoor real-world diversity. The model reported in the main paper is trained on a subset of these clips. Clips are re-encoded to a unified $H \times W$ resolution and frame rate before downstream processing.

**Habitat-rendered simulations (Habitat-Sim split).** For evaluation, we additionally render 50 panoramic clips with the Habitat simulator. Each clip uses a randomized camera trajectory inside a sampled scene and is rendered to ERP via a six-face cubemap composite with the same resolution / frame rate as the real-world clips, then passed through the same annotation pipeline. These provide a controlled synthetic split with ground-truth depth that we use only for evaluation.

## A.2 Unified Annotation Pipeline

All clips, regardless of source, pass through the same three-stage annotation pipeline, which is the core of PanoGeo's contribution:

**Captioning.** Each clip is captioned by a vision-language model with a fixed panoramic-aware prompt template, producing scene-content and dynamics descriptions in a uniform style across sources. Training clips are captioned by Gemini 2.5 Flash and evaluation clips by GPT-5 mini; a random subset of captions is audited manually.

**Depth annotation.** Per-frame depth maps are produced by Depth Any Panoramas (DAP) [Lin et al., 2025], a foundation model designed for equirectangular input. DAP operates natively on panoramic frames without perspective decomposition, producing geometrically consistent depth across the full $360^\circ$ field of view. Maps are stored in half-precision (float16) and serve as the pseudo ground truth $D^{gt}$ for the depth consistency loss (Eq. 4).

**Trajectory annotation.** Dense 2D point trajectories and per-point visibility masks are extracted with CoTracker3 [Karaev et al., 2024] on a regular query grid. For clips with appreciable camera motion, we lift tracked 2D points to 3D using DAP depth and the equirectangular spherical projection, then estimate per-frame rigid camera transformations via the Umeyama algorithm with RANSAC outlier rejection, separating camera ego-motion from object motion. The resulting world-frame 3D tracks (and their velocities / accelerations) serve as the pseudo ground truth $\boldsymbol{\xi}^{\text{gt}}_{p,t}$ for the trajectory consistency loss (Eq. 6). The full pipeline is implemented as a resumable batch-processing script with multi-GPU parallelism, enabling annotation of new clips at constant per-clip cost.

## A.3 Evaluation Split

The main paper reports a balanced set of 150 clips, 50 from each of PanoGeo held-out, the Argus testset, and Habitat-Sim, so that aggregate metrics are not dominated by a single source.

**PanoGeo held-out (50 clips).** Held out from our self-captured corpus before training, with the same camera and the same general environment categories but unseen scenes; this is the in-distribution slice of PanoGeo.

Table 3: **Per-source Stage 1 breakdown.** Same metric panel as Table 1; each source contains 50 clips of the balanced subset. Best per (source, metric) in **bold**, second underlined.

| Source | Method | FVD↓ | FAED↓ | FID↓ | CLIP-T↑ | 3D-Sm↓ | D-$\sigma$ ↓ | T-Life↑ |
|---|---|---|---|---|---|---|---|---|
| PanoGeo held-out | 360DVD [Wang et al., 2024] | 136.19 | 132.81 | 266.48 | 0.263 | 0.189 | 0.052 | 0.889 |
| | Imagine360 [Tan et al., 2025] | 103.08 | 129.68 | 214.58 | 0.266 | 0.278 | 0.102 | 0.580 |
| | ARGUS [Luo et al., 2025] | 89.21 | 103.83 | 211.86 | 0.239 | 0.071 | 0.044 | **0.993** |
| | Follow-Your-Canvas [Chen et al., 2025a] | 128.55 | 121.97 | 243.25 | **0.302** | 0.224 | 0.167 | 0.573 |
| | OmniRoam [Liu et al., 2026] | 166.89 | 194.68 | 253.86 | 0.135 | 0.954 | 0.321 | 0.122 |
| | **PanoWorld(Ours)** | **72.98** | **101.81** | **181.46** | 0.274 | **0.052** | **0.025** | 0.991 |
| Argus testset | 360DVD [Wang et al., 2024] | 90.77 | 86.94 | 191.58 | 0.257 | 1.996 | 0.081 | 0.847 |
| | Imagine360 [Tan et al., 2025] | 72.53 | 93.68 | 146.99 | 0.267 | 0.642 | 0.085 | 0.671 |
| | ARGUS [Luo et al., 2025] | **54.31** | **85.12** | **145.84** | 0.241 | **0.175** | **0.048** | **0.996** |
| | Follow-Your-Canvas [Chen et al., 2025a] | 78.46 | 109.83 | 163.78 | **0.287** | 1.102 | 0.125 | 0.703 |
| | OmniRoam [Liu et al., 2026] | 100.56 | 111.18 | 177.17 | 0.169 | 1.874 | 0.275 | 0.310 |
| | **PanoWorld(Ours)** | 56.93 | 86.01 | 150.65 | 0.278 | 0.216 | 0.050 | 0.971 |
| Habitat-Sim | 360DVD [Wang et al., 2024] | 145.02 | 133.26 | 234.17 | **0.265** | 0.105 | 0.063 | 0.907 |
| | Imagine360 [Tan et al., 2025] | 88.08 | 140.73 | **176.20** | 0.226 | 0.171 | 0.109 | 0.532 |
| | ARGUS [Luo et al., 2025] | 106.84 | 129.33 | 210.83 | 0.156 | 0.054 | 0.065 | **0.984** |
| | Follow-Your-Canvas [Chen et al., 2025a] | 100.72 | 159.24 | 237.18 | 0.248 | 0.064 | 0.130 | 0.525 |
| | OmniRoam [Liu et al., 2026] | **79.48** | 165.81 | 188.61 | 0.093 | 0.240 | 0.243 | 0.167 |
| | **PanoWorld(Ours)** | 94.99 | **125.09** | 191.34 | 0.240 | **0.035** | **0.024** | 0.971 |

**Argus testset (50 clips).** Taken from the public Argus YouTube test split [Luo et al., 2025], covering real-world panoramic content from a different capture distribution (different cameras, different photographers).

**Habitat-Sim (50 clips).** Rendered de novo in Habitat with randomized scenes and trajectories, providing a controlled synthetic distribution with deterministic ground-truth depth.

All three sources are evaluated under the identical annotation pipeline and the identical two-stage inference protocol, so per-source metrics differ only because of the underlying video distribution.

**Per-source breakdown.** The aggregate numbers in the main paper (Table 1) hide source-level behaviour that is informative for a panoramic generator's failure modes: the PanoGeo held-out slice measures within-distribution polish, the Argus testset stresses appearance priors against unseen real-world capture conditions, and Habitat-Sim stresses geometric priors against synthetic but precisely controlled scenes. Table 3 reports the same Stage 1 metrics as Table 1 broken down by source (50 clips each).

**Reading the breakdown.** PanoWorld leads the PanoGeo held-out slice on $5/7$ metrics, the source that most directly reflects whether the depth- and trajectory-aware fine-tuning has internalized our capture distribution. On the Argus testset, ARGUS expectedly wins the appearance metrics (FVD/FAED/FID) and all three geometry scores on its own training distribution, with PanoWorld second on five of the six. On Habitat-Sim the picture is mixed because synthetic renderings have unfamiliar lighting and texture statistics that hurt every model's distributional metrics, but PanoWorld retains the lowest 3D-Smooth and Depth-$\sigma$ by a wide margin. ARGUS's Tr-Life lead on every source reflects its quasi-static rollouts from frame-replicated input (Sec. 4.2).

# B Evaluation Protocol and Metrics

We describe here the full input-parity protocol and metric definitions referenced from the main-paper Experiments section.

## B.1 Two-Stage Input Regime

We benchmark all methods under two input regimes per clip in the PanoGeo evaluation split, designed so that every baseline receives the same total information budget.

**Stage 1: single perspective image + caption.**

Each method is given a single perspective image (a $480 \times 640$ crop of the GT panorama's first frame with $90^\circ$ horizontal FOV, facing yaw $0^\circ$ and pitch $0^\circ$) plus the medium-length caption. Methods that natively expect a perspective *video* (Imagine360 [Tan et al., 2025], ARGUS [Luo et al., 2025], Follow-Your-Canvas [Chen et al., 2025a]) receive the same image replicated to their native input length. Methods that natively expect a per-frame trajectory (OmniRoam [Liu et al., 2026]) receive a fixed forward trajectory. PanoWorld runs its full two-stage inference pipeline (Sec. E).

**Stage 2: single ERP first frame + caption (frame-0 anchored).** Each method is given a single equirectangular frame at $t$=0 (the GT panorama's first frame) plus the caption. Only OmniRoam and PanoWorld natively support ERP image input in this setting. PanoWorld runs Round 2 only, with the supplied ERP frame as the anchor.

Both stages are evaluated with the same correspondence-free metric panel (FVD, FAED, FID, CLIP-T, 3D-Smooth, Depth-$\sigma$, Tr-Life; Sec. B.2), so each (method, stage) pair is directly comparable. All metrics are computed on the generated ERP videos; perspective renderings in figures and on the project page are for visualization only.

Each method generates a $T$=93-frame ERP video at $512 \times 1024$ and 16 fps. Predictions and GT are temporally resampled to a common $T_{\text{eval}}$=80-frame grid for metric computation, and any per-method frame-rate / clip-length differences are absorbed by this resampling. Metrics are reported in aggregate over the balanced 150-clip subset (Table 1) and per source over PanoGeo held-out / Argus testset / Habitat-Sim (Table 3).

### B.2 Metric Definitions

We split metrics into two complementary axes. The *correspondence-free* axis is well-defined under any camera path (both stages); the *correspondence-required* axis assumes frame-aligned predictions against GT and is directly meaningful only when frame 0 is anchored (Stage 2).

**Notation.** Let $\hat{V} = \{\hat{V}_t\}_{t=1}^{T_{\text{eval}}}$ with $\hat{V}_t \in [0,1]^{H\times W\times 3}$ denote the predicted ERP video resampled to the common 80-frame grid, and let $V$ denote the GT counterpart. Geometry scores use $T_g$=20 frames uniformly subsampled from $\hat{V}$ at $512\times1024$. Running PaGeR [Bozic et al., 2026] on $\hat{V}$ yields a per-frame metric depth map $\hat{D}_t^{\text{re}} \in \mathbb{R}^{H\times W}$, and running BootsTAPIR [Doersch et al., 2024] on $\hat{V}$ (a $30\times30$ query grid on frame 0, with ERP circular padding) yields $N$ tracked points with 2D positions $\hat{\mathbf{x}}_{i,t} \in \mathbb{R}^2$, visibilities $v_{i,t} \in \{0,1\}$, and 3D positions $\hat{\mathbf{p}}_{i,t} \in \mathbb{R}^3$ obtained by lifting $(\hat{\mathbf{x}}_{i,t}, \hat{D}_t^{\text{re}}(\hat{\mathbf{x}}_{i,t}))$ through the equirectangular spherical unprojection (the same operation used by Eq. 5). Let $c$ denote the medium-length caption.

**Correspondence-free distributional metrics (FVD, FAED, FID).** Each metric is a Fréchet distance between the empirical Gaussian fit to the encoded predicted videos and to the encoded GT videos:

$$d_F\big(\hat{\mathcal{V}}, \mathcal{V}\big) \;=\; \|\mu_{\hat{F}} - \mu_F\|_2^2 + \mathrm{Tr}\big(\Sigma_{\hat{F}} + \Sigma_F - 2(\Sigma_{\hat{F}}\Sigma_F)^{1/2}\big), \tag{8}$$

where $\mu_{\hat{F}}, \Sigma_{\hat{F}}$ and $\mu_F, \Sigma_F$ are the empirical mean / covariance of feature embeddings $\{F(\hat{V}^{(k)})\}_k$ and $\{F(V^{(k)})\}_k$ over the bucket. The choice of encoder $F$ defines the metric: FVD uses a Kinetics-400-pretrained R(2+1)D-18 [Unterthiner et al., 2019], FAED uses a Kinetics-400-pretrained Swin3D-T (a transformer-based 3D encoder, complementary to FVD's convolutional inductive bias), and FID [Heusel et al., 2017] uses InceptionV3 applied per frame and aggregated.

**Caption alignment (CLIP-T).** The mean cosine similarity between the caption embedding and per-frame CLIP image embeddings:

$$\text{CLIP-T}(\hat{V}, c) \;=\; \frac{1}{T_{\text{eval}}} \sum_{t=1}^{T_{\text{eval}}} \cos\big(E_I^{\text{CLIP}}(\hat{V}_t),\, E_T^{\text{CLIP}}(c)\big). \tag{9}$$

**Geometric self-consistency I – 3D track smoothness (3D-Smooth).** The median over visible track positions of the discrete second-order temporal derivative of the lifted 3D track, in meters:

$$\text{3D-Smooth}(\hat{V}) \;=\; \text{median} \underset{\substack{1\le i\le N\\ 1<t<T_g\\ v_{i,t-1}=v_{i,t}=v_{i,t+1}=1}}{} \big\|\hat{\mathbf{p}}_{i,t-1} - 2\hat{\mathbf{p}}_{i,t} + \hat{\mathbf{p}}_{i,t+1}\big\|_2. \tag{10}$$

A static rigid scene with consistent depth has 3D-Smooth $\to 0$; jittery 3D positions inflate this score regardless of whether they look plausible in 2D.

**Geometric self-consistency II – depth temporal standard deviation (Depth-$\sigma$).** For each pixel $(u, v)$ that has a finite, positive predicted depth across all frames, define the temporal mean $\bar{D}(u, v) = \frac{1}{T_g}\sum_t \hat{D}_t^{\text{re}}(u, v)$ and the temporal standard deviation $\sigma_D(u, v) = \sqrt{\frac{1}{T_g}\sum_t (\hat{D}_t^{\text{re}}(u, v) - \bar{D}(u, v))^2}$. We then report:

$$\text{Depth-}\sigma(\hat{V}) \;=\; \text{median}_{(u,v)\in\mathcal{V}_D} \frac{\sigma_D(u, v)}{\bar{D}(u, v)}, \tag{11}$$

where $\mathcal{V}_D$ is the set of pixels with valid depth in every frame. The per-pixel normalisation by mean depth makes the metric invariant to the absolute depth scale and the size of the scene.

**Geometric self-consistency III – track lifetime fraction (Tr-Life).** The mean (over tracks) of the fraction of frames each track stays visible:

$$\text{Tr-Life}(\hat{V}) \;=\; \frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_g}\sum_{t=1}^{T_g} v_{i,t}. \tag{12}$$

A high value indicates that point identities survive across the rollout, evidencing a coherent underlying scene rather than visually plausible but topologically inconsistent renderings. Combined with 3D-Smooth and Depth-$\sigma$, the three scores triangulate "does the generator produce a self-consistent 3D scene".

### B.3 Evaluator Independence

Our training labels come from DAP and CoTracker3, so scoring with the same models could reward imitation of the labels rather than geometric consistency. The main paper therefore scores geometry with PaGeR (indoor/outdoor metric scale head chosen once per clip by its CLIP router on the GT first frame) and BootsTAPIR; frame sampling, padding, lifting and camera compensation are otherwise unchanged. Table 4 reports the same scores under DAP/CoTracker3: the ablation ordering is identical; against baselines, the swap lets ARGUS edge ahead on 3D-Smooth and Tr-Life in Stage 1, while PanoWorld overtakes OmniRoam on Stage-2 3D-Smooth.

Table 4: **Geometry scores under the DAP/CoTracker3 evaluator** (our training-label models), same 150 clips as Tables 1 and 2.

| Method | 3D-Smooth↓ | Depth-$\sigma$ ↓ | Tr-Life↑ |
|---|---|---|---|
| *Stage 1* | | | |
| 360DVD | 0.204 | 0.038 | 0.922 |
| Imagine360 | 0.115 | 0.045 | 0.915 |
| ARGUS | <u>0.032</u> | <u>0.021</u> | <u>0.993</u> |
| Follow-Your-Canvas | 0.113 | 0.048 | 0.935 |
| OmniRoam | 4.079 | 0.299 | 0.231 |
| PanoWorld (Ours) | **0.025** | **0.013** | **0.994** |
| *Stage 2* | | | |
| OmniRoam | **0.774** | 0.147 | 0.628 |
| PanoWorld (Ours) | 0.981 | **0.032** | **0.896** |
| *Ablation* ($\mathcal{L}_{\text{depth}}$ / $\mathcal{L}_{\text{track}}$) | | | |
| ✗ / ✗ | 0.051 | 0.028 | 0.958 |
| ✓ / ✗ | 0.041 | 0.016 | 0.981 |
| ✗ / ✓ | 0.031 | 0.014 | 0.992 |
| ✓ / ✓ | **0.025** | **0.013** | **0.994** |

## C External Components

The main paper refers to the off-the-shelf components below by their model names. This section lists their exact public versions for reproducibility. The base video world model is Cosmos Predict 2.5 [Agarwal et al., 2025] (2B post-trained checkpoint), which bundles the WAN 2.1 VAE [Wan

et al., 2025] as the spatiotemporal tokenizer and Qwen2.5-VL-7B as the text encoder. For semantic conditioning we use SigLIP2 So400m/14 at $384^2$ input resolution (1152-d hidden, 729 tokens per frame), which is referred to in the main paper simply as *SigLIP2* [Tschannen et al., 2025]. Pseudo-label annotations on PanoGeo are produced once offline using DAP [Lin et al., 2025] for per-frame equirectangular depth and CoTracker3 [Karaev et al., 2024] for dense 2D point tracks. Both are kept frozen throughout training and inference. Geometry metrics are scored with independent models: PaGeR [Bozic et al., 2026] (unified `prs-eth/PaGeR` checkpoint) and BootsTAPIR [Doersch et al., 2024] (PyTorch `bootstapir_checkpoint_v2`). The metric encoders used for FVD / FAED / FID / CLIP-T are described in Sec. B.

# D Two-Stage Training Strategy

We adopt progressive fine-tuning from the base Cosmos 2B pre-trained weights, with the SigLIP2 projection layer ($\mathbf{W}_{\text{proj}} \in \mathbb{R}^{1152\times 2048}$) and I2V cross-attention layers randomly initialized:

**Stage 1: Low-resolution structure learning.** Training at $256 \times 512$ resolution with 93 frames ($T' = 24$ latent frames) for 6,000 iterations. The reduced resolution enables faster iteration on a single H100 80GB GPU, allowing the model to learn panoramic spatial structure, horizontal wrap-around behavior, SigLIP2 feature mapping, and the blended diffusion conditioning mechanism.

The model is trained from the base Cosmos 2B pre-trained weights, and the SigLIP2 projection layer is learned from scratch.

**Stage 2: High-resolution quality refinement.** Training at $512 \times 1024$ resolution with 93 frames, initialized from the Stage 1 checkpoint, for 2,000 iterations. This stage refines texture detail and generation quality while building on the spatial foundations from Stage 1.

Both stages use random FOV sampling ($30^\circ \sim 120^\circ$), random yaw rotation, and horizontal continuity augmentation. The base Cosmos temporal conditioning channel is kept at the standard schedule ($p(\text{num_cond_frames}\in\{0,1,2\})=\{0.5, 0.25, 0.25\}$), with the SigLIP2 cross-attention and reference-latent pathways always co-active and supplying the dominant panoramic conditioning at every training step.

**Optimizer and loss hyperparameters.** Both stages use AdamW ($\beta_1$=0.9, $\beta_2$=0.999, base learning rate $2^{-15}\approx 3.05\times 10^{-5}$ under the base Cosmos warm-up and linear-decay schedule, weight decay $10^{-3}$, batch size 1) on $T$=93-frame clips at 16 fps. The geometry loss weights are $\lambda_d$=0.3 and $\lambda_\tau$=0.06 in $\mathcal{L} = \mathcal{L}_{\text{visual}} + \lambda_d\mathcal{L}_{\text{depth}} + \lambda_\tau\mathcal{L}_{\text{track}}$. The noise-adaptive confidence factor $c(\sigma)$ uses $\sigma_{\max}$=0.85 on the rectified-flow noise scale $\sigma \in [0, 1]$, so the auxiliary losses are switched off only at the noisiest levels. The augmented track-state coefficients are $\alpha$=0.5, $\beta$=0.25. A linear warm-up of 1,000 iterations ramps both auxiliary loss weights from 0 to their full values to let the DiT backbone stabilize before the depth head dominates the gradient flow.

**Frozen modules.** The WAN 2.1 VAE and the SigLIP2 encoder remain frozen throughout. Only the DiT parameters and the latent-space depth head ($\sim$138K params) are updated.

# E Two-Stage Inference Pipeline

Our main-paper training procedure produces a fine-tuned diffusion prior over fixed-length 93-frame panoramic clips, conditioned on a perspective input video, SigLIP2 features, and a panoramic reference latent. Producing the evaluation video from a single perspective input proceeds in two stages: a *Round-1* pass that bootstraps a 93-frame ERP video using the same conditioning interface as training, and a *Round-2* pass that consumes the first frame of Round 1 as an ERP image anchor and produces the final evaluation video via image-conditioned video2world. All quantitative results in the main paper are computed on Round-2 outputs ($T$=93 frames at 16 fps, i.e. $\sim$5.8 seconds at $512 \times 1024$). Round 1 acts as an internal pre-generation step.

**Why a separate Round 2.** A direct image-to-video call on the Round-1 first frame visibly degrades quality compared with Round-1 outputs. The cause is that the fine-tuned DiT was trained to consume two extra inputs at every forward pass, the panoramic reference latent $\mathbf{z}_{\text{ref}}$ and the SigLIP2 cross-attention features $\mathbf{F}$. Feeding zeros (the default behaviour of standard image-to-video calls) is severely

out-of-distribution for the fine-tuned weights and produces washed-out colour, blurry texture, and reduced motion magnitude.

**Conditioning recipe for Round 2.** We preserve the fine-tuning interface in Round 2 with three minor modifications relative to Round 1:

- The reference latent is the VAE-encoding of the entire Round-1 ERP video (93 frames), restoring the panorama-shaped context the DiT expects.
- The Round-1 SigLIP2 features are reused verbatim, since they were derived from a perspective input that is in-distribution with training.
- The spatial mask is set to one only on the first latent frame and zero elsewhere, indicating that Round 2 must trust the anchor frame and synthesize the remaining frames freely. We do *not* apply masked blending at every denoising step in Round 2 (in contrast to Round 1): blending would re-lock the frame-0 region to the ERP anchor at every step and suppress motion across the full sphere.

**Beyond the Round-2 horizon.**

The same procedure could be repeated to extend the horizon by re-encoding the previous round's ERP output as the new reference latent and taking its first frame as the new anchor. We do not evaluate this longer-horizon setting in this work, and report all quantitative metrics on Round-2 outputs only. Additional generated videos, including scenes with moving people and vehicles, are shown on the project page.

# F Qualitative 3D Lifting of Generated Panoramas

Because PanoWorld is depth-supervised at training time, its panoramic output is already approximately geometry-consistent with a panoramic depth estimator, which lets us chain it with closed-form spherical lifting and an off-the-shelf rasterizer to lift a generated panorama into a coloured point cloud and render it from nearby viewpoints. We present this chain qualitatively. The pipeline has four stages.

**Step 1: Perspective image → panoramic video.** Given a single perspective image (with a caption), the two-stage inference of Sec. E produces a 93-frame ERP video at $512 \times 1024$ that captures the full $360^\circ$ surrounding scene with temporal motion.

**Step 2: ERP frame → panoramic depth.** A panoramic depth map $\hat{D}_t \in [0,1]^{H \times W}$ is recovered for each frame. Two depth sources are equally compatible: the latent depth head ($\sim$138K params, Sec. 3.4) used during training, which produces $\hat{D}_t$ in the same forward pass as the diffusion denoiser, or an off-the-shelf panoramic depth predictor such as DAP. Because the diffusion model is trained against DAP pseudo-GT, both choices stay geometrically consistent with the ERP frame at the same time index.

**Step 3: Spherical unprojection → dense 3D point cloud.** Each ERP pixel $(u, v) \in [0,1]^2$ together with its depth $\hat{D}_t(u, v)$ is lifted to a 3D Cartesian point via the same spherical unprojection $\pi_{\mathbb{S}^2}$ used by the trajectory loss (Eq. 5). Stacking all valid pixels per frame yields a dense per-frame point cloud anchored in the world frame, and concatenating across the 93 frames yields a temporally evolving 4D point cloud. A light planar regularizer collapses near-coplanar regions (walls, floor, ceiling) to suppress speckle from depth noise.

**Step 4: Rasterization → novel views.** A virtual camera with pose $(\mathbf{R}, \mathbf{t})$ near the capture position and pinhole intrinsics $\mathbf{K}$ is rendered from the lifted scene. Direct point-based reprojection with $z$-buffering produces the novel views, and substituting per-point Gaussians and a 3D Gaussian Splatting rasterizer yields visually smoother novel-view videos at no extra training cost. Because the point cloud is lifted from a single viewpoint, regions occluded from the capture position are missing and appear as holes once the camera moves far from it.

**Open-source reference implementation.** We release a stand-alone reference implementation of this pipeline. It reads a PanoWorld ERP video, runs DAP on its first frame for stable per-pixel depth, applies the planar regularizer to produce a colored point cloud, renders novel-view trajectories (orbit, walk-through, fly-through) with a 3D Gaussian Splatting rasterizer, and finally upsamples the

rendering with a 2x Real-ESRGAN super-resolution pass to compensate for residual blur near image boundaries. None of these post-processing modules are trained or fine-tuned for this work.

**Scope.** From one perspective photograph and a caption, the chain yields (i) a $360^\circ$ panoramic video, (ii) per-frame panoramic depth, (iii) a 4D point cloud, and (iv) novel views near the capture position, at small additional cost on top of panoramic video generation. We do not evaluate the reconstructed geometry quantitatively, and the result inherits the errors of the monocular depth estimate; dedicated explorable-scene generators such as HunyuanWorld 1.0 [Team HunyuanWorld, 2025] and One2Scene [Wang et al., 2026] target this setting directly.

# G Broader Impacts

**Positive impacts.** Geometry-consistent panoramic video generation lowers the production cost of immersive content for VR/AR education and training, virtual touring of inaccessible spaces (heritage sites, hazardous environments, remote locations), telepresence, and embodied-AI simulators that need full-sphere visual states from limited capture. The 3D lifting chain (Appendix F) may also lower the cost of simple 3D content creation from a single perspective photograph plus a caption.

**Potential negative impacts and mitigations.** Like any high-fidelity generative video model, PanoWorld could in principle be misused for immersive disinformation, fake virtual tours, or deepfake-style content embedded in $360^\circ$ media. We mitigate the most direct privacy risk by anonymizing all faces in the released self-captured corpus with Deface [Xu et al., 2020] (Sec. 4.1, Appendix A), and the released model checkpoints will ship with terms of use that prohibit deepfake / impersonation use, in line with the upstream Cosmos Predict 2.5 safety policy that we inherit and do not bypass. Detection of synthetic panoramic content is an open problem; we encourage downstream users to combine our model with watermarking and provenance-tracking standards as they mature.

## NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and the bullet list of contributions in Sec. 1 state three claims: (i) framing panoramic video generation as a geometry- and dynamics-consistent state modeling problem with two auxiliary losses (depth and 3D track consistency); (ii) spherical-geometry-aware adaptation (ERP RoPE, panorama-shaped multi-path conditioning, SigLIP2 features); and (iii) the PanoGeo dataset with a 150-clip evaluation split (50 clips per source). All three claims are quantitatively supported by Table 1 and Table 2, and qualitatively by Fig. 3, which show that PanoWorld improves geometric self-consistency without sacrificing visual realism on PanoGeo's evaluation split.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Sec. 5 contains an explicit **Limitations** paragraph: PanoWorld generates fixed 93-frame clips, longer rollouts would accumulate drift, and the monocular DAP depth pseudo-labels together with the perspective-video Cosmos prior introduce supervisory bias on equirectangular content. Compute requirements and per-stage compute budgets are stated in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

   Justification: The paper does not present theoretical results. All formulas (Eqs. 4–6) are constructive definitions of the training losses, not theorems.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Sec. 3 fully specifies the architecture (panorama-shaped multi-path conditioning, ERP RoPE, depth head, two losses with explicit weights). Sec. 4.1 specifies dataset splits and the input-parity protocol. Appendix D lists every optimizer / loss hyperparameter (AdamW, base learning rate $2^{-15}\approx 3.05\times 10^{-5}$, $\lambda_d$=0.3, $\lambda_\tau$=0.06, $\sigma_{\max}$=0.85, $\alpha$=0.5, $\beta$=0.25) and the two-stage schedule, Appendix E fully specifies the two-round inference pipeline, Appendix A specifies the PanoGeo construction (sources, captioning, DAP/CoTracker3 annotation), and Appendix B specifies metric definitions and the input-parity protocol. The base perspective prior (Cosmos Predict 2.5, 2B post-trained), tokenizer (WAN 2.1 VAE), text encoder (Qwen2.5-VL-7B), depth model (DAP), tracker (CoTracker3), and semantic encoder (SigLIP2 So400m/14) are listed by exact public version in Appendix C.

   Guidelines:

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

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: Code is available at `https://github.com/ostadabbas/PanoWorld`, including the training and evaluation pipeline (built on Cosmos Predict 2.5), the PanoGeo curation and annotation scripts, the trained PanoWorld checkpoints, and the end-to-end "perspective image $\rightarrow$ explorable 3D scene" pipeline (Appendix F). The full setup is described in Appendix D–B. Public dataset components (x360, 360DVD, Argus, Habitat) and pretrained components (Cosmos Predict 2.5, WAN 2.1 VAE, Qwen2.5-VL, SigLIP2, DAP, CoTracker3) are redistributed under their original licenses, listed in Appendix C.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Sec. 4.1 states the data splits (150-clip evaluation, balanced $50/50/50$ across PanoGeo held-out / Argus testset / Habitat-Sim), the two-stage training schedule, and

the metric panel. Appendix D provides the full optimizer configuration, loss weights, augmentations, frozen modules, and stage-wise iteration counts. Appendix B provides the full input-parity protocol that fixes input length, FoV, and trajectory budget across all baselines.

Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: Following common practice in panoramic and long-form video generation benchmarks, we report a single inference per (method, clip) pair and aggregate over the 150-clip split (50 per source). Re-sampling each method on every clip with multiple random seeds is prohibitive at the resolution and clip length we operate ($512\times1024$, 93 frames per inference, 5 baselines + ablations across two input regimes). The distributional metrics (FVD, FAED, FID) are themselves population-level statistics computed over 150 predicted vs. 150 GT clips, which absorbs per-clip generation noise; the per-clip metrics (CLIP-T, 3D-Smooth, Depth-$\sigma$, Tr-Life) are mean / median values over the same 150 clips.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Sec. 4.1 and Appendix D state that training runs on a single H100 80GB GPU with batch size 1, 93-frame clips, in two stages (6,000 iterations at $256\times512$ followed by 2,000 iterations at $512\times1024$). Appendix E specifies the two-round pipeline. Each Round-2 output is a 93-frame clip at 16 fps ($\sim$5.8 s of video) and $512\times1024$. Pseudo-label generation (DAP depth and CoTracker3 trajectories) is described in Appendix A as a one-off offline

step that scales linearly per clip. Total compute budget, including failed exploratory runs that did not enter the paper, is roughly an order of magnitude larger than the reported runs, dominated by ablation training and offline annotation.

Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics. The self-captured panoramic video corpus was acquired by the authors in publicly accessible spaces; every face is anonymized via Deface [Xu et al., 2020] prior to training and evaluation (see Sec. 4.1 and Appendix A). Public datasets (x360, 360DVD, Argus, Habitat) are used under their original licenses, and external pretrained models are used under their public terms.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Appendix G discusses both sides. Positive: PanoWorld can lower the cost of producing geometry-consistent immersive content for VR/AR education, telepresence, virtual touring, and embodied-AI simulators. Negative: like any high-quality video generative model, panoramic generation could in principle be misused for immersive disinformation or deepfake content; we discuss mitigations including face anonymization in the released dataset, terms-of-use on model release, and the inheritance of upstream Cosmos Predict 2.5 safety filters.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [Yes]

    Justification: All faces in the self-captured portion of PanoGeo are anonymized with Deface [Xu et al., 2020] before any training, evaluation, or release (Sec. 4.1, Appendix A). The released checkpoints and dataset will ship with terms of use that prohibit deepfake / impersonation use; the underlying Cosmos Predict 2.5 perspective prior already includes upstream safety filters that we do not bypass during fine-tuning. Public ERP source corpora (x360, 360DVD, Argus) are redistributed only via the original publishers' channels.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All external assets are credited in the main paper and listed by exact public version in Appendix C: Cosmos Predict 2.5 [Agarwal et al., 2025] (NVIDIA OneWay Noncommercial License), WAN 2.1 VAE [Wan et al., 2025] (Apache 2.0), Qwen2.5-VL-7B (Apache 2.0), SigLIP2 So400m/14 [Tschannen et al., 2025] (Apache 2.0), DAP [Lin et al., 2025] (per its public release), CoTracker3 [Karaev et al., 2024] (CC-BY-NC). Public ERP video corpora x360 [Chen et al., 2024], 360DVD [Wang et al., 2024], and the Argus YouTube test split [Luo et al., 2025] are used under their original distribution terms, and Habitat-rendered scenes are produced from publicly available 3D assets under their original licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: We introduce two new assets: (i) the PanoGeo dataset, fully documented in Appendix A (sources, capture protocol, captioning, depth and trajectory annotation, the 150-clip evaluation split with 50 clips per source, and the face anonymization preprocessing); (ii) the PanoWorld model checkpoints, documented by the architecture description in Sec. 3, the two-stage training recipe in Appendix D, and the two-round inference pipeline in Appendix E. Both assets are released alongside the paper at `https://github.com/ostadabbas/PanoWorld`, with a structured datasheet / model card and an explicit license.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: We do not run crowdsourcing experiments or human-subject studies. The self-captured panoramic videos are recorded by the authors themselves in publicly accessible spaces; no third-party participants are involved.

    Guidelines:

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

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

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

    Answer: [Yes]

Justification: The study received an IRB exemption from Northeastern University based on its scope (no formal human-subject experimentation; only incidental capture of bystanders in publicly accessible spaces). As an additional safeguard, every face in the self-captured corpus is anonymized with Deface [Xu et al., 2020] prior to use in training, evaluation, and any release; this is documented in Sec. 4.1 and Appendix A.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [Yes]

Justification: LLMs / VLMs appear at two non-core points in our pipeline, both fully disclosed. (i) Captions for PanoGeo are produced by Gemini and cross-checked by an OpenAI model under a fixed panoramic-aware prompt template (Sec. 4.1, Appendix A); these captions are pseudo-labels consumed as training/evaluation inputs and are not part of the core method. (ii) The text-conditioning pathway of the diffusion backbone reuses Qwen2.5-VL-7B as the text encoder, which is a frozen pre-existing component of Cosmos Predict 2.5 (Appendix C); we do not modify or fine-tune it. The novel components of PanoWorld (panorama-shaped multi-path conditioning, ERP RoPE, depth and 3D track losses) do not involve LLMs.

Guidelines:

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