PANODPG: DECOUPLED POSE AND GEOMETRY ESTIMATION FOR HIGH-RESOLUTION PANORAMIC 3D RECONSTRUCTION

Anonymous authors
Paper under double-blind review

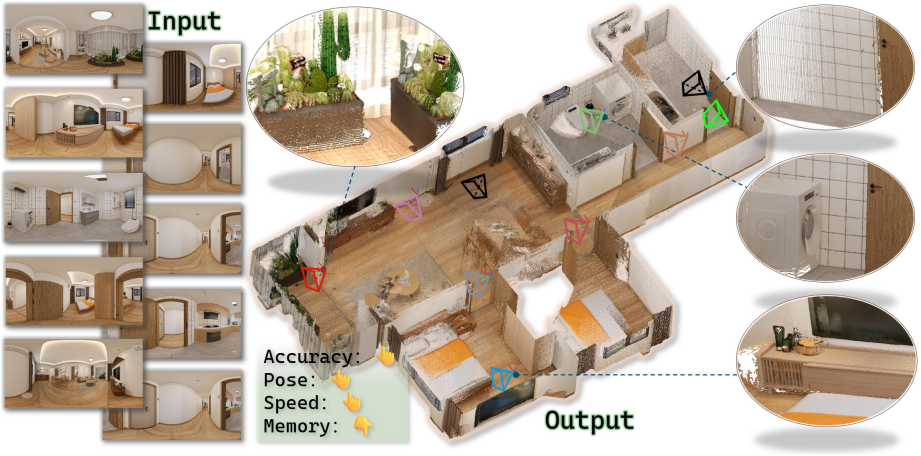


Figure 1: **PanoDPG** recovers coherent 3D scenes with fine geometric details and accurate camera poses from multi-view panoramas, while maintaining low computational and memory costs.

ABSTRACT

Panoramic imagery captures 360° scene context for applications such as robotics and virtual reality. However, its wide field of view limits local detail at low resolution, making high-resolution inputs important for accurate, high-fidelity 3D reconstruction. Feed-forward panoramic 3D reconstruction becomes computationally expensive at high resolution, especially when pose estimation and dense geometry reconstruction share global interactions. Our key observation is that **increasing spatial resolution substantially improves geometric accuracy, while providing only limited gains in pose accuracy.** This motivates task-specific processing for pose and geometry. We propose PanoDPG, a dual-branch framework that performs global camera reasoning on low-resolution equirectangular images and detailed geometry reconstruction on high-resolution cubemap views. This design preserves the broad panoramic context needed for pose estimation while reducing spherical distortion and retaining fine-grained details for geometry prediction. The two branches communicate through register tokens, which also exchange context across cubemap groups without all-to-all interaction among high-resolution tokens. Experiments on multiple datasets show that PanoDPG achieves competitive or state-of-the-art reconstruction accuracy with substantially reduced computational cost for high-resolution panoramicreconstruction.

1 INTRODUCTION

Recovering poses and dense geometry from multi-view images is fundamental to robotic perception and digital mapping Cadena et al. (2016); Koestler et al. (2022); Li et al. (2026b). Feed-forward methods, including VGGT Wang et al. (2025), π^3 Wang et al. (2026c), Depth Anything 3 Lin et al.

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.

(2025), and VGGT-Ω Wang et al. (2026b), jointly predict poses and 3D structures using a single network. Panoramic imagery provides a 360° field of view and scene context (Ai et al., 2025; Gao et al., 2022). PanoVGGT Guo et al. (2026), Argus Li et al. (2026a), and Wid3R Jung et al. (2026) extend this paradigm to panoramic or wide-field imagery. However, distributing pixels over the surrounding scene reduces local angular sampling density under comparable pixel budgets, making high-resolution inputs important for recovering geometric details.

Joint architectures share representations and cross-frame interactions between pose and geometry prediction Wang et al. (2025; 2026b). For N views containing L tokens each, dense global attention scales as $\mathcal{O}((NL)^2)$. Increasing resolution raises global interaction costs for both tasks, despite their different requirements: pose estimation benefits from scene-level context, whereas dense geometry requires local features. A shared projection introduces another compromise. Equirectangular projections (ERPs) preserve the panoramic layout but introduce nonuniform stretching Ling et al. (2023), while local perspective views reduce distortion but require communication across viewing directions. Geometry reconstruction requires cross-view context, but obtaining it need not involve dense global interaction among all high-resolution tokens Wang et al. (2026a).

Our key insight is that **pose and geometry benefit differently from resolution and representation.** Our resolution study in Fig. 2 shows that point-cloud accuracy improves with increasing input resolution, whereas pose accuracy peaks at 1036×518. Beyond resolution, ERP represents the full panoramic context in a single image, while cubemaps provide local perspective views with reduced projection distortion. These properties motivate using ERP for global pose reasoning and cubemaps for detailed geometry recovery. This choice is supported by Holo360D Ou et al. (2026), which reports higher pose AUC@30 with ERP inputs and lower geometric reconstruction errors with perspective split views.

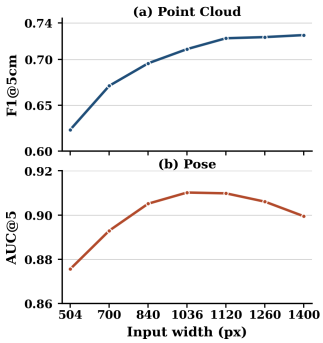


Figure 2: Resolution sensitivity. Higher input resolution consistently improves panoramic point-cloud reconstruction accuracy, while pose accuracy peaks at 1036×518.

This asymmetry suggests that geometric detail can be improved without proportionally increasing the computational burden of pose estimation. Efficient attention studies further support compact alternatives to dense global interaction Sun et al. (2026); Wang et al. (2026b); Wang & Xu (2026); Wang et al. (2026a). We therefore recast **high-resolution panoramic reconstruction as task-specific pose and geometry inference coupled through context exchange and scale alignment.** We reorganize joint reconstruction by assigning global pose reasoning to low-resolution panoramas and detailed geometry recovery to high-resolution local views, connected through communication and scale coordination.

We propose **PanoDPG**, a **task-decoupled dual-branch framework**. Low-resolution ERP features support global multi-view pose reasoning, while high-resolution cubemap features preserve geometric details. A **register-token-based communication module** exchanges context across representations and cubemap groups, supporting reconstruction with scene-level information. Experiments demonstrate state-of-the-art high-resolution panoramic reconstruction with low computational overhead.

Our contributions are summarized as follows:

- We propose **PanoDPG**, a **task-decoupled framework** combining low-resolution ERP pose estimation with high-resolution cubemap geometry reconstruction, tailoring resolution, projection, and interaction to each task for efficient panoramic reconstruction.
- We develop **compact cross-branch and cross-group communication** using register tokens to share global context across pose and geometry branches and cubemap groups without dense attention over all high-resolution tokens.
- Extensive experiments demonstrate that PanoDPG achieves state-of-the-art performance in high-resolution panoramic 3D reconstruction while maintaining low computational overhead.

2 Related Work

2.1 Feed-Forward Multi-View Reconstruction

Traditional multi-view reconstruction relies on iterative SfM and MVS pipelines, including feature matching, pose estimation, triangulation, and bundle adjustment Lowe (2004); Schonberger & Frahm (2016); Yao et al. (2018); Wang et al. (2021); Hartley et al. (2003). Recent feed-forward methods replace these stages with a single neural network. DUSt3R Wang et al. (2024) performs dense point-map prediction for image pairs, while VGGT Wang et al. (2025) extends this idea to multi-view inputs and jointly predicts camera poses, depth, and point maps. π^3 Wang et al. (2026c) improves robustness to input-view ordering through permutation-equivariant processing, and subsequent works such as Depth Anything 3 Lin et al. (2025) and VGGT-Ω Wang et al. (2026b) further explore ray-based representations and model scaling. However, these methods are mainly designed for perspective images and become less effective on panoramic inputs with spherical distortion and high token costs.

2.2 Panoramic Geometry Estimation

Panoramic geometry estimation commonly operates either directly on equirectangular projections (ERPs) using distortion-aware or spherical representations, or on multiple perspective views obtained through cubemap or tangent-plane projection Cao & Wang (2024); Eder & Frahm (2019); Sun et al. (2021); Tateno et al. (2018); Yun et al. (2023). VGGT-360 Yuan et al. (2026a) instead applies adaptive perspective projection and correlation-based correction to obtain geometry-consistent panoramic depth. Although these methods effectively reduce projection distortion, they mainly focus on single-panorama geometry estimation.

2.3 Panoramic Multi-View Reconstruction

Recent methods extend feed-forward reconstruction to panoramic multi-view inputs. PanoVGGT Guo et al. (2026) jointly predicts poses, depth, and point maps from unordered panoramas using spherical-aware positional embeddings, while Wid3R Jung et al. (2026) supports a broader range of wide-field-of-view cameras through ray-based representations and camera-model tokens. Argus Li et al. (2026a) further investigates metric panoramic reconstruction with large-scale indoor RGB-D data. Nevertheless, existing methods generally use a unified resolution and interaction pattern for pose and geometry estimation. In contrast, PanoDPG adopts task-specific branches: low-resolution ERP features with global interaction for pose estimation, and high-resolution cubemap features for detailed geometry reconstruction. Register-token interaction is further introduced to preserve consistency between the two branches.

3 Methodology

3.1 Framework Overview

Given a sequence of N panoramic images at 2K resolution, our goal is to jointly estimate the camera poses and dense 3D point cloud. Existing panoramic reconstruction methods Guo et al. (2026); Li et al. (2026a); Jung et al. (2026) typically perform pose and geometry estimation using a unified, relatively low-resolution feature representation. Directly scaling this unified paradigm to high-resolution inputs is computationally prohibitive, as the substantially increased number of visual tokens greatly amplifies the cost of dense global interaction Liu et al. (2021); Shen et al. (2025); Yuan et al. (2026b). As discussed in the introduction, pose and geometry benefit differently from resolution and representation: The former primarily relies on cross-view relationships and is more robust to resolution variations and spherical distortion, whereas the latter requires fine-grained local details and a panoramic representation with minimal projection distortion.

These mismatches motivate PanoDPG, a task-decoupled dual-branch architecture for panoramic pose estimation and geometry reconstruction. Specifically, the pose branch operates on low-resolution ERP inputs and performs global multi-view interaction to estimate camera poses, whereas the geometry branch processes high-resolution cubemap features to recover fine-grained structures while mitigating spherical distortion Wang et al. (2020); Bai et al. (2024). The two branches are coupled through a register-based interaction module to preserve geometric consistency.

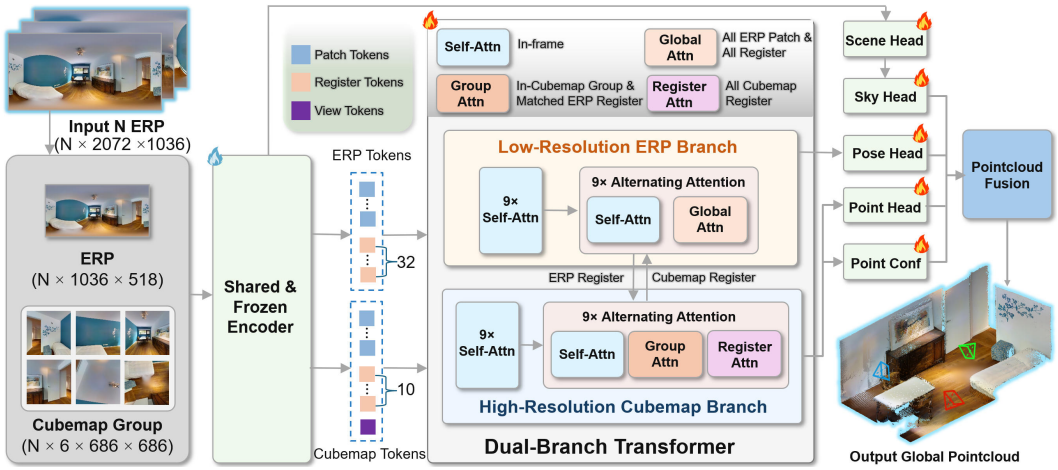


Figure 3: Overview of PanoDPG. A shared encoder and dual-branch transformer process ERP and cubemap inputs with register-token communication. Prediction heads estimate poses, point maps, confidence, scene type, and sky masks.

As illustrated in Fig. 3, each input panorama (2072×1036, 2K) is downsampled to a low-resolution ERP image of size (1036×518, 1K) for pose estimation, while its original 2K panorama is projected into a high-resolution cubemap group comprising six 686×686 faces for geometry reconstruction. A shared and frozen visual encoder extracts features from both representations. The low-resolution ERP features are processed by the ERP branch, which performs global cross-view interaction to capture scene-level context and estimate camera poses. In parallel, the high-resolution cubemap features are processed by the cubemap branch to preserve fine-grained visual details, reduce the impact of projection distortion, and predict dense local geometry.

These branch-specific features are decoded by task-specific heads to predict camera poses, local point maps and point confidence. We attach lightweight scene-classification and sky-mask heads to the pretrained DINOv2 features Oquab et al. (2023) to distinguish between indoor and outdoor scenes and filter out invalid sky points from the final point map.

3.2 Shared Feature Encoding

For all ERP images and cubemap faces, we adopt the pretrained π^3 encoder to extract features and keep its parameters frozen throughout training. Holo360D Ou et al. (2026) demonstrates that the frozen π^3 encoder provides sufficient representations for accurate pose estimation from ERP inputs and geometry reconstruction from perspective inputs. We therefore share the frozen encoder across the two inputs, avoiding the memory and computation required to back-propagate.

Let $\mathbf{X}_i^{\text{erp}}$ denote tokens extracted from the i-th ERP image, and let $\mathbf{X}_{i,f}^{\text{cube}}$ denote tokens extracted from the f-th face of the i-th cubemap group. We organize the encoded features as follows:

$$\mathbf{F}_i^{\text{erp}} = [\mathbf{X}_i^{\text{erp}}, \mathbf{R}_i^{\text{erp}}],$$

$$\mathbf{F}_i^{\text{cube}} = \left[\mathbf{X}_{i,1}^{\text{cube}}, \mathbf{R}_{i,1}^{\text{cube}}, \mathbf{v}_{i,1}^{\text{cube}}, \mathbf{X}_{i,2}^{\text{cube}}, \mathbf{R}_{i,2}^{\text{cube}}, \mathbf{v}_{i,2}^{\text{cube}}, \ldots, \mathbf{X}_{i,6}^{\text{cube}}, \mathbf{R}_{i,6}^{\text{cube}}, \mathbf{v}_{i,6}^{\text{cube}}\right].$$

where $\mathbf{F}_i^{\text{erp}}$ and $\mathbf{F}_i^{\text{cube}}$ denote tokens organized from the i-th ERP image and the matched cubemap group. $\mathbf{R}$ represents register tokens Darcet et al. (2024), and $\mathbf{v}$ represents a view token associated with an individual cubemap face. Following the framework in Fig. 3, the ERP tokens contain a small set of ERP register tokens, while cubemap tokens contain cubemap register tokens and a view token. The register tokens are learnable latent tokens that aggregate scene-level information and serve as compact communication interfaces between branches and views.

3.3 Task-Decoupled Dual-Branch Transformer

At this stage, the organized ERP tokens $\mathbf{F}_i^{\text{erp}}$ and cubemap-group tokens $\mathbf{F}_i^{\text{cube}}$ are fed into the task-decoupled dual-branch Transformer. The two branches use task-specific resolutions and attention

patterns for pose and geometry estimation, respectively. Their main patch-token computation remains decoupled, while compact register tokens facilitate communication across branches.

Low-resolution ERP pose branch. As shown in Fig. 3, the ERP branch operates on low-resolution ERP features to capture inter-frame correspondence for camera pose estimation. It consists of nine intra-frame self-attention layers followed by nine alternating-attention Wang et al. (2025).

The first nine layers apply self-attention independently within each ERP frame to model interactions among its patch and register tokens. Through this intra-frame interaction, the 32 ERP register tokens aggregate information from the correspondence patch tokens, providing a compact representation of each frame for subsequent in-branch and cross-branch interaction. Each subsequent alternating-attention block comprises an intra-frame self-attention layer and a global-attention layer. Specifically, the global-attention layer jointly processes the patch and register tokens from all ERP frames, together with the register tokens from all cubemap groups. It thereby establishes cross-frame correspondence and injects geometric context from the cubemap branch.

High-resolution cubemap geometry branch.

The cubemap branch processes high-resolution features for dense point-cloud estimation with low impact of spherical distortion. It first applies nine layers of intra-frame self-attention independently to each cubemap face. These layers model their local geometric features. The branch then employs nine alternating-attention blocks, each consisting of self-attention, group attention, and register attention. Self-attention continues to preserve the representation within each face. Group attention is performed over the tokens from the same cubemap group together with the register tokens from the matched ERP frame, enabling cross-face geometric interaction and injecting compact scene-level context from the ERP branch. Register attention is applied to the register tokens of all cubemap groups, facilitating global information propagation across groups while avoiding dense interaction among all high-resolution patch tokens.

3.4 Geometric Consistency Enhancement

PanoDPG decouples pose estimation and geometry reconstruction considering their respective requirements. However, the two branches must remain geometrically coupled: the ERP pose should be aligned with the cubemap point maps in scale, and the point maps from different cubemap groups should be consistent. A straightforward solution is to apply patch-token attention both between the ERP and cubemap branches and across different cubemap groups. However, such interaction would largely eliminate the computational advantage of the task-decoupled design for high-resolution inputs. Instead of relying on dense patch-token interactions, PanoDPG efficiently enforces global geometric consistency using two complementary mechanisms: Register-token-based interaction.

Cross-branch interaction. The two branches exchange information by incorporating register tokens from the other branch as additional keys and values. In each ERP global-attention layer, register tokens from all cubemap groups are included as keys and values, allowing ERP queries to access geometry context across frames. In each cubemap group-attention layer, only the ERP register tokens from the correspondence frame are included as keys and values, providing panoramic context to that cubemap group. Each ERP frame contains 32 register tokens, while each cubemap group contains 60, with 10 allocated to each face.

Intra-branch register attention. Within the cubemap branch, dedicated register attention operates over register tokens from all cubemap groups. All participating register tokens serve as queries, keys, and values, allowing each to aggregate information from the others for cross-group communication. We additionally introduce a view-specific token for each cubemap face to enhance the modeling of relationships among faces within each cubemap group.

4 Training Objective

We train PanoDPG with complementary geometric losses that supervise local point maps, ERP camera poses, globally registered point maps, and cubemap-view poses:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{local}} + \mathcal{L}_{\text{ERP}} + 0.5\mathcal{L}_{\text{global}} + 0.1\mathcal{L}_{\text{persp}}. \tag{1}$$

A single sequence-level scale s normalizes the predictions, while s^{gt} normalizes the ground truth. Both scales are shared across all frames and cubemap views, preventing independent per-view normalization from concealing relative scale inconsistencies.

Local point-map loss. We directly supervise local geometry in each cubemap camera coordinate system, providing a reconstruction objective that does not depend on the predicted ERP poses:

$$\mathcal{L}_{\mathrm{local}} = \frac{1}{|\Omega|} \sum_{(t,v,p)\in\Omega} \left\| \frac{\hat{\mathbf{X}}_{t,v,p}}{s} - \frac{\mathbf{X}_{t,v,p}}{s^{\mathrm{gt}}} \right\|_1, \tag{2}$$

where Ω is the set of valid pixels, and $\hat{\mathbf{X}}_{t,v,p}$ and $\mathbf{X}_{t,v,p}$ are the predicted and ground-truth points at pixel p of cubemap view v in frame t.

Camera pose losses. The ERP pose loss supervises the camera transformations needed to register panoramic frames. The perspective pose loss supervises cubemap-view poses across frames, encouraging the register-attention pathway to learn cross-view correspondences that support geometric alignment. For a camera transformation $\mathbf{T} = [\mathbf{R} \mid \mathbf{t}]$, we define

$$d_{\mathrm{pose}}(\hat{\mathbf{T}}, \mathbf{T}) = \|\hat{\mathbf{R}} - \mathbf{R}\|_1 + \lambda_{\mathrm{trans}} \left\| \frac{\hat{\mathbf{t}}}{s} - \frac{\mathbf{t}}{s^{\mathrm{gt}}} \right\|_1, \tag{3}$$

where $\lambda_{\mathrm{trans}} = 100$ and $\|\cdot\|_1$ denotes the element-wise L_1 norm. The pose losses are

$$\mathcal{L}_{\mathrm{ERP}} = \frac{1}{T} \sum_{t=1}^{T} d_{\mathrm{pose}} \left(\hat{\mathbf{T}}_t^{\mathrm{ERP}}, \mathbf{T}_t^{\mathrm{ERP}} \right), \qquad \mathcal{L}_{\mathrm{persp}} = \frac{1}{TV} \sum_{t=1}^{T} \sum_{v=1}^{V} d_{\mathrm{pose}} \left(\hat{\mathbf{T}}_{t,v}^{\mathrm{persp}}, \mathbf{T}_{t,v}^{\mathrm{persp}} \right). \tag{4}$$

where T is the number of panoramic frames and $V = 6$. Ground-truth cubemap-view poses are derived from the ERP poses and fixed cubemap rotations, providing geometrically compatible supervision for the two branches.

Global point loss. Separate supervision of local geometry and camera poses does not directly penalize errors in their composed reconstruction. We therefore transform each local point prediction into a common ERP reference coordinate system:

$$\hat{\mathbf{X}}_{t,v,p}^{g} = \hat{\mathbf{R}}_t^{\mathrm{ERP}} \mathbf{R}_v^{c} \hat{\mathbf{X}}_{t,v,p} + \hat{\mathbf{t}}_t^{\mathrm{ERP}}, \tag{5}$$

where $\mathbf{R}_v^c$ maps cubemap view v to the ERP camera frame, and the six cubemap views share the ERP camera center. The registered points are then supervised by

$$\mathcal{L}_{\mathrm{global}} = \frac{1}{|\Omega|} \sum_{(t,v,p)\in\Omega} \left\| \frac{\hat{\mathbf{X}}_{t,v,p}^{g}}{s} - \frac{\mathbf{X}_{t,v,p}^{g}}{s^{\mathrm{gt}}} \right\|_1. \tag{6}$$

Here, $\mathbf{X}_{t,v,p}^{g}$ denotes the corresponding ground-truth point in the ERP reference coordinate system. This loss jointly constrains the pose and geometry predictions through their composed transformation, promoting consistency across branches and cubemap views.

5 Experiments

5.1 Experiment Setup

Training Datasets. For training, we combine PanoCity Guo et al. (2026), which contains synthetic outdoor scenes, Holo360D Ou et al. (2026), which contains real-world indoor and outdoor scenes; Matterport3D Chang et al. (2017) and Stanford2D3D Armeni et al. (2017), which contain real-world indoor scenes. To account for differences in dataset size and scene distribution, we sample training data from Holo360D, Matterport3D, PanoCity, and Stanford2D3D at a fixed ratio of 3:3:3:1, respectively. This mixed-data strategy exposes the model to diverse indoor and outdoor environments while preventing the largest dataset from dominating the training process.

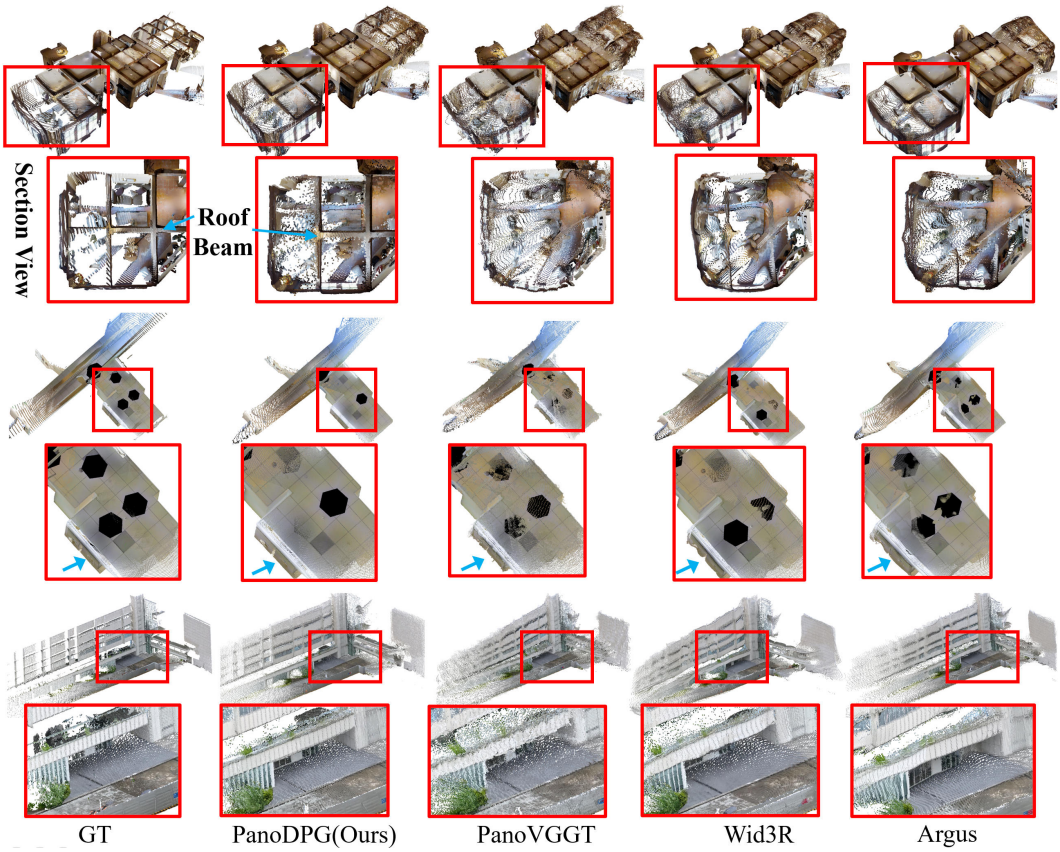


Figure 4: Qualitative comparison of panoramic 3D reconstruction on indoor and outdoor scenes. We compare PanoDPG with PanoVGGT, Wid3R, and Argus, with enlarged regions highlighting local structures and geometric details such as roof beams and floor boundaries.

Evaluation Datasets and Evaluation Metrics. For evaluation, we benchmark PanoDPG on four panoramic datasets: Matterport3D, PanoCity, Stanford2D3D, and Holo360D. We follow the dataset splits used by PanoVGGT Guo et al. (2026) for Matterport3D, PanoCity, and Stanford2D3D, while adopting the official split of Holo360D Ou et al. (2026).

For Matterport3D, PanoCity, and Stanford2D3D, we evaluate each sequence using up to five panoramic frames. Sequences with five or fewer frames use all available frames; otherwise, five frames are randomly sampled. For Holo360D, we partition each long sequence chronologically into 15-frame subsequences and randomly sample five frames from each subsequence for evaluation.

We evaluate camera pose accuracy and point cloud reconstruction quality. For camera pose, we report the area under the accuracy curve (AUC) at pose-error thresholds of 5° (AUC@5), together with the mean rotation error (R-mean) and mean translation error (T-mean). For reconstructed point clouds, we report accuracy (Acc), completeness (Comp), which jointly measure geometric precision. All metrics are computed consistently across the four datasets.

5.2 Qualitative and Quantitative Evaluation

Comparative Baselines. We compare PanoDPG with three recent feed-forward reconstruction methods, including PanoVGGT Guo et al. (2026), Argus Li et al. (2026a), Wid3R Jung et al. (2026). All methods are evaluated on the four datasets described above. For each method, we evaluate both its publicly released pretrained checkpoint and a version independently fine-tuned on the same training splits used to train PanoDPG.

Due to GPU memory constraints, we train PanoVGGT, Argus, and Wid3R at 1036×518, matching PanoDPG's training frame-count range of 2 to 10. To ensure a fair comparison at a common output resolution, all methods are evaluated at 1036×518. PanoDPG's predictions are also projected onto a 1036×518 ERP grid before computing evaluation metrics against the ground truth.

Evaluation. We evaluate PanoDPG under two input settings: the standard configuration using 2072×1036 panoramas and a lower-resolution configuration using 1036×518 panoramas. In the latter, inputs to both the ERP and cubemap branches are derived from the same low-resolution panoramas to match the baselines' input resolution.

As shown in Tab. 1 and Tab. 2, PanoDPG achieves state-of-the-art or highly competitive performance on both camera pose estimation and point cloud reconstruction across most datasets. Notably, changing PanoDPG's input resolution from 2072×1036 to 1036×518 causes only marginal changes in pose and point-cloud accuracy. In contrast, PanoVGGT exhibits a substantial performance degradation under the 2072×1036 setting. This result suggests that PanoDPG is less sensitive to changes in input resolution.

Moreover, as illustrated in Fig. 4, PanoDPG produces high-fidelity visual reconstructions in both indoor and outdoor scenes, outperforming previous methods in terms of structural completeness and visual quality. We attribute these improvements to its ability to scale to high-resolution inputs and its use of lower-distortion cubemaps for geometric reconstruction.

Table 1: Quantitative evaluation of camera pose estimation on Holo360D, Matterport3D, Stanford2D3D, and PanoCity. We report AUC@5↑, mean rotation error (Rmean↓), and mean translation error (Tmean↓). Pink and orange indicate the best and second-best results, respectively.

	Holo360D			Matterport3D			Stanford2D3D			PanoCity		
Method	AUC@5↑	Rmean↓	Tmean↓	AUC@5↑	Rmean↓	Tmean↓	AUC@5↑	Rmean↓	Tmean↓	AUC@5↑	Rmean↓	Tmean↓
Argus	0.033	19.848	17.610	0.792	2.448	1.545	0.832	5.778	5.015	0.768	0.692	2.241
Argus*	0.781	0.996	1.723	0.917	1.316	1.237	0.952	0.602	0.580	0.943	0.314	0.683
Wid3R	0.445	2.396	3.777	0.906	1.771	0.948	0.830	4.381	2.129	0.000	1.044	87.252
Wid3R*	0.674	0.857	2.645	0.825	0.826	1.252	0.829	0.917	1.184	0.975	0.216	0.486
PanoVGGT(1036×518)	0.737	1.219	2.445	0.942	0.679	0.594	0.956	0.859	0.851	**0.997**	0.098	0.148
PanoVGGT*(1036×518)	0.883	0.539	1.133	**0.969**	0.538	**0.476**	0.945	2.058	1.191	**0.997**	0.097	**0.142**
PanoVGGT* (2072×1036)	0.004	33.760	33.015	0.005	33.342	26.772	0.006	36.505	26.336	0.133	10.441	14.682
PanoDPG (1036×518)	0.902	**0.380**	1.036	0.968	0.470	0.487	**0.976**	0.427	0.504	0.992	0.097	0.248
PanoDPG (2072×1036)	**0.903**	**0.380**	**1.035**	**0.969**	**0.465**	0.481	0.975	**0.423**	**0.500**	0.993	**0.096**	0.247

Table 2: Quantitative evaluation of point-cloud reconstruction. Acc. and Comp. denote accuracy and completeness errors, respectively, for which lower values are better. A superscript * denotes a retrained model, while the resolution in parentheses indicates the input resolution used during inference. Pink and orange backgrounds indicate the best and second-best results, respectively.

	Holo360D		Matterport3D		Stanford2D3D		PanoCity	
Method	Acc.↓	Comp.↓	Acc.↓	Comp.↓	Acc.↓	Comp.↓	Acc.↓	Comp.↓
Argus	0.837	0.819	0.085	0.066	0.186	0.175	0.917	0.899
Argus*	0.182	0.163	0.048	0.044	**0.033**	0.032	0.238	0.227
Wid3R	0.288	0.237	0.062	0.049	0.200	0.129	1.000	1.000
Wid3R*	0.169	0.126	0.037	0.027	0.037	**0.026**	0.242	0.210
PanoVGGT(1036×518)	0.185	0.147	0.064	0.027	0.060	0.049	0.156	0.139
PanoVGGT*(1036×518)	0.115	0.084	0.032	**0.021**	0.082	0.073	0.154	0.139
PanoVGGT* (2072×1036)	0.966	0.959	0.821	0.793	0.915	0.900	0.988	0.981
PanoDPG (1036×518)	0.117	0.097	**0.031**	0.026	0.035	0.033	**0.112**	0.180
PanoDPG (2072×1036)	**0.105**	**0.076**	0.032	**0.021**	0.034	0.028	0.143	**0.133**

5.3 Ablation Study

To evaluate the contribution of each proposed component, we conduct an ablation study on Holo360D, with results summarized in Tab. 3. We consider the following configurations: (1) a single-branch ERP model as the baseline; (2) a dual-branch model without register-based interaction (RBI); (3) the dual-branch model augmented with our RBI. The consistent improvements support the effectiveness of our

Table 3: Ablation study of PanoDPG, evaluating the dual-branch design, register-token interaction (RBI). Lower Acc. and Comp. errors are better.

Configuration	Point Cloud		Camera Pose		
	Acc.↓	Comp.↓	AUC@5↑	Rmean↓	Tmean↓
Single-branch ERP	0.947	0.969	**0.903**	0.398	1.037
Dual-branch	0.126	0.089	0.819	0.655	1.612
Dual-branch + RBI	**0.105**	**0.076**	**0.903**	**0.380**	**1.035**

Table 4: Maximum number of input frames supported for training at 2K resolution on an NVIDIA A800 GPU with 80 GB memory.

Method	Mem.	Max. Frames	Scale-up
Argus	80 GB	2	1.0×
Wid3R	80 GB	3	1.5×
PanoVGGT	80 GB	4	2.0×
PanoDPG	**80 GB**	**10**	**5.0×**

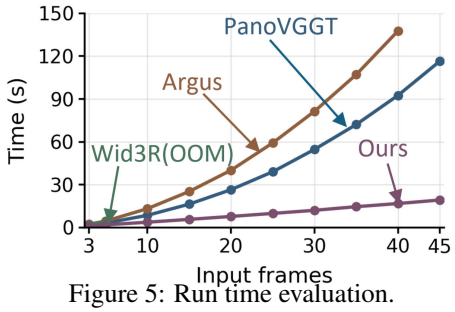


Figure 5: Run time evaluation.

dual-branch design. Adding RBI further validates the proposed interaction scheme, demonstrating that lightweight register-token communication enables effective information exchange and promotes geometric consistency across branches and cubemap groups.

5.4 Efficiency Evaluation

We evaluate training capacity and inference speed for panoramic reconstruction at 2072×1036 resolution. Table 4 reports the maximum number of input frames supported during training on a single NVIDIA A800 GPU with 80 GB of memory. PanoDPG supports 10 frames, compared with 2 for Argus, 3 for Wid3R, and 4 for PanoVGGT, enabling training with $2.5\times$ as many frames as PanoVGGT under the same GPU memory budget.

We further measure inference time with 3 to 45 input frames. As shown in Fig. 5, PanoDPG consistently achieves the lowest latency across all tested sequence lengths. With 45 frames, it achieves approximately $6.0\times$ speedup over the next-fastest method, PanoVGGT. Together, these results demonstrate the efficiency of our task-decoupled dual-branch architecture, which supports scaling to higher input resolutions and longer sequences while enabling more frames during training and faster inference.

6 Conclusion

We presented PanoDPG, a task-decoupled framework for efficient high-resolution panoramic 3D reconstruction. Our key observation is that camera pose estimation and dense geometry reconstruction benefit from different input resolutions, feature representations, and interaction patterns. Based on this insight, PanoDPG estimates globally consistent camera poses from low-resolution ERP features while recovering fine-grained geometry from high-resolution cubemap features. Register-token interactions facilitate efficient information exchange between the two branches and across cubemap groups, thereby promoting geometric consistency without incurring expensive dense feature interactions. Extensive experiments demonstrate that PanoDPG achieves accurate pose estimation and high-quality geometric reconstruction across diverse indoor and outdoor panoramic datasets. Meanwhile, its memory-efficient design supports training with longer input sequences and scales effectively to higher resolutions, while also enabling faster inference. These results highlight the effectiveness and scalability of task-decoupled modeling and provide a practical solution for large-scale panoramic 3D reconstruction.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and revision, language editing, LaTeX preparation, and cross-section consistency checks. AI tools also supported discussions and refinement of methodological descriptions and interpretations of experimental results. AI-generated suggestions were treated as draft material requiring author verification and were not used as experimental evidence or as independent validation of mathematical, algorithmic, or empirical claims. All retained AI-assisted content, including technical descriptions, references, and interpretations of results, was manually checked by the authors against the implementation, experimental records, and cited literature. The pretrained models used within PanoDPG are components of the evaluated method, and their roles are described in Section 3. The authors take full responsibility for the research design, implementation, experiments, results, references, and conclusions, including the accuracy of all AI-assisted content retained in the final manuscript.

ETHICS STATEMENT

This work studies panoramic 3D reconstruction using the existing PanoCity, Holo360D, Matterport3D, and Stanford2D3D datasets. It does not involve new data collection, human-subject experiments, identification of individuals, or inference of personal attributes. Nevertheless, real-world panoramic imagery and reconstructed 3D scenes may contain people, private indoor spaces, location information, or other privacy-sensitive content. The use and redistribution of these datasets and derived reconstructions should therefore comply with the original datasets' licenses, access conditions, and applicable privacy requirements. Reconstruction and pose-estimation errors may also affect downstream applications such as robotic perception, navigation, digital mapping, and virtual reality. PanoDPG does not provide a safety guarantee and should be independently validated, together with appropriate privacy and failure-handling safeguards, before deployment in safety-critical or privacy-sensitive settings.

REPRODUCIBILITY STATEMENT

(This section is **recommended** and does not count toward the page limit.)

Section 3 describes the PanoDPG architecture, input representations and resolutions, dual-branch attention patterns, and register-token-based communication mechanisms. Section 4 specifies the training objective and geometric supervision. Section 5.1 reports the training and evaluation datasets, dataset sampling ratios, evaluation splits, sequence construction procedures, and metrics. Section 5.2 describes the evaluated baselines, fine-tuning protocol, and input- and output-resolution alignment used for fair comparison. Sections 5.3 and 5.4 provide component ablations and the efficiency and scalability evaluation protocols, respectively. These descriptions, together with the reported experimental configurations, are intended to support reproduction and independent verification of the results.

REFERENCES

Hao Ai, Zidong Cao, and Lin Wang. A survey of representation learning, optimization strategies, and applications for omnidirectional vision: H. ai et al. *International Journal of Computer Vision*, 133 (8):4973–5012, 2025.

Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

Jiayang Bai, Haoyu Qin, Shuichang Lai, Jie Guo, and Yanwen Guo. Glpanodepth: Global-to-local panoramic depth estimation. *IEEE Transactions on Image Processing*, 33:2936–2949, 2024.

Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332, 2016.

Zidong Cao and Lin Wang. Monocular 360 depth estimation via spherical fully-connected crfs. *IEEE Robotics and Automation Letters*, 10(2):1409–1416, 2024.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International conference on learning representations*, volume 2024, pp. 2632–2652, 2024.

Marc Eder and Jan-Michael Frahm. Convolutions on spherical images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–5, 2019.

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.

Yijing Guo, Mengjun Chao, Luo Wang, Tianyang Zhao, Haizhao Dai, Yingliang Zhang, Jingyi Yu, and Yujiao Shi. Panovggt: Feed-forward 3d reconstruction from panoramic imagery. *arXiv preprint arXiv:2603.17571*, 2026.

Richard Hartley, Andrew Zisserman, et al. *Multiple view geometry in computer vision*, volume 2. Cambridge university press Cambridge, 2003.

Dongki Jung, Jaehoon Choi, Adil Qureshi, Somi Jeong, Dinesh Manocha, and Suyong Yeon. Wid3r: Wide field-of-view 3d reconstruction via camera model conditioning. *arXiv preprint arXiv:2602.05321*, 2026.

Lukas Koestler, Nan Yang, Niclas Zeller, and Daniel Cremers. Tandem: Tracking and dense mapping in real-time using deep multi-view stereo. In *Conference on Robot Learning*, pp. 34–45. PMLR, 2022.

Xi Li, Linyuan Li, Yan Wu, Tong Rao, Kai Zhang, Xinchen Hui, and Cihui Pan. Argus: Metric panoramic 3d reconstruction for indoor scenes. *arXiv preprint arXiv:2606.30047*, 2026a.

Zhuoxiao Li, Xinyi Liu, Taoyu Wu, Yinrui Ren, Tongyan Hua, Ou Jing, Shuai Zhang, Dongli Wu, Rongjun Qin, Ge Lin Kan, et al. Skyanchor: Updating metric-scale aerial 3d gaussian scenes from unposed ground-view sequences. *arXiv preprint arXiv:2609.13903*, 2026b.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Zhixin Ling, Zhen Xing, Xiangdong Zhou, Manliang Cao, and Guichun Zhou. Panoswin: a pano-style swin transformer for panorama understanding. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17755–17764. IEEE, 2023.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9992–10002. Ieee, 2021.

David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Jing Ou, Zidong Cao, Yinrui Ren, Zhuoxiao Li, Jinjing Zhu, Tongyan Hua, Shuai Zhang, Hui Xiong, and Wufan Zhao. Holo360d: A large-scale real-world dataset with continuous trajectories for advancing panoramic 3d reconstruction and beyond. *arXiv preprint arXiv:2604.22482*, 2026.

Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4104–4113, 2016.

You Shen, Zhipeng Zhang, Yansong Qu, Xiawu Zheng, Jiayi Ji, Shengchuan Zhang, and Liujuan Cao. Fastvggt: Training-free acceleration of visual geometry transformer. *arXiv preprint arXiv:2509.02560*, 2025.

Cheng Sun, Min Sun, and Hwann-Tzong Chen. Hohonet: 360 indoor holistic understanding with latent horizontal features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2573–2582, 2021.

Xianbing Sun, Zhikai Zhu, Zhengyu Lou, Bo Yang, Jinyang Tang, Liqing Zhang, He Wang, and Jianfu Zhang. Avggt: Rethinking global attention for accelerating vggt. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 251–260, 2026.

Keisuke Tateno, Nassir Navab, and Federico Tombari. Distortion-aware convolutional filters for dense prediction in panoramic images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 707–722, 2018.

Chung-Shien Brian Wang, Christian Schmidt, Jens Piekenbrinck, and Bastian Leibe. Block-sparse global attention for efficient multi-view geometry transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14546–14555, 2026a.

Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14189–14198. IEEE, 2021.

Fu-En Wang, Yu-Hsuan Yeh, Min Sun, Wei-Chen Chiu, and Yi-Hsuan Tsai. Bifuse: Monocular 360 depth estimation via bi-projection fusion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 459–468. IEEE, 2020.

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

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. VGGT-Ω. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026b.

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

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. pi^3: Permutation-equivariant visual geometry learning. In *International Conference on Learning Representations*, volume 2026, pp. 10481–10497, 2026c.

Zipeng Wang and Dan Xu. Flashvggt: Efficient and scalable visual geometry transformers with compressed descriptor attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21826–21835, 2026.

Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *European conference on computer vision*, pp. 785–801. Springer, 2018.

Jiayi Yuan, Haobo Jiang, De Wen Soh, and Na Zhao. Vggt-360: Geometry-consistent zero-shot panoramic depth estimation. *arXiv preprint arXiv:2603.18943*, 2026a.

Shuai Yuan, Yantai Yang, Xiaotian Yang, Xupeng Zhang, Zhonghao Zhao, Lingming Zhang, and Zhipeng Zhang. Infinitevggt: Visual geometry grounded transformer for endless streams. *arXiv preprint arXiv:2601.02281*, 2026b.

Ilwi Yun, Chanyong Shin, Hyunku Lee, Hyuk-Jae Lee, and Chae Eun Rhee. Egformer: Equirectangular geometry-biased transformer for 360 depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6101–6112, 2023.

A VISUALIZATION

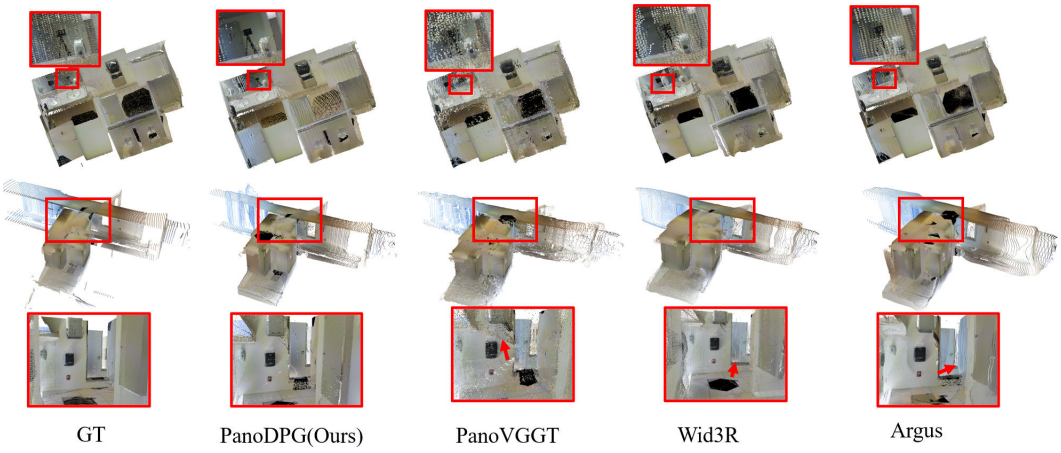


Figure 6: Visualization example 1.

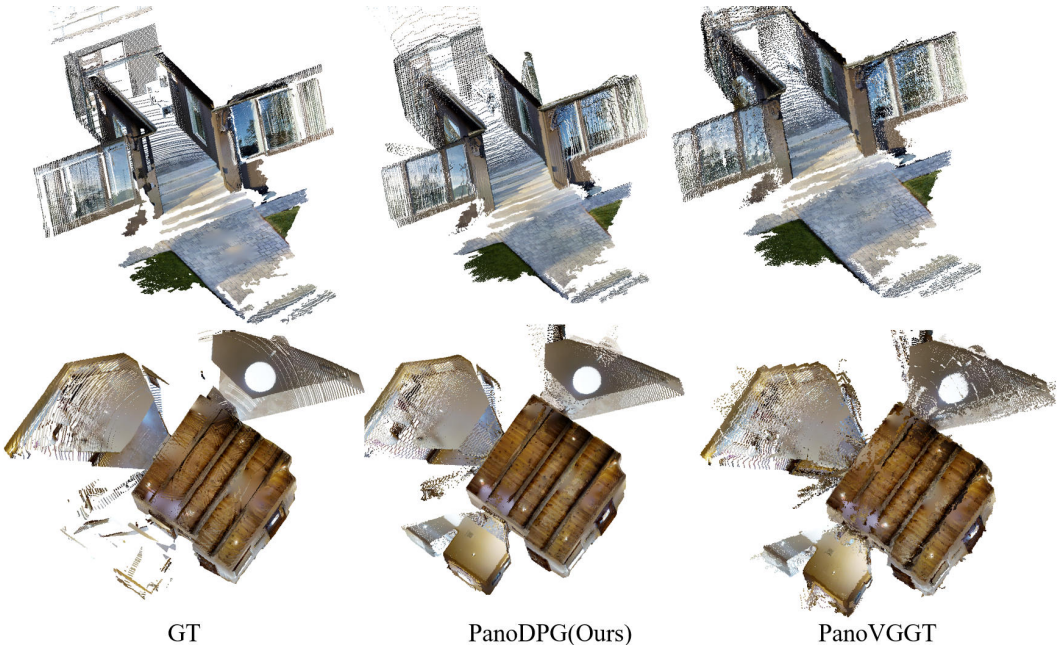


Figure 7: Visualization example 2.

A.1 TRAINING DETAILS OF DUALPI3

The model contains an ERP branch and a perspective branch built from the large Pi3 decoder. Both branches share the same VGGT image encoder, which is initialized from the pretrained VGGT weights and kept frozen during this training stage. The decoder uses RoPE positional encoding (`rope100`) and gradient checkpointing. The perspective encoder is evaluated in chunks of eight images to limit peak memory usage.

The model receives an ERP panorama at resolution 1036×518. Each ERP frame is additionally projected into six perspective views, each of size 686×686. The frame-aware register design supports at most ten ERP frames and sixteen perspective groups. Each ERP frame has a dedicated frame identity embedding; the corresponding six perspective views reuse the same frame identity and



Figure 8: Visualization example 3.

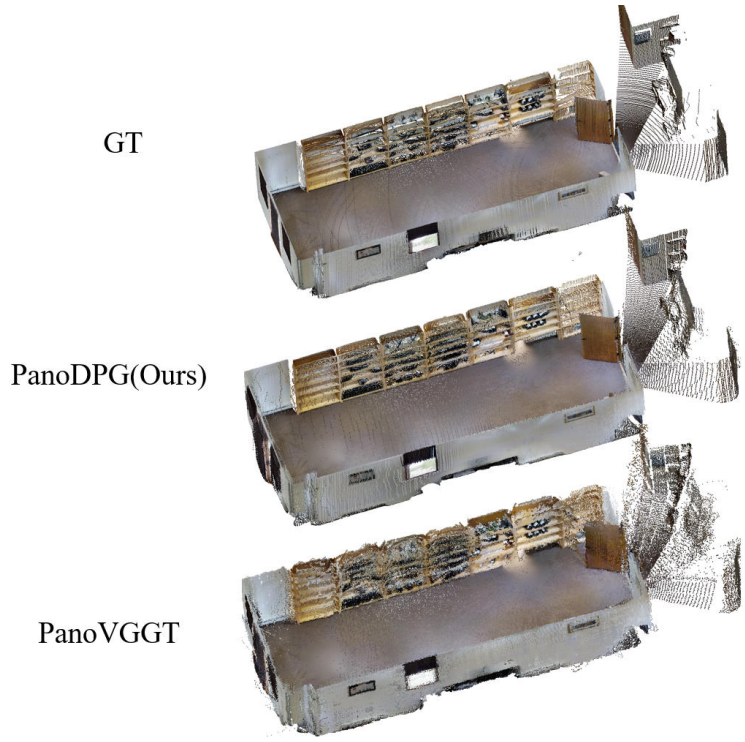


Figure 9: Visualization example 4.

receive an additional view identity embedding. The two branches exchange register tokens through the decoder. Register-mediated cross-branch attention is enabled from decoder block 18 onward, while the first half of the alternating global-attention blocks is removed to reduce computation. Perspective-register global attention is also enabled from block 18 onward.

Training data and sampling. Training uses a mixed dataset consisting of CO3D-V2, Matterport3D-2K, PanoCity, and Stanford2D3D. The sampling probabilities are respectively 0.30, 0.30, 0.30, and 0.10. The training split uses the following source locations and split manifests:

Each training sample contains five ERP frames in the underlying dataset format, while the dynamic batch sampler selects an effective sequence length $F \sim \mathcal{U}\{2, \ldots, 10\}$ for each iteration. The same F is used for all samples in a synchronized global batch. The maximum number of ERP frames assigned to each GPU is ten, so the local batch size is chosen as

$$B_{\text{local}} = \max\left(1, \left\lfloor \frac{10}{F} \right\rfloor\right). \tag{7}$$

Scenes are sampled without replacement within a global batch whenever the dataset provides enough valid sequences. The dynamic sampler generates 1000 iterations per epoch, and all ranks construct the same frame-count and scene sampling plan before taking their rank-specific slice.

During training, ERP panoramas are augmented with a random yaw, pitch, and roll perturbation. The augmentation probability is 0.5, with maximum angles of $45°$, $15°$, and $10°$, respectively. RGB inputs also use mild JPEG, loss, and blur augmentation. Background masking is enabled for the training data. Random crops and focal-length augmentation are disabled. The training seed is 2024 for dataset construction and 666 for the training sampler and worker initialization.