PANO123: SINGLE-PANORAMA TO EXPLORABLE 3D SCENE GENERATION

Anonymous authors
Paper under double-blind review

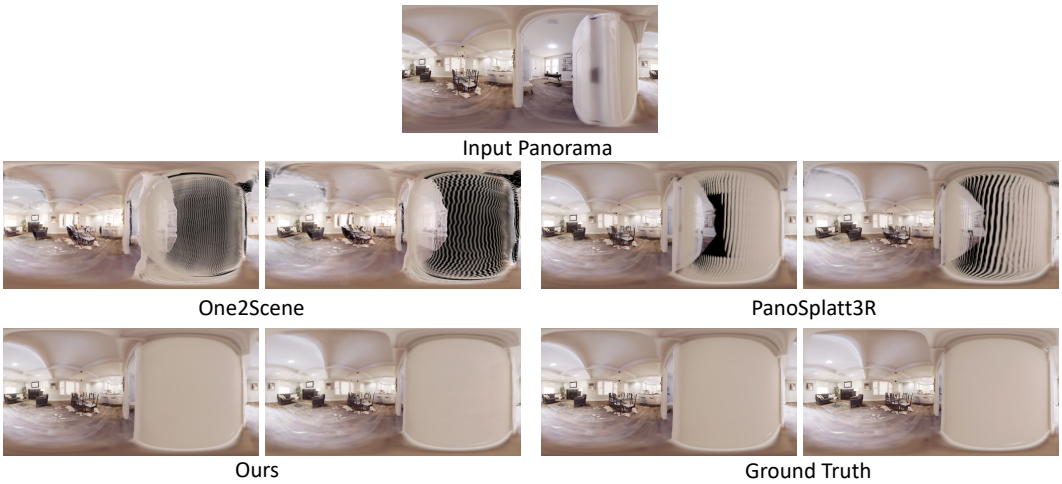

Figure 1: Single-panorama explorable 3D scene generation. Each panel shows two rendered novel views of the same scene, under a moderate (left) and a large (right) camera translation. The baselines (second row) degrade as the translation grows: One2Scene exhibits seam artifacts and holes in disoccluded regions, and PanoSplatt3R leaves stretched surfaces. Pano123 (bottom left) remains coherent under both translations and closely matches the ground truth (bottom right), while running in a single feed-forward pass without per-scene optimization.

ABSTRACT

Generating an explorable 3D scene from a single equirectangular panorama remains underconstrained because camera translation reveals surfaces occluded in the input. These disoccluded regions contain no observed geometry or appearance, making their completion the key bottleneck. We present Pano123, a feed-forward framework that predicts a complete 3D Gaussian Splatting scene from a single panorama in one forward pass, without per-scene optimization. Our key insight is that disocclusion should be treated as a sparse, geometrically structured completion problem rather than as dense scene hallucination. Disoccluded regions typically occupy only a small portion of a target view and arise predictably near occlusion boundaries. They can therefore be localized first and completed only where necessary. Accordingly, Pano123 comprises three stages: a sphere-aware Gaussian initializer that accounts for the non-uniform sampling of equirectangular projection; an annotation-free query generator trained with cross-view reprojection pseudo-labels to localize disocclusions and sample candidate points behind visible surfaces; and a context-grounded decoder that synthesizes new Gaussians from neighboring attribute priors and bounded residuals, while preserving visible Gaussians. Experiments on indoor benchmarks show that we outperform prior feed-forward methods, with the largest gains under large camera translations.

1 INTRODUCTION

Reconstructing an explorable 3D scene from a single image has broad applications in virtual reality, robotic navigation, and spatial understanding. Among common image modalities, the equirectangular panorama (ERP) is particularly well suited to this setting: it is the minimal single capture that observes the full surrounding environment, and commodity 360° cameras make such inputs

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

widely accessible. Yet full angular coverage does not imply complete scene coverage. Foreground objects still occlude surfaces behind them, and these hidden regions become visible when the virtual camera translates. This phenomenon, known as *disocclusion*, creates the central difficulty in single-panorama reconstruction because the newly exposed geometry and appearance are entirely absent from the input.

Recent feed-forward 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) methods achieve impressive novel-view synthesis from perspective images (Charatan et al., 2023; Chen et al., 2024a; Mescheder et al., 2025; Jiang et al., 2025), but they do not directly transfer to panoramic inputs. ERP exhibits strongly non-uniform angular sampling: pixels near the poles subtend much smaller solid angles than those near the equator. This breaks the planar-perspective assumptions used by standard Gaussian parameterizations and often causes distorted geometry and polar artifacts. Panorama-specific methods have begun to address this issue, but remain limited. PanoSplatt3R (Ren et al., 2025) and Splatter-360 (Chen et al., 2025) require multiple panoramic inputs, and MVSplat360 (Chen et al., 2024b) relies on sparse multi-view observations; they are therefore unsuitable when only one capture is available. One2Scene (Wang et al., 2026a) decomposes a panorama into perspective crops and processes them independently, which weakens global geometric consistency across crop boundaries. More fundamentally, existing methods do not explicitly reason about where missing content will be revealed. They rely on implicit completion throughout the scene, often leaving holes or blurred structures near occlusion boundaries, as illustrated in Fig. 1.

Our key idea is to cast single-panorama reconstruction as a *localize-then-complete* problem: first identify where disocclusion will occur, then synthesize new content only at those locations. This formulation follows from three observations.

First, disocclusion is sparse and geometrically structured. Newly exposed regions usually occupy only a small fraction of a target view and tend to appear near depth discontinuities. Treating the entire scene as missing is therefore unnecessarily expensive and increases the risk of cross-view inconsistency. Explicit localization concentrates model capacity on the regions that actually require hallucination.

Second, disocclusion completion should be additive rather than corrective. The visible Gaussians already explain the observed panorama, so modifying them can degrade input-view fidelity. A better strategy is to keep the visible scene fixed and add new Gaussians only behind occluding surfaces. This separates faithful reconstruction from uncertain completion, and converts unconstrained scene generation into bounded residual prediction around locally observed geometry and appearance.

Third, disocclusion can be supervised without manual annotation. Multi-view panoramic training data provides the necessary signal through cross-view reprojection. A reconstruction built from one view can be rendered into a held-out view, where missing or erroneous pixels reveal the regions that require completion. This yields scalable pseudo-labels for disocclusion localization.

Guided by these observations, we propose Pano123, a three-stage feed-forward framework for reconstructing an explorable 3DGS scene from a single equirectangular panorama. In Stage I, a *Sphere-Aware Gaussian Initializer* back-projects the panorama into visible Gaussians in angular coordinates, corrects their scales according to latitude, and refines them with a zero-initialized residual head trained through differentiable rendering. It also extracts multi-scale features shared by later stages. In Stage II, a lightweight *Disocclusion Query Generator* predicts a per-pixel disocclusion heatmap, selects sparse locations using spherically uniform pooled top-K sampling, and ray-marches behind the visible surface to generate 3D query candidates. In Stage III, a *Context-Grounded Attribute Decoder* attends to nearby visible Gaussians, initializes each candidate from neighbor-borrowed attribute priors, and predicts bounded residuals to synthesize new Gaussians. These Gaussians are added to the unchanged visible set, yielding a completed scene in a single forward pass without per-scene optimization.

On standard benchmarks including HM3D and Replica, Pano123 outperforms prior feed-forward methods on every metric under large camera displacements, where disocclusion is most severe, and remains highly competitive under small displacements. Our contributions are threefold:

- A *sphere-aware Gaussian parameterization* that respects ERP geometry through latitude-dependent scale correction and predicts position residuals in angular coordinates, mitigating polar artifacts and stabilizing optimization.

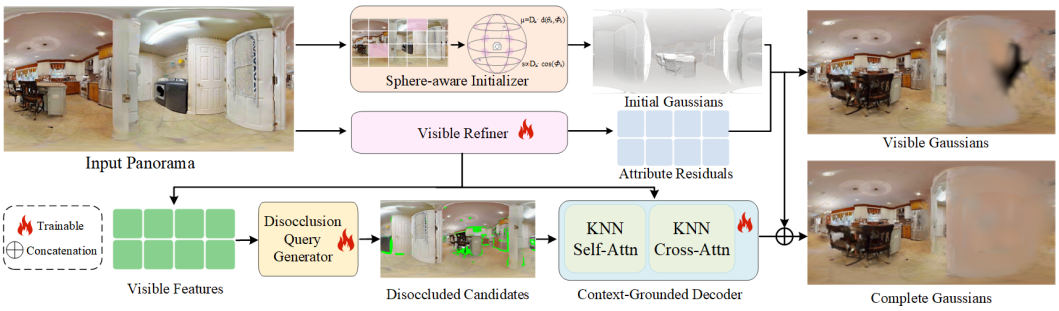

Figure 2: Overview of the Pano123 framework. Given a single equirectangular panorama, our three-stage pipeline produces a complete 3D Gaussian scene: (1) sphere-aware initialization generates visible Gaussians, (2) the disocclusion query generator localizes sparse regions requiring completion and spawns ray-marched query candidates, and (3) the context-grounded attribute decoder synthesizes new Gaussians for these candidates, which are added to the untouched visible set.

- An *annotation-free disocclusion query generator* that localizes completion targets via learned heatmaps trained with cross-view reprojection pseudo-labels, operating deterministically at only the sparse locations requiring new content.
- A *context-grounded additive decoder* that synthesizes occluded-region Gaussians from neighbor-borrowed attribute priors with zero-initialized bounded residuals, achieving a stable cold start while leaving visible Gaussians untouched to preserve input-view fidelity.

2 Related Work

Feed-Forward 3D Gaussian Splatting. Feed-forward methods predict 3D representations directly from images, avoiding per-scene optimization, following earlier radiance-field approaches such as pixelNeRF (Yu et al., 2021). pixelSplat (Charatan et al., 2023) uses epipolar transformers for image pairs, while MVSplat (Chen et al., 2024a) builds cost volumes for efficient multi-view prediction. Sharp (Mescheder et al., 2025) predicts dense Gaussians from a single perspective image, LGM (Tang et al., 2024) first generates multi-view images with diffusion, and AnySplat (Jiang et al., 2025) supports unconstrained input views. These methods, however, assume perspective projection and do not model the non-uniform sampling of equirectangular panoramas or explicitly address panoramic disocclusion. Pano123 extends feed-forward Gaussian prediction to single-panorama input through explicit *localize-then-complete* reasoning.

Panoramic 3D Reconstruction. Panoramas provide full angular coverage and have been widely used for 3D reconstruction, alongside work on generating panoramas from text or narrow-view images (Li & Bansal, 2023; Wu et al., 2023; Feng et al., 2023; Lee et al., 2023; Zheng et al., 2025). PanoSplatt3R (Ren et al., 2025), Splatter-360 (Chen et al., 2025), PanSplat (Zhang et al., 2025), and MVSplat360 (Chen et al., 2024b) produce high-quality panoramic reconstructions but require multiple input views. One2Scene (Wang et al., 2026a) processes perspective crops independently, weakening global consistency, while CylinderSplat (Wang et al., 2026b) predicts dense Gaussian attributes without explicitly separating observed and disoccluded regions. Scene4U (Huang et al., 2025b) restores occluded content with diffusion-based layered inpainting, but requires per-scene hierarchical optimization. In contrast, Pano123 directly processes a single panorama, requires no optimization, and explicitly localizes and completes disoccluded regions.

Disocclusion and Scene Completion. 2D inpainting methods such as RePaint (Lugmayr et al., 2023) and LaMa (Suvorov et al., 2022) fill missing regions without 3D consistency, while 3D-aware methods including SPIn-NeRF (Mirzaei et al., 2023) and 3DGIC (Huang et al., 2025a) add cross-view coherence but require per-scene optimization. WonderJourney (Yu et al., 2024) and WonderWorld (Yu et al., 2025) expand scenes from a single perspective image, while LayerPano3D (Yang et al., 2025) and DreamScene360 (Zhou et al., 2024) hallucinate panoramic disocclusions with diffusion at high cost and limited multi-view consistency. Volumetric completion (ScanComplete; Dai et al., 2018) does not transfer to Gaussians. Pano123 instead localizes sparse disocclusions via cross-view pseudo-labels and completes them deterministically in a single forward pass.

3 PANO123

3.1 PROBLEM FORMULATION AND OVERVIEW

Given a single equirectangular panorama $I \in \mathbb{R}^{3\times H\times W}$ of height H and width W, our goal is to predict a complete 3D Gaussian Splatting (Kerbl et al., 2023) representation $\mathcal{G} = \{\boldsymbol{\mu}_k, \mathbf{s}_k, \mathbf{q}_k, \mathbf{c}_k, \alpha_k\}_{k=1}^{N}$ of N Gaussians that enables high-fidelity rendering from novel viewpoints away from the capture position, including disoccluded regions not visible in the input view. For the k-th Gaussian, $\boldsymbol{\mu}_k \in \mathbb{R}^3$ is its center, $\mathbf{s}_k \in \mathbb{R}^3$ the anisotropic scale, $\mathbf{q}_k \in \mathbb{R}^4$ a unit quaternion encoding rotation, $\mathbf{c}_k \in \mathbb{R}^3$ the RGB color, and $\alpha_k \in [0, 1]$ the opacity.

The task couples two difficulties: the equirectangular projection (ERP) assigns vastly different solid angles to pixels at different latitudes, and the disoccluded regions that must be completed are distributed differently in every scene. We address both through a three-stage decomposition (Fig. 2). **Stage I**, a *Sphere-Aware Gaussian Initializer*, lifts the panorama into N_{vis} visible Gaussians $\mathcal{G}_{\text{vis}}$ while compensating for the ERP Jacobian, and extracts multi-scale features shared by the later stages. **Stage II**, a *Disocclusion Query Generator*, localizes where new Gaussians are needed and ray-marches behind the visible surface to produce 3D query candidates. **Stage III**, a *Context-Grounded Attribute Decoder*, synthesizes full attributes for these candidates from nearby visible Gaussians and merges them additively with the untouched visible set.

3.2 SPHERE-AWARE GAUSSIAN INITIALIZATION

The ERP mapping is *not area-preserving*: a pixel at latitude ϕ subtends a solid angle proportional to $\cos\phi$, so any initialization that treats pixels uniformly in image space mis-sizes the resulting 3D primitives and over-densifies the poles. Stage I therefore lifts the panorama with explicit compensation for this non-uniform metric.

Spherical Back-Projection. Each pixel (u, v) on the ERP grid corresponds to a unique viewing direction on the unit sphere:

$$\theta = \left(\frac{u}{W} - \frac{1}{2}\right) 2\pi, \quad \phi = \left(\frac{1}{2} - \frac{v}{H}\right) \pi, \\ \mathbf{d}(\theta, \phi) = \big(\cos\phi \sin\theta,\ -\sin\phi,\ \cos\phi\cos\theta\big). \tag{1}$$

We obtain a panoramic depth map D of the input view from an off-the-shelf panoramic depth estimator (Lin et al., 2025). Given D, we instantiate one Gaussian per grid location and index it by k, so that (θ_k, ϕ_k) are the angular coordinates of pixel k and D_k its depth. Each center is initialized via spherical back-projection:

$$\boldsymbol{\mu}_k = D_k \cdot \mathbf{d}(\theta_k, \phi_k). \tag{2}$$

Latitude-Aware Scale Correction. Since the solid angle element satisfies $d\Omega = \cos\phi\, d\theta\, d\phi$, equal-area Gaussians on the sphere must have *unequal* scales in image space: uncorrected polar Gaussians overestimate their tangential extent by a factor of $1/\cos\phi$, producing visible stretching in rendered novel views. We therefore modulate the base scale by latitude:

$$s_k \propto D_k \cdot \cos(\phi_k), \tag{3}$$

where s_k is the isotropic base scale from which the anisotropic $\mathbf{s}_k$ is obtained by learned per-axis residuals, subject to a lower bound that prevents degenerate splats at extreme latitudes; this yields uniform splat density across the sphere regardless of latitude.

Multi-Scale Feature Extraction. To supply rich representations to downstream stages, we employ a pre-trained vision transformer as a frozen encoder coupled with a feature pyramid decoder that fuses multi-scale representations via skip connections. The decoder produces two complementary feature maps: texture features $\mathbf{F}_{\text{tex}}$ encoding appearance information, and geometry features $\mathbf{F}_{\text{geo}}$ encoding structural information. These features serve as shared context for the subsequent stages, feeding the disocclusion detector in Stage II and the query initialization in Stage III.

Learnable Residual Refinement. Back-projection from the conditioning depth alone is not accurate enough for high-quality novel-view rendering, because noise in the depth map directly translates into misplaced Gaussian centers. We therefore attach a lightweight residual head h_ψ on $\mathbf{f}_k$, the concatenation of $\mathbf{F}_{\text{tex}}$ and $\mathbf{F}_{\text{geo}}$ at grid location k, which predicts per-Gaussian residual attributes:

$$(\Delta\theta_k, \Delta\phi_k, \Delta z_k, \Delta\mathbf{s}_k, \Delta\mathbf{q}_k, \Delta\mathbf{c}_k, \Delta\alpha_k) = h_\psi(\mathbf{f}_k). \tag{4}$$

Position residuals are composed in the angular and inverse-depth space to remain consistent with the ERP sampling grid:

$$\begin{aligned} \theta'_k &= \theta_k + \lambda_{xy}\Delta\theta_k, \\ \phi'_k &= \mathrm{clip}\big(\phi_k + \lambda_{xy}\Delta\phi_k,\ -\tfrac{\pi}{2} + \epsilon,\ \tfrac{\pi}{2} - \epsilon\big), \\ D'_k &= \big(\mathrm{softplus}(b_k + \lambda_z\Delta z_k) + \epsilon\big)^{-1}, \end{aligned} \tag{5}$$

where Δz_k is the residual in inverse-depth space, $b_k = \mathrm{softplus}^{-1}(D_k^{-1})$ is the unconstrained parameter of the base inverse depth, λ_{xy} and λ_z bound the residual magnitudes, and ϵ is a small positive constant guarding the polar and zero-depth limits. The refined center is then re-projected as $\boldsymbol{\mu}_k = D'_k \cdot \mathbf{d}(\theta'_k, \phi'_k)$, and the remaining attributes follow bounded residual composition around their initialized values. The head is zero-initialized, so before learning the network exactly reproduces the spherical scaffold; training thus starts from a physically meaningful initialization and only learns local corrections.

Stage I Training. Stage I is trained end-to-end with differentiable rendering. The objective combines (1) a target-view reconstruction loss mixing L1, LPIPS, and SSIM terms, (2) a source-view reconstruction loss that anchors the scaffold to the input panorama, (3) an opacity coverage loss that penalizes transparent renderings on valid pixels, and (4) a hinge regularizer on the position residuals that permits small local corrections while discouraging large deviations from the depth-initialized scaffold.

3.3 Disocclusion Query Generator

Challenge and Insight. Disoccluded regions are *spatially sparse*, covering only a small fraction of target-view pixels, and *geometrically predictable* from the depth discontinuities and occlusion boundaries visible in the input. Completion is therefore better posed as *selective spatial detection* than as dense global hallucination: we first identify *where* new content is needed, then allocate generation capacity exclusively to those locations, leaving the already high-quality visible Gaussians untouched.

Learnable Disocclusion Detection. We localize disocclusion sources on the input ERP grid with a lightweight convolutional detector. Its input concatenates the Stage I feature maps $\mathbf{F}_{\text{tex}}$ and $\mathbf{F}_{\text{geo}}$ with three geometric cues: the normalized inverse depth, its spatial gradients (highlighting occlusion boundaries), and the magnitude of the Stage I residuals (a proxy for reconstruction uncertainty). The detector outputs a per-pixel disocclusion probability

$$p_{\text{disocc}} = \sigma\big(\mathbf{H}_d(\mathbf{F}_{\text{tex}}, \mathbf{F}_{\text{geo}}, D^{-1}, \nabla D^{-1}, |\Delta|)\big), \tag{6}$$

where $\mathbf{H}_d$ denotes the detector, $\sigma(\cdot)$ the logistic sigmoid, and $|\Delta|$ the Stage I residual magnitudes of (4); the map $p_{\text{disocc}} \in [0,1]^{H' \times W'}$ lives on the grid where the Gaussians are instantiated.

Spherically-Uniform Selection. A naive global top-N_{sel} selection on p_{disocc} clusters selections at a few dominant occlusion boundaries and over-samples high latitudes, where ERP pixels are denser per solid angle. We instead adopt a *pooled top-K* scheme: the ERP grid is partitioned into rectangular cells whose 2:1 aspect ratio matches that of the ERP so that each cell covers an approximately equal solid angle; we max-pool p_{disocc} over cells, select the top N_{sel} cells, and keep the highest-probability pixel within each selected cell. This guarantees a minimum spacing between any two selections and distributes the generation budget by solid angle rather than pixel count. Since only selected pixels receive gradient feedback through the rendering losses, a purely greedy top-K would leave early false negatives permanently uncorrected; we therefore replace a small fraction ϵ_{rand} of training selections with random valid pixels for exploration, while inference remains fully deterministic.

Ray-Marched Query Candidates. Occluded surfaces lie *behind* the visible surface along the viewing ray. For each selected pixel with view direction $\mathbf{d}$ and visible depth D, we sample M candidate depths $D^{(1)}, \ldots, D^{(M)}$ uniformly in inverse depth over the interval $\big(D + \epsilon_D,\ \min(\rho_{\text{far}} D,\ D_{\text{max}})\big]$, where ϵ_D is a small offset placing candidates strictly behind the visible surface, ρ_{far} caps the depth extent relative to D, and D_{max} is an absolute far bound:

$$\boldsymbol{\mu}_{\text{query}}^{(j)} = D^{(j)} \cdot \mathbf{d}, \qquad j = 1, \ldots, M, \tag{7}$$

yielding $N_q = N_{\text{sel}} \times M$ query candidates that bracket plausible hidden geometry at multiple depths.

Self-Supervised Training Signal. Training labels are generated online without manual annotation. We render the visible Gaussians $\mathcal{G}_{\text{vis}}$ to held-out target views and mark a pixel as disoccluded if its accumulated opacity falls below a threshold (missing geometry) or its reconstruction error lies in the top quantile (incorrect rendering). Because these labels live on target views while the detector operates on the source ERP, we map the mask back through a pure direction transform: each labeled target pixel is unprojected to a ray, rotated into the source frame, and converted to ERP coordinates, requiring no depth information. A morphological dilation compensates quantization, and far-field pixels beyond a depth cutoff are filtered out since their parallax under camera translation is negligible. The detector is trained with a class-balanced binary cross-entropy against the resulting binary pseudo-label y_{disocc},

$$\mathcal{L}_{\text{select}} = \text{BCE}_{\text{bal}}\big(p_{\text{disocc}},\ y_{\text{disocc}}\big), \tag{8}$$

first alone during a brief warm-up phase, and thereafter jointly with the Stage III decoder, so that pixel selection adapts continuously to the generator it serves.

3.4 Context-Grounded Attribute Decoder

Design Rationale. The final stage must predict complete Gaussian attributes at locations where no direct pixel evidence exists, and follows two principles. Completion is *purely additive*: since the visible Gaussians already reconstruct observed regions with high fidelity, the decoder only synthesizes new Gaussians and never modifies existing ones, preserving input-view quality by construction. Predictions are *grounded in local context*: occluded surfaces typically continue the geometry and appearance of their visible surroundings (walls extend behind furniture, floors continue under tables), so each query derives attribute priors from its nearest visible Gaussians and learns only bounded residual corrections around them.

Visible Context Encoding and Query Initialization. The visible Gaussians are converted into per-point tokens by concatenating their pixel-aligned texture and geometry features with their geometric attributes and Stage I residual statistics, then contextualized by a lightweight point transformer operating on nearest-neighbor graphs in inverse-depth-weighted coordinates. Each query candidate is initialized from a sinusoidal encoding of its spherical coordinates $(\theta, \phi, \log r)$, where $r = \|\boldsymbol{\mu}\|$ is the radial distance from the capture center, together with the texture and geometry features of its parent pixel and its relative depth offset from the visible surface.

Local Cross-Attention Decoding. The decoder stacks L layers, each interleaving two attention operations: (1) *KNN self-attention* among query tokens, enabling coherent predictions across queries that fill the same disoccluded structure, and (2) *KNN cross-attention* from each query to its K_c nearest visible tokens, grounding predictions in observed context. Restricting both attentions to local neighborhoods keeps the cost linear in the number of tokens, avoiding the quadratic scaling of global attention over the full visible set.

Neighbor-Grounded Residual Prediction. Zero-initialized linear heads map each decoded query token to one raw residual output per attribute. Position residuals are parameterized in spherical coordinates around each candidate:

$$\begin{gathered} \Delta\theta = \theta_{\text{max}} \tanh(\delta_\theta), \quad \Delta\phi = \phi_{\text{max}} \tanh(\delta_\phi), \\ r = D\big(1 + \rho_{\text{max}}\, \sigma(\delta_r)\big), \end{gathered} \tag{9}$$

where θ_{max} and ϕ_{max} bound the angular displacement to a few ERP pixels, D is the visible depth at the candidate's parent pixel, and ρ_{max} bounds the depth ratio; the radial parameterization therefore

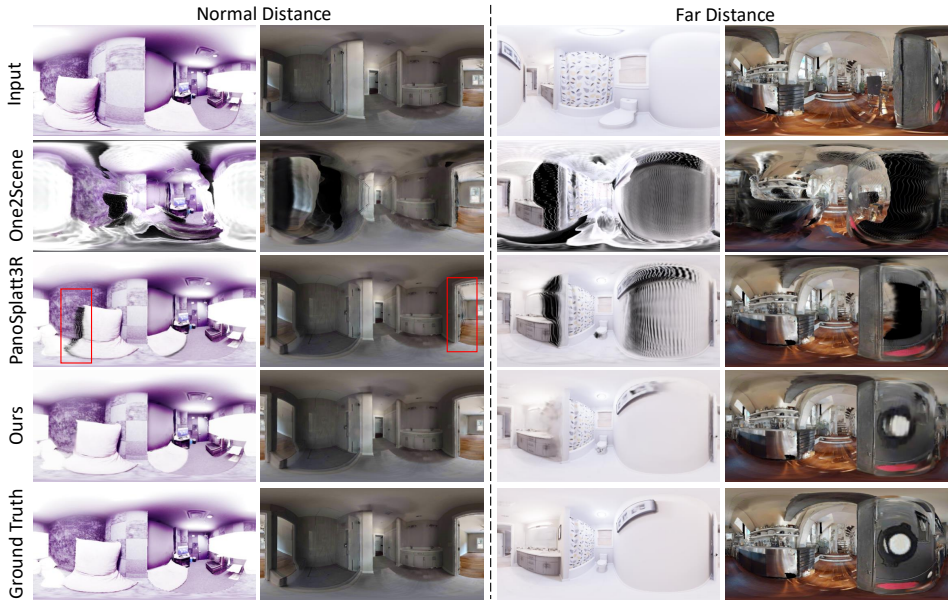

Figure 3: Qualitative comparison on novel view synthesis. One2Scene suffers distortions and black artifacts in disoccluded regions, and PanoSplatt3R exhibits stretched surfaces and stripe artifacts (red boxes), while Pano123 stays close to the ground truth under both protocols.

constrains every generated Gaussian to lie strictly *behind* the visible surface at its parent pixel. The remaining attributes are composed as bounded residuals around distance-weighted priors from the nearest visible Gaussians: additive in color space, multiplicative in log-scale space (with a minimum angular footprint), quaternion composition onto the sign-aligned rotation mean, and logit-space residuals for opacity. Because all heads are zero-initialized, the decoder initially reproduces its neighbor priors at the candidate positions, providing the same stable cold start as Stage I.

Additive Merging. The N_q synthesized Gaussians form the occluded set $\mathcal{G}_{\text{occ}}$, and the final scene concatenates the two sets without any modification of the visible Gaussians:

$$\mathcal{G}_{\text{full}} = \text{concat}\big(\mathcal{G}_{\text{vis}},\ \mathcal{G}_{\text{occ}}\big). \tag{10}$$

No pruning or thresholding is applied: how much each candidate contributes is controlled by its learned opacity, shaped end-to-end by the rendering objectives below.

Training Objectives. Stage III is trained end-to-end with rendering losses on $\mathcal{G}_{\text{full}}$, jointly with the Stage II detector:

$$\begin{aligned}\mathcal{L} = \lambda_{\text{t}}\mathcal{L}_{\text{target}} + \lambda_{\text{d}}\mathcal{L}_{\text{disocc}} + \lambda_{\text{p}}\mathcal{L}_{\text{preserve}} + \lambda_{\alpha}\mathcal{L}_{\alpha} \\ + \lambda_{\text{a}}\mathcal{L}_{\text{anchor}} + \lambda_{\text{o}}\mathcal{L}_{\text{opaque}} + \lambda_{\text{s}}\mathcal{L}_{\text{select}}.\end{aligned} \tag{11}$$

Each λ is a fixed scalar weight: $\mathcal{L}_{\text{target}}$ (combining L1, LPIPS, and SSIM) ensures overall rendering quality on target views; $\mathcal{L}_{\text{disocc}}$ concentrates supervision on disoccluded regions through the same dual-source mask used for pseudo-labels, with a linear ramp-up of its weight so that freshly initialized Gaussians are not prematurely dragged into disoccluded areas; $\mathcal{L}_{\text{preserve}}$ enforces input-view consistency by penalizing photometric and geometric deviation at the source viewpoint; $\mathcal{L}_{\alpha}$ supervises rendered opacity against ground-truth coverage; $\mathcal{L}_{\text{anchor}}$ is a hinge regularizer keeping angular and radial residuals close to their candidates; $\mathcal{L}_{\text{opaque}}$ encourages generated Gaussians to become fully opaque rather than translucent ghosts; and $\mathcal{L}_{\text{select}}$ is the detection loss of (8).

4 Experiments

4.1 Experimental Setup

Datasets. We train on HM3D (Ramakrishnan et al., 2021), which provides large-scale indoor environments with accurate mesh geometry; following the scene-level train/test split of Splatter-360 (Chen et al., 2025), all stages are trained exclusively on the HM3D training scenes. Evaluation

Table 1: Novel view synthesis comparison under *Normal* and *Far Distance* camera translations. Best results in **bold**.

		HM3D			Replica		
Protocol	Method	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Normal	One2Scene	16.55	0.8218	0.4107	17.48	0.8876	0.3043
	PanoSplatt3R	22.12	0.8931	**0.2419**	**24.91**	**0.9374**	**0.1554**
	Pano123 (Ours)	**25.13**	**0.9154**	0.2538	24.35	0.9364	0.1820
Far	One2Scene	13.87	0.7680	0.4951	14.68	0.8462	0.3749
	PanoSplatt3R	17.50	0.8351	0.3450	19.94	0.8976	0.2306
	Pano123 (Ours)	**22.49**	**0.8921**	**0.3291**	**22.45**	**0.9181**	**0.2272**

Table 2: Ablation of the disocclusion completion (Stages II and III). *Stage I only* renders the visible Gaussians without any completion. Best results in **bold**.

		HM3D			Replica		
Protocol	Configuration	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Normal	Stage I only	24.03	0.9042	0.2799	23.32	0.9208	0.1936
	Full model	**25.13**	**0.9154**	**0.2538**	**24.35**	**0.9364**	**0.1820**
Far	Stage I only	20.13	0.8600	0.3556	20.29	0.9043	0.2311
	Full model	**22.49**	**0.8921**	**0.3291**	**22.45**	**0.9181**	**0.2272**

is performed on two test sets with high-quality ground-truth geometry: the held-out HM3D test scenes, and Replica (Straub et al., 2019). Replica is never seen during training and therefore serves as a cross-dataset generalization test. All panoramas are at 512×1024 equirectangular resolution, and each test scene provides a panoramic video trajectory. For each dataset we randomly sample 10,000 (source, target) view pairs: source frames are drawn uniformly from all test frames (with replacement when fewer than 10,000 are available), and the target frame is selected at a random frame gap within the protocol's range, with circular wrap-around along the trajectory. Metrics are averaged over all pairs.

Baselines. We compare against two feed-forward baselines. One2Scene (Wang et al., 2026a) decomposes the input panorama into multiple perspective sub-images and applies independent feed-forward 3D prediction to each, lacking cross-view geometric consistency. PanoSplatt3R (Ren et al., 2025) is a panoramic reconstruction method designed for multi-view input; we adapt it to the single-image setting by providing duplicated views.

Metrics. We report PSNR, SSIM, and LPIPS (Zhang et al., 2018) computed on rendered target views. We evaluate under two camera displacement protocols defined by the frame gap between source and target views along the capture trajectory: *Normal Distance* (gap sampled from $[5, 50]$ frames) tests near-field novel view quality, while *Far Distance* (gap sampled from $[30, 70]$ frames) probes robustness under large extrapolation where significant disocclusion occurs.

Implementation Details. We use the frozen backbone from DAP (Lin et al., 2025), coupled with a DPT decoder producing 256×512 feature maps. Stage I is trained first and kept frozen while Stages II and III are trained jointly, using the objectives described in the Method section. Training runs on 8 GPUs with the AdamW optimizer, a batch size of 4, and per-module learning rates between 1×10^{-4} and 5×10^{-4}, linear warm-up followed by cosine decay. Complete architecture and training hyperparameters are provided in Appendix A.

4.2 Quantitative Results

Tab. 1 shows that Pano123 achieves substantial improvements over One2Scene on both datasets: +8.58 dB PSNR on HM3D and +6.87 dB on Replica under Normal Distance, with gains maintained at +8.62 dB and +7.77 dB under Far Distance. The performance gap widens in perceptual quality: LPIPS improves by 38% (HM3D) and 40% (Replica) under Normal Distance.

We attribute the large margin to two fundamental limitations of One2Scene. First, One2Scene processes each perspective sub-image independently; this leads to seam artifacts and incoherent depth at boundaries between adjacent views. Second, One2Scene lacks an explicit disocclusion mechanism; regions invisible in the input view are simply absent from its output, leading to holes and stretched artifacts in novel views. In contrast, Pano123 addresses both issues through sphere-aware Gaussian placement (correcting ERP geometry) and targeted disocclusion completion (synthesizing content only where needed). Notably, the performance gap is *larger* under Far Distance, where disoccluded regions expand significantly, confirming that our selective completion strategy provides increasing benefit as novel view difficulty grows.

The comparison against PanoSplatt3R is protocol dependent and reveals a complementary pattern. Under Normal Distance, Pano123 leads on HM3D by +3.01 dB PSNR, while PanoSplatt3R attains higher scores on Replica, where small camera displacements keep most target pixels visible in the

input and reward pure visible-surface reconstruction. Under Far Distance, however, Pano123 surpasses PanoSplatt3R on every metric of both datasets (+4.99 dB PSNR on HM3D and +2.51 dB on Replica). This pattern matches our design focus: PanoSplatt3R reconstructs visible surfaces faithfully but has no explicit disocclusion mechanism, so its quality drops rapidly once translation exposes unobserved regions, whereas our localize-then-complete strategy sustains quality under large viewpoint changes.

4.3 Ablation Study

We isolate the contribution of the disocclusion completion pipeline by comparing the full model against a *Stage I only* variant in Tab. 2: the latter renders the visible Gaussians $\mathcal{G}_{\text{vis}}$ directly, without the Stage II query generator or the Stage III attribute decoder.

Analysis. Because Stage III is purely additive and never modifies the visible Gaussians, any difference between the two rows is attributable entirely to the Gaussians generated in disoccluded regions. The completion stages deliver consistent improvements under both protocols, and the gains grow with camera displacement: +1.10 dB PSNR (HM3D) and +1.03 dB (Replica) under Normal Distance versus +2.36 dB and +2.16 dB under Far Distance, consistent with the expanding area of disoccluded regions. SSIM and LPIPS improve consistently across all settings, confirming that the synthesized Gaussians raise both structural and perceptual quality rather than merely filling pixels. The qualitative comparison in Fig. 4 confirms that the completed regions are clearly more coherent than the holes left by the Stage I-only variant.



Figure 4: Qualitative ablation under Far Distance. The Stage I-only variant leaves holes in disoccluded regions, which the full model fills with coherent content while keeping visible regions untouched.

4.4 Qualitative Results

Fig. 3 presents qualitative comparisons across representative scenes under both distance protocols. Pano123 produces geometrically coherent completions in disoccluded regions, where One2Scene exhibits severe distortions and black artifacts, and PanoSplatt3R leaves stretched surfaces and stripe artifacts that grow once the camera moves far from the capture position. The sphere-aware parameterization prevents visible artifacts near the poles, and the selective strategy focuses generation capacity on disoccluded regions while leaving visible Gaussians untouched, preserving input fidelity in well-observed areas.

Fig. 4 isolates the effect of the completion stages under Far Distance, where disocclusion is most pronounced. Without them, regions occluded in the input render as holes; the full model fills these regions with content that continues the surrounding geometry and appearance, while the visible regions remain identical by construction, demonstrating the effectiveness of the angular residual prediction and the neighbor-grounded attribute priors.

5 Conclusion

We presented Pano123, a feed-forward framework that reconstructs a complete 3D Gaussian scene from a single equirectangular panorama via a *localize-then-complete* strategy, combining sphere-aware Gaussian initialization, self-supervised disocclusion localization, and context-grounded additive completion that preserves visible Gaussians. Experiments on HM3D and Replica show substantial gains over One2Scene and consistent improvements over PanoSplatt3R under large camera displacements, validating the importance of explicit disocclusion handling. A current limitation is the fixed completion-token budget, which may be insufficient under extensive occlusion; future work will investigate adaptive token allocation and multi-panorama reconstruction.

AI USE STATEMENT

In this work, we have not used generative AI tools for any task with required disclosure; such tasks are not applicable to this work. We used generative AI tools only to assist in sourcing information and identifying relevant literature during the related-work survey. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

David Charatan, Sizhe Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. *arXiv preprint arXiv:2312.12337*, 2023. 2, 3

Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European conference on computer vision*, pp. 370–386. Springer, 2024a. 2, 3

Yuedong Chen, Chuanxia Zheng, Haofei Xu, Bohan Zhuang, Andrea Vedaldi, Tat-Jen Cham, and Jianfei Cai. Mvsplat360: Feed-forward 360 scene synthesis from sparse views. *Advances in Neural Information Processing Systems*, 37:107064–107086, 2024b. 2, 3

Zheng Chen, Chenming Wu, Zhelun Shen, Chen Zhao, Weicai Ye, Haocheng Feng, Errui Ding, and Song-Hai Zhang. Splatter-360: Generalizable 360° gaussian splatting for wide-baseline panoramic images. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21590–21599. IEEE Computer Society, 2025. 2, 3, 7

Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4578–4587, 2018. 3

Mengyang Feng, Jinlin Liu, Miaomiao Cui, and Xuansong Xie. Diffusion360: Seamless 360 degree panoramic image generation based on diffusion models. *arXiv preprint arXiv:2311.13141*, 2023. 3

Sheng-Yu Huang, Zi-Ting Chou, and Yu-Chiang Frank Wang. 3d gaussian inpainting with depth-guided cross-view consistency. In *Proceedings of the computer vision and pattern recognition conference*, pp. 26704–26713, 2025a. 3

Zilong Huang, Jun He, Junyan Ye, Lihan Jiang, Weijia Li, Yiping Chen, and Ting Han. Scene4u: Hierarchical layered 3d scene reconstruction from single panoramic image for your immerse exploration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26723–26733, 2025b. 3

Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *ACM Transactions on Graphics (TOG)*, 44(6):1–16, 2025. 2, 3

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2, 4

Yuseung Lee, Kunho Kim, Hyunjin Kim, and Minhyuk Sung. Syncdiffusion: Coherent montage via synchronized joint diffusions. *Advances in Neural Information Processing Systems*, 36:50648–50660, 2023. 3

Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. *Advances in neural information processing systems*, 36:21878–21894, 2023. 3

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. 4, 8

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and L Repaint Van Gool. Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11461–11471, 2023. 3

Lars Mescheder, Wei Dong, Shiwei Li, Xuyang Bai, Marcel Santos, Peiyun Hu, Bruno Lecouat, Mingmin Zhen, AmaÃĢl Delaunoy, Tian Fang, et al. Sharp monocular view synthesis in less than a second. *arXiv preprint arXiv:2512.10685*, 2025. 2, 3

Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G Derpanis, Jonathan Kelly, Marcus A Brubaker, Igor Gilitschenski, and Alex Levinshtein. Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20669–20679, 2023. 3

Santhosh K Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. *arXiv preprint arXiv:2109.08238*, 2021. 7

Jiahui Ren, Mochu Xiang, Jiajun Zhu, and Yuchao Dai. Panosplatt3r: Leveraging perspective pretraining for generalized unposed wide-baseline panorama reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 28959–28969, 2025. 2, 3, 8

Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 8

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2149–2159, 2022. 3

Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2024. 3

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*, 2026a. 2, 3, 8

Qiwei Wang, Xianghui Ze, Jingyi Yu, and Yujiao Shi. Cylindersplat: 3d gaussian splatting with cylindrical triplanes for panoramic novel view synthesis. *arXiv preprint arXiv:2603.05882*, 2026b. 3

Tianhao Wu, Chuanxia Zheng, and Tat-Jen Cham. Panodiffusion: 360-degree panorama outpainting via diffusion. *arXiv preprint arXiv:2307.03177*, 2023. 3

Shuai Yang, Jing Tan, Mengchen Zhang, Tong Wu, Gordon Wetzstein, Ziwei Liu, and Dahua Lin. Layerpano3d: Layered 3d panorama for hyper-immersive scene generation. In *Proceedings of the special interest group on computer graphics and interactive techniques conference conference papers*, pp. 1–10, 2025. 3

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4578–4587, 2021. 3

Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6658–6667, 2024. 3

Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5916–5926, 2025. 3

Cheng Zhang, Haofei Xu, Qianyi Wu, Camilo Cruz Gambardella, Dinh Phung, and Jianfei Cai. Pansplat: 4k panorama synthesis with feed-forward gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11437–11447, 2025. 3

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 conference on computer vision and pattern recognition*, pp. 586–595, 2018. 8

Dian Zheng, Cheng Zhang, Xiao-Ming Wu, Cao Li, Chengfei Lv, Jian-Fang Hu, and Wei-Shi Zheng. Panorama generation from nfov image done right. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 21610–21619, 2025. 3

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

A Appendix

This appendix provides the complete set of architecture and training hyperparameters (Appendix A.1) and additional method details omitted from the main text due to space constraints (Appendix A.2). All notation follows Sec. 3.

A.1 Implementation Details

A.1.1 Architecture Hyperparameters

Table 3 lists all architecture hyperparameters of the three stages. We detail the design choices below.

Stage I. Gaussians are instantiated on a 256×512 grid obtained by striding the 512×1024 input by a factor of 2. The residual factors $\lambda_{xy} = \lambda_z = 0.001$ rescale the raw network outputs before composition, acting as an effective learning-rate reduction on the position channels; analogous factors of 0.1 (color) and 1.0 (scale, rotation, opacity) are applied to the remaining attribute residuals. The anisotropic scale is composed as the latitude-corrected base scale of Eq. (3) multiplied by a learned factor bounded to $[0, 10]$, and the latitude term $\cos\phi_k$ is clamped to a minimum of 0.1 to prevent degenerate splats at the poles.

Stage II. The pooled top-K selection uses cells of 2×4 grid pixels, matching the 2:1 width-to-height aspect ratio of the ERP so that each cell covers an approximately equal solid angle. $N_{\text{sel}} = 10{,}000$ cells are selected per panorama, and $M = 2$ candidate depths are sampled per selected ray, yielding $N_q = 20{,}000$ query candidates.

Stage III. Visible Gaussians are converted to 128-dimensional tokens by a 4-block point transformer with a KNN neighborhood of 16. Query positional encodings use sinusoidal embeddings with 8 frequency bands over $(\theta, \phi, \log r)$. The residual bounds of Eq. (9) are $\theta_{\max} = \phi_{\max} = 8$ pixels of the 512×1024 input ERP (≈ 0.049 rad) and $\rho_{\max} = 2.0$, constraining every generated Gaussian to lie strictly behind the visible surface at its parent pixel, at a depth of at most $3\times$ the visible depth. The remaining attribute residuals are bounded to ± 1.0 in log-scale space and ± 0.2 in color space; attribute priors are aggregated from the 8 nearest visible Gaussians with distance-based Gaussian-kernel weighting.

A.1.2 Training Loss Weights

For Stage I, the target-view reconstruction loss combines L1, LPIPS (VGG backbone), and SSIM terms with weights 1.0, 1.0, and 0.2; the source-view reconstruction L1 uses weight 1.0; the opacity coverage loss uses weight 0.01; and the hinge regularizer on position residuals uses weight 1.0. For the joint Stage II+III phase, the weights of Eq. (11) are $\lambda_{\text{t}} = 0.02$, $\lambda_{\text{d}} = 1.0$, $\lambda_{\text{p}} = 0.01$, $\lambda_{\alpha} = 0.2$, $\lambda_{\text{a}} = 0.1$, $\lambda_{\text{o}} = 0.01$, and $\lambda_{\text{s}} = 1.0$. Within $\mathcal{L}_{\text{target}}$, the LPIPS and SSIM terms are weighted by 0.2

Table 3: Architecture hyperparameters of Pano123.

Stage	Hyperparameter	Value
I	Input panorama resolution	512×1024
	Gaussian grid resolution $H' \times W'$	256×512
	Position residual factors λ_{xy}, λ_z	0.001, 0.001
	Scale multiplier range	$[0, 10]$
II	Selected cells N_{sel}	10,000
	Pooling cell size (h $\times$ w)	2×4
	Exploration ratio ϵ_{rand}	0.1
	Candidate depths per ray M	2
	Total query candidates N_q	20,000
	Depth interval bounds ρ_{far}, D_{max}	3.0, 50 m
III	Visible-token encoder blocks	4
	Token / hidden dimension	128
	Encoder KNN neighborhood	16
	Decoder layers L	4
	Attention heads / MLP ratio	4 / 4
	Self-attention KNN neighbors	8
	Cross-attention neighbors K_c	32
	Attribute prior KNN neighbors	8

and 0.04 relative to the L1 term; within $\mathcal{L}_{\text{preserve}}$, the inverse-depth consistency term is weighted by 0.1 relative to the photometric term.

A.2 Additional Method Details

A.2.1 Ray-Marched Candidate Sampling

For each selected pixel with visible depth D, the M candidate depths are sampled uniformly in inverse depth over the interval $\big(D + \epsilon_D,\ \min(\rho_{\text{far}} D,\ D_{\text{max}})\big]$ with

$$\epsilon_D = \max(0.05\, D,\ 0.05\,\text{m}), \tag{12}$$

$\rho_{\text{far}} = 3.0$, and $D_{\text{max}} = 50\,\text{m}$. The relative offset guarantees that candidates are placed strictly behind the visible surface at all depth ranges, while the absolute floor of 0.05 m prevents vanishing offsets at very close surfaces. Sampling uniformly in inverse depth allocates more candidates near the visible surface, where disoccluded geometry is statistically most likely to reside.

A.2.2 Pseudo-Label Generation

The self-supervised training signal of Stage II (Eq. (8)) is generated online at every training step as follows.

1. **Visible-only rendering.** The frozen Stage I Gaussians $\mathcal{G}_{\text{vis}}$ are rendered to the held-out target view, producing a color image and an accumulated opacity map α_{vis}.
2. **Disocclusion marking.** A target pixel is marked as disoccluded if $\alpha_{\text{vis}} < 0.5$, indicating missing geometry, or if its photometric error against the ground truth lies in the top 20% per image, indicating incorrect rendering.
3. **Reprojection to the source ERP.** Each marked target pixel is unprojected to a viewing ray, rotated into the source camera frame, and converted to ERP coordinates on the 256×512 Gaussian grid. This is a pure direction transform and requires no depth.
4. **Morphological dilation.** The reprojected binary mask is dilated with a 5×5 kernel to compensate for quantization on the coarse grid.
5. **Far-field filtering.** Pixels whose source depth exceeds a hard cutoff of 50 m are removed; among the remaining marked pixels, those beyond the 90th depth percentile (computed per sample within the marked region) are additionally removed, since far geometry exhibits negligible parallax under camera translation. If fewer than 10,000 marked pixels survive filtering, the unfiltered mask is used instead to guarantee a sufficient positive-label budget.

The resulting binary map y_{disocc} supervises the detector through the class-balanced binary cross-entropy of Eq. (8), where the positive and negative sets are re-weighted inversely to their pixel counts so that sparse disocclusion regions are not dominated by the abundant negatives.

A.2.3 DUAL-SOURCE DISOCCLUSION MASK FOR $\mathcal{L}_{\text{DISOCC}}$

The disocclusion-focused rendering loss $\mathcal{L}_{\text{disocc}}$ concentrates supervision on the regions the completion stages are responsible for. Its per-pixel mask follows the same dual-source criterion as the pseudo-labels of the previous subsection, evaluated on the target view: (1) *coverage failures*, pixels with visible-only accumulated opacity $\alpha_{\text{vis}} < 0.5$ (missing geometry), and (2) *rendering failures*, pixels whose photometric error against the ground truth lies in the top 20% per image (incorrect geometry or appearance). The masked L1 error is normalized by the mask area, so scenes with small disoccluded regions are not under-weighted.

A.2.4 ATTRIBUTE PRIOR COMPOSITION

Each decoded query composes its final attributes as bounded residuals around priors aggregated from its 8 nearest visible Gaussians, weighted by a distance-based Gaussian kernel whose bandwidth is set per query to the median neighbor distance:

- **Color**: additive residual around the weighted color mean, bounded to ± 0.2 by a scaled $\tanh$.
- **Scale**: multiplicative residual in log-space around the weighted scale mean, with the log-multiplier bounded to ± 1.0; a minimum angular footprint prevents degenerate splats.
- **Rotation**: the neighbor quaternions are sign-aligned to a reference hemisphere before averaging, and the learned residual is composed onto the normalized mean.
- **Opacity**: additive residual in logit space around the weighted opacity mean.

All residual heads are zero-initialized linear layers, so at initialization every generated Gaussian exactly reproduces its neighbor priors at the candidate position, which stabilizes the early phase of joint training.