PXDEPTH: LEARNING PIXEL-SPACE FEATURES FOR STRUCTURE-PRESERVING MONOCULAR DEPTH ESTIMATION

Anonymous authors
Paper under double-blind review

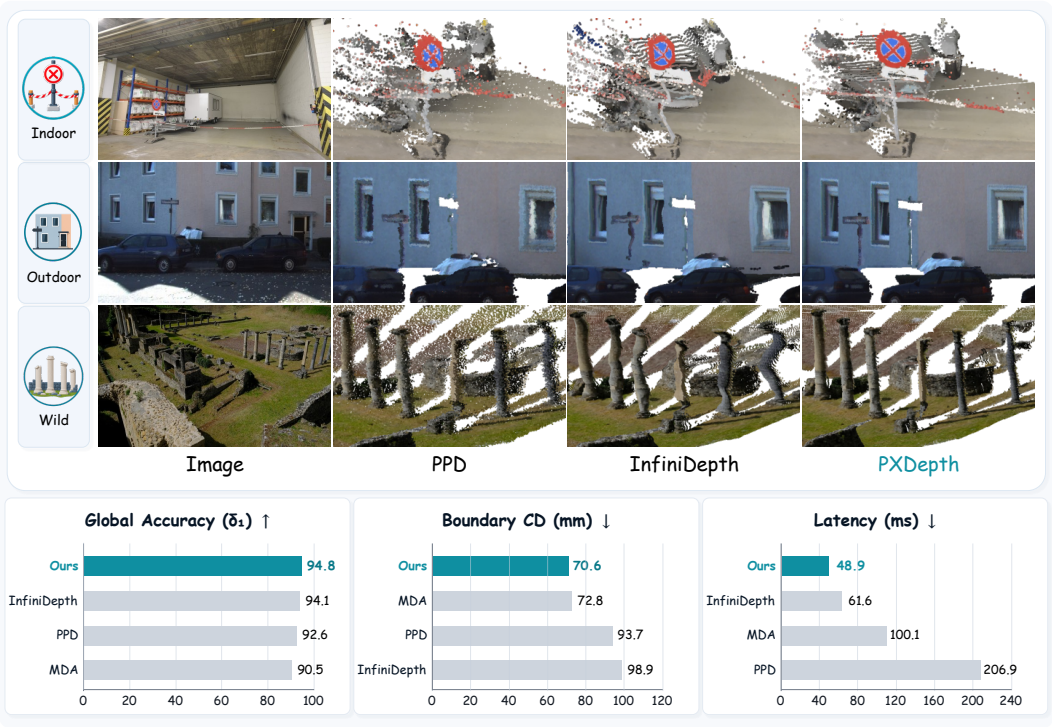


Figure 1. We present **PXDEPTH**, a discriminative monocular depth estimation model that learns pixel-space features for structure-preserving depth prediction. Compared with structure-aware methods (Yu et al., 2026a; Bian et al., 2026; Xu et al., 2025), PXDEPTH better preserves local structures without sacrificing global accuracy, while requiring less inference time.

ABSTRACT

Recent monocular depth estimators achieve strong zero-shot generalization, yet often struggle to preserve fine-grained structures. We attribute this limitation to the prevalent combination of large-patch ViT encoders and convolutional decoders. Coarse tokenization can weaken pixel-level cues, while convolutional decoding struggles to model low-frequency regions and high-frequency details simultaneously. To address this issue, we propose PXDEPTH, a discriminative monocular depth model that separates global context modeling from pixel-space feature learning for depth prediction. Specifically, a large-patch ViT captures global scene context, while a *Pixel-Space Depth Predictor* composed of *Context-Modulated Pixel Transformer blocks* learns pixel-space features throughout depth estimation. This design preserves fine structures without sacrificing global depth consistency. Across diverse zero-shot benchmarks, PXDEPTH substantially improves local geometric fidelity while retaining competitive global depth accuracy and lower inference latency than multi-step generative methods.

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.

1 INTRODUCTION

Monocular depth estimation (MDE) is a fundamental geometric task that aims to recover depth from a single image. Driven by ViT backbones and large-scale training data, recent depth estimation models (Ranftl et al., 2022; Bhat et al., 2023; Bochkovskii et al., 2025; Yang et al., 2024a;b; Wang et al., 2025a;b) have achieved strong robustness and zero-shot generalization. Although their predictions are often visually plausible in 2D, they can still contain structural distortions that become more apparent when converted into point clouds. These distortions limit practical applications in 3D reconstruction, free-viewpoint rendering, robotic manipulation, and immersive content creation.

Existing MDE methods can be broadly divided into discriminative and generative paradigms. Discriminative models provide efficient feed-forward inference and typically pair a large-patch ViT encoder (Dosovitskiy et al., 2021) with a convolution-based upsampling decoder (Ranftl et al., 2021; Bochkovskii et al., 2025; Yang et al., 2024a;b; Wang et al., 2025a;b). The encoder represents the input as a low-resolution token grid and applies self-attention to capture global information. While effective for global modeling, this coarse tokenization can weaken pixel-level cues and high-frequency details. Moreover, convolutional decoding with spatially shared kernels struggles to model low-frequency regions and high-frequency details simultaneously (Knaebel et al., 2026).

Generative methods offer another direction for detailed depth estimation. Many diffusion-based methods build on pretrained latent diffusion models and benefit from strong generative priors, producing visually detailed depth predictions. However, these methods typically compress depth into a variational autoencoder (VAE) latent space, which can weaken sharp depth discontinuities and introduce structural distortions. Recent pixel-space generative methods, such as Pixel-Perfect Depth (PPD) (Xu et al., 2025) and PointDiT (Xu et al., 2026), suggest that maintaining the prediction state in pixel space is particularly beneficial for structural fidelity. They still rely on multi-step denoising, resulting in high inference cost. More broadly, recent advances in pixel-space generative modeling (Chen et al., 2026; Yu et al., 2026b; Ma et al., 2026; Wang et al., 2026) highlight the benefits of maintaining pixel-space representations for preserving fine-grained information.

Motivated by these observations, we propose PXDEPTH, a discriminative architecture for efficient, structure-preserving monocular depth estimation. PXDEPTH retains a large-patch ViT as its *Global Context Encoder* to provide global context. The *Pixel-Space Depth Predictor* is initialized directly from RGB pixels and progressively refines a persistent full-resolution feature stream, while the low-resolution ViT representation serves only as global conditioning. This design preserves high-frequency cues throughout depth prediction. Since self-attention over the full-resolution pixel grid is computationally expensive, we adopt pixel-token compaction (Yu et al., 2026b) to enable efficient feature interaction. We further use a coarse-to-fine compaction design to balance computational efficiency and geometric detail. Experiments across multiple zero-shot benchmarks show that PXDEPTH produces depth predictions with better structural fidelity than strong discriminative baselines. It also achieves faster inference than multi-step generative methods (see Figure 1).

In summary, our main contributions are as follows:

- We identify a low-resolution prediction bottleneck in mainstream discriminative MDE models, where pixel-level depth is reconstructed from spatially compressed encoder representations, making fine-grained geometry difficult to recover.
- We propose PXDEPTH, a discriminative depth model that decouples global context extraction from pixel-space prediction. A Global Context Encoder provides global context, while a Pixel-Space Depth Predictor directly takes RGB pixels as input and retains pixel-space features throughout depth prediction under global context guidance.
- Extensive zero-shot experiments show that PXDEPTH substantially improves local geometric fidelity while retaining competitive global depth accuracy and efficient single-pass inference.

2 RELATED WORK

Monocular Depth Estimation. Early monocular depth estimation relied on hand-crafted monocular cues and graphical models to infer scene layout from a single image (Hoiem et al., 2007; Saxena et al., 2007). Deep learning later turned MDE into a supervised dense prediction task,

with CNN-based models such as multi-scale prediction (Eigen et al., 2014), fully convolutional residual networks (Laina et al., 2016), ordinal regression (Fu et al., 2018), and local planar guidance (Lee et al., 2019). Later relative-depth methods focused more on cross-dataset generalization. MiDaS (Ranftl et al., 2022) showed that mixing heterogeneous datasets improves zero-shot transfer, while DPT (Ranftl et al., 2021) introduced a strong transformer-based dense prediction framework. Recent foundation models, such as Depth Anything (Yang et al., 2024a;b), further improve open-domain robustness with large-scale training. In parallel, metric-depth methods (Bhat et al., 2021; 2023; Yin et al., 2023; Hu et al., 2024; Piccinelli et al., 2024; 2025; Bochkovskii et al., 2025; Wang et al., 2025b; Song et al., 2026) aim to recover depth with absolute scale and have steadily improved metric accuracy and generalization. Despite these advances, recent discriminative models often use large-patch ViT encoders with convolutional upsampling decoders. Such early tokenization weakens high-frequency details, making detailed geometry difficult to recover during decoding.

Depth Estimation with Detailed Geometry. Recent methods increasingly focus on improving depth details and local geometry. Several discriminative methods enhance details through local refinement, high-resolution prediction, or multi-scale decoding (Bhat et al., 2022; Li et al., 2024a;b; Zhang et al., 2026). DepthPro (Bochkovskii et al., 2025) improves sharp metric depth with a high-resolution design and boundary-aware training. MoGe (Wang et al., 2025a) improves local geometry by predicting affine-invariant point maps with global and local geometry supervision, and MoGe-2 (Wang et al., 2025b) further improves details with refined training data. MDA (Bian et al., 2026) models per-pixel depth ambiguity with a mixture-density representation to produce flying-point-free point maps. InfiniDepth (Yu et al., 2026a) represents depth as a neural implicit field and uses a local implicit decoder, enabling arbitrary-resolution querying and sharper depth details. These methods can produce sharp depth maps, but structural distortions may still appear after back-projection into point clouds, limiting applications that rely on accurate 3D geometry. Concurrent with our work, SurGe (Knaebel et al., 2026) attributes local surface distortions to the difficulty of fixed convolutional kernels in reconstructing high-frequency, high-amplitude signals and introduces a Neighborhood Attention Decoder for adaptive local feature mixing. MoGe-3 (Kong et al., 2026) recovers fine geometric structures through Self-Guided Sparse Volumetric Refinement, which repeatedly re-voxelizes and refines a coarse point map in sparse 3D space but requires multiple refinement steps.

Generative methods (Liu et al., 2026; Ke et al., 2024; Pavlovic et al., 2026; He et al., 2025b;a; Xu et al., 2025; 2026) provide another direction for detailed depth estimation. Latent diffusion methods such as Marigold (Ke et al., 2024) and Marigold V2 (Pavlovic et al., 2026) introduce diffusion priors for monocular depth by adapting image diffusion models. Lotus-2 (He et al., 2025b;a) further adapts generative priors to dense geometry with a deterministic two-stage framework. These latent-space generative models may suffer from compression-induced structural distortion. On the other hand, Pixel-Perfect Depth (Xu et al., 2025) performs pixel-space diffusion and uses semantic guidance with a cascade DiT design, improving structural details, while PointDiT (Xu et al., 2026) adopts a minimalist pixel-space diffusion transformer to the same end. Both still require multi-step denoising, resulting in high inference latency. In contrast, our method retains the discriminative formulation while learning pixel-space features conditioned on global context from the Global Context Encoder, improving depth details and point-cloud structures with efficient inference (see Figure 2).

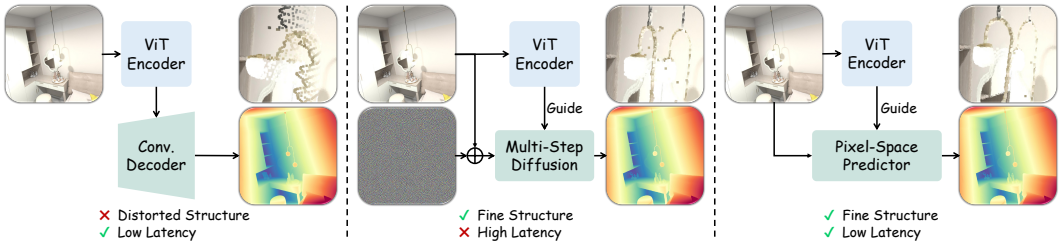
3 Motivation

Overview. Given an input image $I \in \mathbb{R}^{H\times W\times 3}$, the MDE task is to predict a pixel-aligned depth map $\hat{D} \in \mathbb{R}^{H\times W}$.

$$\hat{D} = f(I). \tag{1}$$

Most discriminative MDE methods pair a large-patch ViT encoder (Dosovitskiy et al., 2021; Darcet et al., 2024; Siméoni et al., 2025) with a convolutional decoder (Yang et al., 2024a;b; Wang et al., 2025a;b; Ranftl et al., 2021). Specifically, a large-patch ViT encoder $\mathcal{E}_{\text{vit}}$ first tokenizes the input image with non-overlapping patches of size P, and then performs self-attention to produce low-resolution features F_{lr}. A convolution-based decoder $\mathcal{D}_{\text{conv}}$ then upsamples these features to recover the pixel-aligned depth map.

$$\begin{aligned} F_{\text{lr}} &= \mathcal{E}_{\text{vit}}(I), \qquad F_{\text{lr}} \in \mathbb{R}^{\frac{H}{P}\times\frac{W}{P}\times d_{\text{lr}}}, \\ \hat{D} &= \mathcal{D}_{\text{conv}}(F_{\text{lr}}), \quad \hat{D} \in \mathbb{R}^{H\times W}. \end{aligned} \tag{2}$$



(a) Discriminative Architecture **(b)** PPD **(c)** Ours

Figure 2. **Architecture comparison of monocular depth estimators.** Conventional discriminative models decode depth from low-resolution ViT features using a convolutional decoder (Ranftl et al., 2021), PPD (Xu et al., 2025) performs iterative pixel-space denoising, and PXDEPTH predicts depth with a Pixel-Space Depth Predictor conditioned on global context in a single feed-forward pass.

This architecture is effective in capturing global structure through self-attention, but high-frequency cues are inevitably weakened during large-patch tokenization. Moreover, the decoder (Ranftl et al., 2021) mainly relies on convolutional upsampling with spatially shared kernels, making it difficult to handle low-frequency regions and high-frequency details simultaneously (Knaebel et al., 2026). As a result, the predicted depth may appear sharp, while still containing structural distortions.

Generative models offer another direction for detailed depth estimation. Many diffusion-based methods build on pretrained latent diffusion models (Ke et al., 2024; Pavlovic et al., 2026) and benefit from strong generative priors. However, they typically compress depth maps into a variational autoencoder (VAE) latent space, which can weaken sharp depth discontinuities and structural fidelity.

Main Idea. PPD (Xu et al., 2025) and PointDiT (Xu et al., 2026) avoid VAE-based latent compression by performing diffusion directly in pixel space, improving structural fidelity. However, these methods still require multi-step denoising, resulting in high inference cost. Motivated by these observations, we retain the efficient discriminative formulation while learning pixel-space features for depth prediction under global context guidance. Formally, our model is written as follows.

$$\begin{aligned} F_{\text{ctx}} &= \mathcal{E}_{\text{ctx}}(I), & F_{\text{ctx}} &\in \mathbb{R}^{\frac{H}{P}\times\frac{W}{P}\times d_{\text{ctx}}}, \\ \hat{D} &= \mathcal{P}_{\text{pix}}(I, F_{\text{ctx}}), & \hat{D} &\in \mathbb{R}^{H\times W}. \end{aligned} \tag{3}$$

Here, $\mathcal{E}_{\text{ctx}}$ and $\mathcal{P}_{\text{pix}}$ denote Global Context Encoder and Pixel-Space Depth Predictor, respectively. Unlike conventional discriminative models that reconstruct depth from low-resolution encoder features, our formulation separates global context modeling from pixel-space feature learning, allowing $\mathcal{P}_{\text{pix}}$ to preserve fine-grained cues under the guidance of F_{ctx} without sacrificing global consistency.

4 METHOD

As illustrated in Figure 3, PXDEPTH consists of a Global Context Encoder for extracting global context and a Pixel-Space Depth Predictor that learns pixel-space features. The Pixel-Space Depth Predictor maintains full-resolution pixel features, progressively refines them through Context-Modulated Pixel Transformer (CM-PiT) blocks conditioned on the global context, and finally maps the refined features to a depth map with a lightweight depth head.

4.1 GLOBAL CONTEXT ENCODER

The Global Context Encoder $\mathcal{E}_{\text{ctx}}$ is implemented with a large-patch ViT to provide global context for the Pixel-Space Depth Predictor. Given the input image I, the encoder tokenizes it into non-overlapping patches of size P, producing a low-resolution token grid of size $\frac{H}{P} \times \frac{W}{P}$. Self-attention aggregates global information over this grid to produce d_{ctx}-dimensional context features $F_{\text{ctx}} = \mathcal{E}_{\text{ctx}}(I)$. The Global Context Encoder can inherit strong priors from pretrained vision foundation models, such as DINOv3 (Siméoni et al., 2025) and MoGe-2 (Wang et al., 2025b).

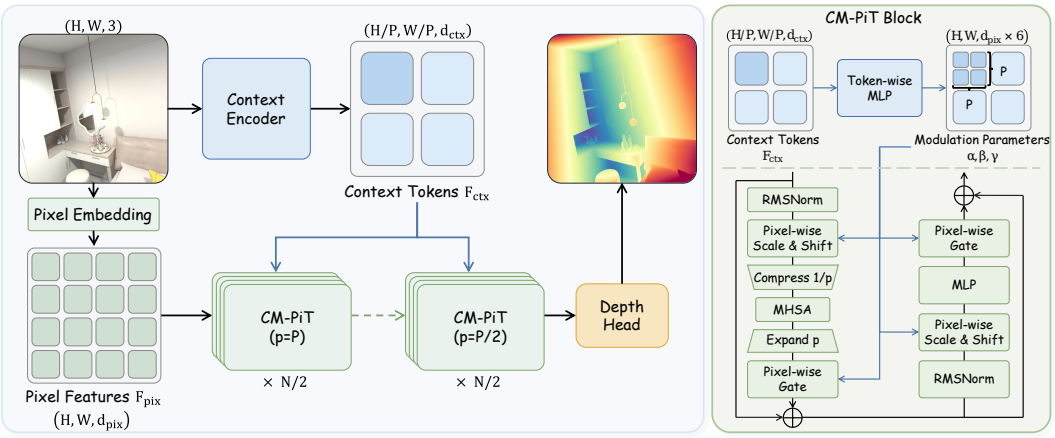


Figure 3. **Overview of the proposed PXDEPTH.** PXDEPTH combines a Global Context Encoder with a Pixel-Space Depth Predictor built from CM-PiT blocks. Context features F_{ctx} generate pixel-wise modulation parameters for CM-PiT blocks. The Pixel-Space Depth Predictor maintains pixel-space features and changes the compaction size from P to $P/2$ for coarse-to-fine refinement.

4.2 PIXEL-SPACE DEPTH PREDICTOR

Pixel Embedding. Given the input image I, the pixel embedding layer ϕ_{in} maps each RGB pixel to a d_{pix}-dimensional feature.

$$F^0_{\text{pix}} = \phi_{\text{in}}(I), \qquad F^0_{\text{pix}} \in \mathbb{R}^{H\times W\times d_{\text{pix}}}, \tag{4}$$

where ϕ_{in} is implemented by a 1×1 convolution and F^0_{pix} denotes initial pixel-space features. Unlike conventional decoders that reconstruct depth from low-resolution features, this embedding initializes a pixel-space feature stream that is progressively refined throughout the Pixel-Space Depth Predictor.

Pixel-Space Feature Learning. We process the initial pixel features F^0_{pix} with a sequence of N CM-PiT blocks guided by the context feature F_{ctx}.

$$F^{i+1}_{\text{pix}} = \text{CM-PiT}^i(F^i_{\text{pix}}, F_{\text{ctx}}), \qquad i = 0, \ldots, N-1. \tag{5}$$

Here, F^i_{pix} denotes the pixel features before the i-th block.

Depth Head. A linear layer maps the refined pixel features to the depth map.

$$\hat{D} = \phi_{\text{out}}(F^N_{\text{pix}}), \qquad \hat{D} \in \mathbb{R}^{H\times W}, \tag{6}$$

where ϕ_{out} is implemented by a 1×1 convolution.

4.3 CONTEXT-MODULATED PIXEL TRANSFORMER (CM-PIT) BLOCK

Each CM-PiT block processes pixel-space features by conditioning its attention and feed-forward network on F_{ctx}. We first compare alternative guidance strategies, then present the adopted Context-Guided Adaptive Normalization.

Alternative Guidance Strategies. Figure 4 compares three strategies for incorporating F_{ctx} into pixel-space depth prediction. Direct addition merges F_{ctx} with compacted pixel tokens before self-attention, while cross-attention introduces an additional attention layer to learn the correspondence between the two representations. In contrast, Context-Guided Adaptive Normalization, originally introduced in PixelDiT (Yu et al., 2026b), projects F_{ctx} into pixel-wise scale, shift, and residual gates, allowing global context to condition pixel-space features by controlling the normalization and residual updates in each CM-PiT block. We ultimately adopt this design as it provides the best balance between performance and efficiency. Unlike cross-attention, which must learn the correspondence between F_{ctx} and F_{pix} implicitly, its pixel-wise modulation preserves this correspondence explicitly, thereby improving inference efficiency and potentially facilitating pixel-space feature learning.

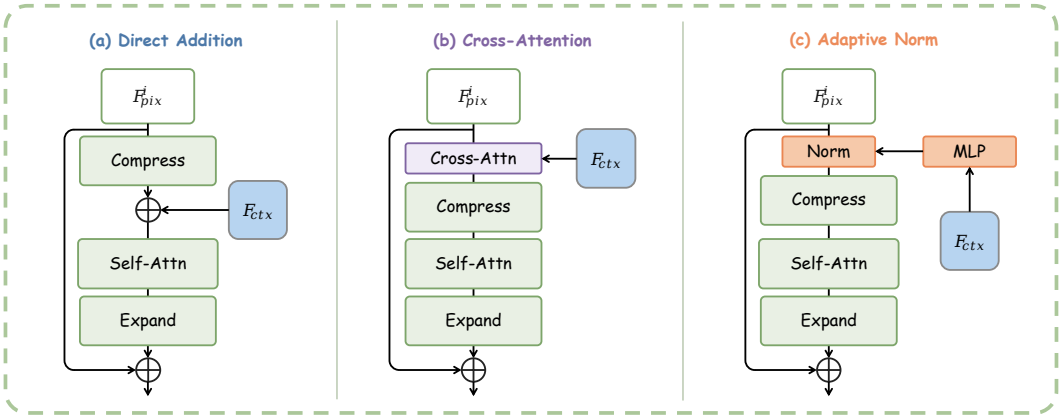


Figure 4. **Alternative guidance strategies.** Feature flow within the candidate CM-PiT designs.

Context-Guided Adaptive Normalization. As shown in Figure 3, F_{ctx} generates adaptive normalization parameters for the CM-PiT block. Learned projections $\alpha(\cdot)$, $\beta(\cdot)$, and $\gamma(\cdot)$ each map a context token to $P^2 d_{\text{pix}}$ values, which are reshaped into $P \times P$ pixel-aligned maps to provide the residual gate, shift, and scale for the attention layer, respectively. We formulate it as

$$\bar{F}^i_{\text{pix}} = F^i_{\text{pix}} + \alpha(F_{\text{ctx}}) \odot \mathcal{A}\big(\text{RMSNorm}(F^i_{\text{pix}}) \odot \gamma(F_{\text{ctx}}) + \beta(F_{\text{ctx}})\big), \tag{7}$$

where $\bar{F}^i_{\text{pix}}$ denotes the output pixel features. To avoid applying self-attention directly over the full-resolution $H \times W$ pixel grid, we adopt pixel token compaction (Yu et al., 2026b). The attention operator $\mathcal{A}(\cdot)$ combines token compaction, self-attention, and expansion, and is expressed as $\mathcal{A}(\cdot) = \mathcal{U}_p(\text{Self-Attn}(\mathcal{C}_p(\cdot), \text{RoPE}))$. Here, $\mathcal{C}_p$ compacts the p^2 features in each $p \times p$ local region into one token, and $\mathcal{U}_p$ expands each token back to its original pixel region. The attention sequence length is reduced from HW to $(H/p)(W/p)$, giving a p^2-fold reduction, while the residual path maintains pixel-space features and preserves high-frequency cues. Finally, $\bar{F}^i_{\text{pix}}$ is passed through a feed-forward network with the same modulation form to produce F^{i+1}_{pix} for the next CM-PiT block.

Coarse-to-Fine Pixel Compaction. To progressively refine depth from coarse structures to fine details, we adopt the coarse-to-fine pixel compaction design illustrated in Figure 3. The early $N/2$ CM-PiT blocks use a compaction size of $p = P$, matching the patch size of the Global Context Encoder. This reduces the number of tokens and enables efficient coarse-structure modeling. The later $N/2$ blocks reduce the compaction size to $p = P/2$, increasing the token density for fine-detail refinement. This coarse-to-fine design balances computational efficiency with geometric fidelity.

5 Experiments

5.1 Implementation Details

Training Datasets. We train on multiple synthetic RGB-D datasets, including Hypersim (Roberts et al., 2021), MVS-Synth (Huang et al., 2018), TartanAir (Wang et al., 2020), TartanGround (Patel et al., 2025), UnrealStereo4K (Tosi et al., 2021), GTA-SfM (Wang & Shen, 2020), Structured3D (Zheng et al., 2020), UrbanSyn (Gómez et al., 2025), ParallelDomain-4D (Van Hoorick et al., 2024), VKITTI2 (Cabon et al., 2020), SceneNetRGBD (McCormac et al., 2017), RobbySim (Tan et al., 2026), and Synscapes (Wrenninge & Unger, 2018).

Model Configuration. We use ViT-L/14 as the Global Context Encoder, with $P = 14$ and $d_{\text{ctx}} = 1024$ and model weights initialized from MoGe-2. The Pixel-Space Depth Predictor contains $N = 8$ CM-PiT blocks with a pixel feature dimension of $d_{\text{pix}} = 16$. In our coarse-to-fine pixel compaction design, the first 4 blocks use a compaction size of $p = 14$, while the remaining 4 blocks use $p = 7$. We additionally introduce a mask head for valid mask prediction (Wang et al., 2025a;b).

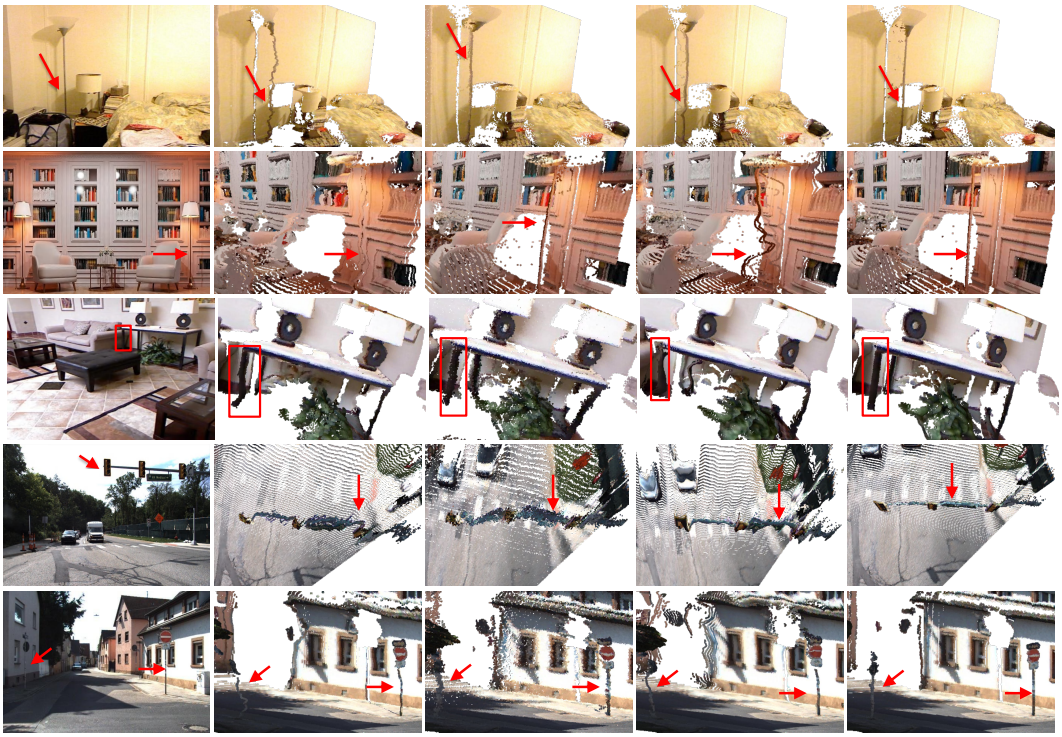


Figure 5. **Qualitative comparison on diverse scenes**. PXDEPTH better preserves local structures.

Training Configuration. We train the model in two stages, with 500K iterations at 518×518, followed by 200K iterations at varying resolutions with the image area kept equivalent to 1022×770. Additional optimization and loss details are provided in Appendix C.

5.2 Evaluation Setup and Metrics

Benchmarks and Baselines. We conduct zero-shot evaluation on the MoGe Benchmark (Wang et al., 2025a;b) and the MDA Benchmark (Bian et al., 2026). The MoGe Benchmark includes NYUv2 (Silberman et al., 2012), KITTI (Geiger et al., 2012), ETH3D (Schöps et al., 2017), iBims-1 (Koch et al., 2018), Sintel (Butler et al., 2012), DDAD (Guizilini et al., 2020), DIODE (Vasiljevic et al., 2019), and HAMMER (Jung et al., 2023). The MDA Benchmark consists of NRGBD (Azinović et al., 2022), 7Scenes (Shotton et al., 2013), and HiRoom (Lin et al., 2025). We compare our method with seven recent approaches, including Depth Anything V2 (DA V2) (Yang et al., 2024b), DepthPro (Bochkovskii et al., 2025), InfiniDepth (Yu et al., 2026a), MoGe-2 (Wang et al., 2025b), PPD (Xu et al., 2025), PointDiT (Xu et al., 2026), and MDA (Bian et al., 2026). Moreover, we compare with the contemporary structure-aware depth prediction methods MoGe-3 (Kong et al., 2026) and SurGe (Knaebel et al., 2026) in Appendix B.

Evaluation Metrics. Following the standard affine-invariant depth evaluation protocol, we apply scale-and-shift alignment and report the widely used relative error (Rel↓) and accuracy (δ_1 ↑). Alignment details are provided in Appendix E.3. Both metrics are reported as percentages.

Boundary metrics assess the structural accuracy and sharpness of depth maps. Following PPD (Xu et al., 2025) and MDA (Bian et al., 2026), we extract boundaries from GT depth using a Canny detector and remove edges adjacent to invalid regions. Predicted and GT depths at the retained boundary pixels are back-projected into 3D, and the predicted boundary point cloud is aligned to the GT using the same point-to-point iterative closest point (ICP) registration as MDA. We report accuracy (Acc↓) and Chamfer distance (CD↓) in millimeters. See Appendix E.4 for more details.

Table 1. **Evaluation on the MoGe Benchmark.** Zero-shot depth estimation across eight datasets. * indicates direct resizing to the required input resolution.

Category	Method	NYUv2		KITTI		ETH3D		iBims-1		Sintel		DDAD		DIODE		HAMMER		Mean	
		Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑
Global Prediction	DA V2	4.4	98.0	6.9	95.7	4.6	97.9	3.5	98.3	21.2	72.8	13.6	86.0	5.0	96.4	4.9	99.1	8.0	93.0
	DepthPro*	3.8	98.0	5.7	96.1	5.1	96.2	3.0	**98.7**	16.0	79.9	12.7	83.5	4.2	96.7	3.3	**99.4**	6.7	93.6
	MoGe-2	**3.1**	**98.4**	**4.2**	**97.9**	**2.9**	**99.0**	**2.4**	**98.7**	**14.0**	**81.4**	**8.5**	**92.1**	**2.9**	**98.0**	**3.0**	99.3	**5.1**	**95.6**
Structure Aware	PPD	4.1	97.8	6.9	94.6	4.9	97.1	3.4	98.3	18.5	77.1	13.8	81.3	4.9	95.3	3.1	99.3	7.5	92.6
	InfiniDepth	4.6	97.8	5.6	96.6	4.2	98.1	3.5	98.4	19.3	79.5	12.6	86.5	4.6	96.9	3.0	99.2	7.2	94.1
	PointDiT-L*	3.8	**98.0**	6.2	95.6	3.5	97.6	2.7	98.5	14.7	82.0	12.9	84.9	**3.5**	**97.3**	6.0	97.4	6.7	93.9
	MDA	3.6	97.6	7.0	94.8	5.4	94.5	3.1	98.0	17.4	77.7	21.8	67.9	4.7	95.2	2.6	98.7	8.2	90.5
	Ours	**3.5**	**98.0**	**5.2**	**96.8**	**2.9**	**98.6**	**2.7**	**98.5**	**14.3**	**82.3**	**11.5**	**88.5**	3.6	96.7	**2.4**	**99.4**	**5.8**	**94.8**

Table 2. **Quantitative comparison on the MDA Benchmark.** Acc and CD are reported in millimeters. * indicates direct resizing to the required input resolution.

Category	Method	(a) Global Accuracy								(b) Boundary Fidelity								Time↓ (ms)
		NRGBD		7Scenes		HiRoom		Mean		NRGBD		7Scenes		HiRoom		Mean		
		Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Acc↓	CD↓	Acc↓	CD↓	Acc↓	CD↓	Acc↓	CD↓	
Global Prediction	DA V2	2.6	98.9	7.7	92.6	3.7	98.1	4.7	96.5	**118.8**	**112.1**	97.5	115.1	115.0	111.1	110.5	112.8	60.5
	DepthPro*	**2.2**	**99.0**	7.0	92.8	2.5	**99.2**	3.9	**97.0**	140.2	131.5	86.2	97.3	101.3	90.4	109.2	106.4	223.7
	MoGe-2	**2.2**	98.6	**6.4**	**93.0**	**1.9**	99.1	**3.5**	96.9	136.1	128.4	**80.3**	**90.8**	**88.0**	**81.6**	**101.5**	**100.3**	**20.5**
Structure Aware	PPD	2.5	98.7	7.1	92.4	3.6	98.3	4.4	96.5	74.9	85.9	92.1	103.6	84.0	91.5	83.6	93.7	206.9
	InfiniDepth	2.9	98.7	7.6	**92.7**	3.6	98.5	4.7	96.7	85.9	88.5	100.1	117.7	92.1	90.5	92.7	98.9	61.6
	PointDiT-L*	2.4	98.6	7.7	92.4	2.1	98.9	4.1	96.7	102.4	90.8	91.4	97.7	66.3	62.5	86.7	83.7	104.6
	MDA	**2.0**	98.4	**6.8**	**92.7**	2.0	99.0	**3.6**	96.7	**50.9**	67.6	**79.5**	**91.3**	**49.1**	59.3	**59.8**	72.8	100.1
	Ours	**2.0**	**98.8**	6.9	**92.7**	**1.8**	**99.4**	**3.6**	**97.0**	60.2	**64.7**	83.0	92.0	56.1	**54.9**	66.5	**70.6**	**48.9**

5.3 Evaluation Results

Global Depth Accuracy. As shown in Table 1 & Table 2 (a), our method achieves better global depth accuracy than structure-aware methods, while remaining comparable to state-of-the-art methods that focus on global depth prediction. The remaining performance gap with MoGe-2 (Wang et al., 2025b) may be attributed to its substantially larger and more diverse training data.

Boundary Fidelity. As shown in Table 2 (b), our method achieves the best mean CD and the second-best mean Acc. These metrics assess geometric deviations at depth boundaries, including errors associated with flying points and structural distortions. DA V2, DepthPro, InfiniDepth, and MoGe-2 recover depth from low-resolution features through upsampling, which often introduces flying points and distorted structures. PPD preserves relatively sharp boundaries, but exhibits noise near them. MDA achieves comparable performance because it produces fewer flying points near boundaries but exhibits more distorted structure. In contrast, our method preserves accurate boundary geometry, yielding strong overall boundary fidelity, as shown in Figure 5.

Inference Efficiency. Table 2 compares per-image inference time. These runtime measurements are conducted at a resolution of 518×518 on a single NVIDIA RTX 5880 Ada Generation GPU. Our method is slower only than MoGe-2 and is substantially faster than PointDiT-L and PPD. This efficiency advantage over generative methods follows from predicting depth in a single feed-forward pass rather than through multi-step denoising.

5.4 Ablation and Analysis

To justify the effectiveness of our method, we analyze our design using Hypersim (Roberts et al., 2021) and UrbanSyn (Gómez et al., 2025) for training and HiRoom (Lin et al., 2025) for evaluation.

Effect of Global Context Encoder. Table 3 (a) shows that removing the Global Context Encoder substantially degrades both global and boundary metrics. MoGe-2 initialization further improves these metrics over DINOv3, suggesting that geometry-aware initialization provides a stronger prior for depth prediction. Figure 6 (a & b) shows that removing the Global Context Encoder causes the model to struggle with capturing global structures.

Table 3. **Quantitative ablation on model components.** Acc and CD are reported in millimeters.

Ablation	Variant	Rel↓	δ_1 ↑	Acc↓	CD↓	Time (ms)↓
(a) Global Context Encoder	w/o Global Context Encoder	17.87	72.87	261.0	385.6	**28.7**
	DINOv3 Initialization	3.02	98.73	74.8	76.9	50.3
	MoGe-2 Initialization	**2.74**	**99.01**	**70.3**	**72.5**	48.9
(b) Pixel-Space Predictor	w/ conv decoder	2.73	99.00	84.0	77.1	**20.5**
	w/o coarse-to-fine	**2.72**	98.98	74.9	76.8	38.7
	Ours	2.74	**99.01**	**70.3**	**72.5**	48.9
(c) Guidance Strategy	Addition	3.19	98.91	79.3	80.2	**46.7**
	Cross-Attn	2.98	98.88	79.7	80.9	56.6
	Adaptive Norm	**2.74**	**99.01**	**70.3**	**72.5**	48.9

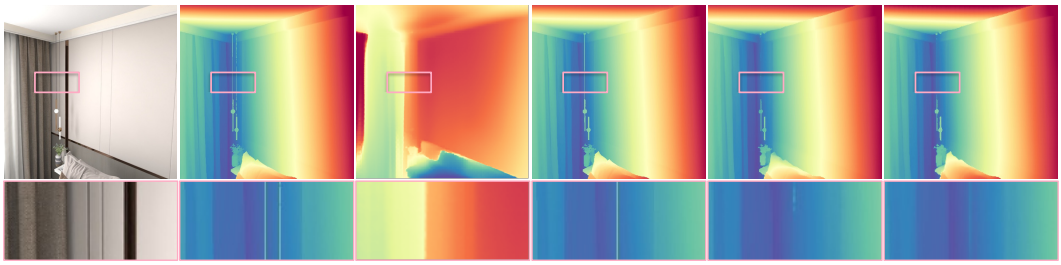


Image | (a) Ours | (b) w/o Context Encoder | (c) w/o Coarse-to-Fine | (d) w/ Addition Guidance | (e) w/ Cross-Attn Guidance

Figure 6. **Qualitative ablation on model components.** The panels show (a) our full model, (b) without the Global Context Encoder, (c) without coarse-to-fine pixel compaction, (d) addition guidance, and (e) cross-attention guidance.

Effect of Pixel-Space Predictor. As shown in Table 3 (b), we compare Pixel-Space Depth Predictor with a convolutional decoder (Wang et al., 2025b) and evaluate the effect of coarse-to-fine pixel compaction in Pixel-Space Depth Predictor. Both pixel-space prediction and coarse-to-fine compaction improve boundary quality while maintaining comparable global depth accuracy. The comparison in Figure 6 (a & c) further shows that the coarse-to-fine design preserves the local structure. These results indicate that reducing the compaction size in later blocks helps refine local geometric details.

Effect of Context Guidance Strategy. The guidance-strategy ablation compares the three designs introduced in Figure 4. As shown in Table 3 (c) and Figure 6 (a, d & e), the adopted Adaptive Normalization achieves better global depth accuracy and boundary quality than direct addition and cross-attention, with only a modest increase in inference time over direct addition.

6 Conclusion

In this paper, we revisit the challenge of preserving fine-grained structures in monocular depth estimation. We identify a low-resolution prediction bottleneck in discriminative models, where pixel-level depth is reconstructed from spatially compressed encoder representations. To address this, we present PXDepth, a single-pass discriminative model that decouples global context extraction from pixel-space prediction. A Global Context Encoder provides global context, while a Pixel-Space Depth Predictor directly takes RGB pixels as input and retains pixel-space features throughout depth prediction under global context guidance. Extensive zero-shot experiments demonstrate that PXDepth improves local geometric fidelity while retaining competitive global depth accuracy and substantially lower latency than multi-step generative methods.

Limitations and future work. One limitation of our approach is that depth annotations for transparent objects can be ambiguous or unreliable, which may lead to inaccurate predictions on transparent and reflective surfaces. Another limitation is that our relative-depth formulation cannot recover metric scale without an external reference. In future work, we aim to improve supervision for challenging materials and incorporate metric cues to enable scale-aware depth estimation.

AI USE STATEMENT

We used generative AI tools, including ChatGPT and Codex, to assist with manuscript preparation, grammar and language polishing, LaTeX formatting, and code organization and formatting. All AI-assisted text and code were reviewed by the authors, who take full responsibility for the final content of this work.

ETHICS STATEMENT

This work studies monocular depth estimation using existing synthetic RGB-D training datasets and publicly available evaluation benchmarks, without collecting new data or conducting studies involving human participants. We do not identify specific ethical concerns arising from this study. As depth estimation errors may affect downstream decisions, the method should be independently validated before deployment in safety-critical applications.

REPRODUCIBILITY STATEMENT

The method and experimental setup are described in the main paper, with additional training, architecture, and evaluation details in Appendices C, D, and E, respectively. To support reproducibility, we will publicly release our training and evaluation code, model weights, and data preprocessing code.

REFERENCES

Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, 2022. 7, 20

Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *CVPR*, 2021. 3

Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Localbins: Improving depth estimation by learning local distributions. In *ECCV*, 2022. 3

Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 2, 3

Siyuan Bian, Congrong Xu, and Jun Gao. Modeling depth ambiguity: A mixture-density representation for flying-point-free depth estimation. *arXiv preprint arXiv:2606.02552*, 2026. 1, 3, 7, 15, 20

Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. In *ICLR*, 2025. 2, 3, 7, 18, 19, 20

D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 7, 20

Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 6, 16

Zhennan Chen, Junwei Zhu, Xu Chen, Jiangning Zhang, Xiaobin Hu, Hanzhen Zhao, Chengjie Wang, Jian Yang, and Ying Tai. Dip: Taming diffusion models in pixel space. In *CVPR*, 2026. 2

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *ICLR*, 2024. 3

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 2, 3

David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, 2014. 3

Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018. 3

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 7, 20

Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *CVPR*, 2020. 7, 20

Jose L. Gómez, Manuel Silva, Antonio Seoane, Agnés Borràs, Mario Noriega, German Ros, Jose A. Iglesias-Guitian, and Antonio M. López. All for one, and one for all: Urbansyn dataset, the third musketeer of synthetic driving scenes. *Neurocomputing*, 2025. 6, 8, 16

Jing He, Haodong Li, Mingzhi Sheng, and Ying-Cong Chen. Lotus-2: Advancing geometric dense prediction with powerful image generative model. *arXiv preprint arXiv:2512.01030*, 2025a. 3

Jing He, Haodong Li, Wei Yin, Yixun Liang, Leheng Li, Kaiqiang Zhou, Hongbo Zhang, Bingbing Liu, and Ying-Cong Chen. Lotus: Diffusion-based visual foundation model for high-quality dense prediction. In *ICLR*, 2025b. 3

Derek Hoiem, Alexei A Efros, and Martial Hebert. Recovering surface layout from an image. *IJCV*, 2007. 2

Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *TPAMI*, 2024. 3

Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *CVPR*, 2018. 6, 16

HyunJun Jung, Patrick Ruhkamp, Guangyao Zhai, Nikolas Brasch, Yitong Li, Yannick Verdie, Jifei Song, Yiren Zhou, Anil Armagan, Slobodan Ilic, Aleš Leonardis, Nassir Navab, and Benjamin Busam. On the importance of accurate geometry data for dense 3d vision tasks. In *CVPR*, 2023. 7, 20, 22

Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, 2024. 3, 4

Karim Knaebel, Gonzalo Martin Garcia, Christian Schmidt, Ilya Fradlin, Lucas Nunes, Daan de Geus, and Bastian Leibe. Surge: Improved surface geometry in point maps. *arXiv preprint arXiv:2605.31577*, 2026. 2, 3, 4, 7, 14, 15, 16, 20

Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Korner. Evaluation of cnn-based single-image depth estimation methods. In *ECCVW*, 2018. 7, 20, 22

Lingyu Kong, Ruicheng Li, Ruicheng Wang, Sicheng Xu, Chengtang Yao, Jianfeng Xiang, and Jiaolong Yang. MoGe-3: Fine-detail monocular geometry estimation with self-guided sparse volumetric refinement. *arXiv preprint arXiv:2607.17967*, 2026. 3, 7, 14, 15, 20

Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 3

Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019. 3

Zhenyu Li, Shariq Farooq Bhat, and Peter Wonka. Patchfusion: An end-to-end tile-based framework for high-resolution monocular metric depth estimation. In *CVPR*, 2024a. 3

Zhenyu Li, Shariq Farooq Bhat, and Peter Wonka. Patchrefiner: Leveraging synthetic data for real-domain high-resolution monocular metric depth estimation. In *ECCV*, 2024b. 3

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. 7, 8, 20

Bingde Liu, Wu Ran, Jinglei Zhang, Huanhuan Yuan, and Chao Ma. Efficient and high-quality depth estimation via pixel-space diffusion with linear attention. In *ECCV*, 2026. 3

Zehong Ma, Longhui Wei, Shuai Wang, Shiliang Zhang, and Qi Tian. Deco: Frequency-decoupled pixel diffusion for end-to-end image generation. In *CVPR*, 2026. 2

John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J Davison. Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? In *ICCV*, 2017. 6, 16

Manthan Patel, Fan Yang, Yuheng Qiu, Cesar Cadena, Sebastian Scherer, Marco Hutter, and Wenshan Wang. Tartanground: A large-scale dataset for ground robot perception and navigation. In *IROS*, 2025. 6, 16

Igor Pavlovic, Thiemo Wandel, Anton Obukhov, Luca Bartolomei, Andrey Davydov, Fabio Tosi, Matteo Poggi, Sabine Süsstrunk, and Dengxin Dai. Marigold v2: Revisiting diffusion transformers for monocular depth estimation. *TOG*, 2026. 3, 4

Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *CVPR*, 2024. 3

Luigi Piccinelli, Christos Sakaridis, Yung-Hsu Yang, Mattia Segu, Siyuan Li, Wim Abbeloos, and Luc Van Gool. Unidepthv2: Universal monocular metric depth estimation made simpler. *TPAMI*, 2025. 3

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, 2021. 2, 3, 4

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *TPAMI*, 2022. 2, 3, 18

Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *ICCV*, 2021. 6, 8, 16

Ashutosh Saxena, Min Sun, and Andrew Y Ng. Learning 3-d scene structure from a single still image. In *ICCVW*, 2007. 2

Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, 2017. 7, 20

Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, 2013. 7, 20

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 7, 20

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025. 3, 4

Zhongqiang Song, Guanying Chen, Yuqi Zhang, Yin Zou, Chuanyu Fu, Zhiyuan Yuan, Chuan Huang, Shuguang Cui, and Xiaochun Cao. Aerialmetric: Benchmarking and adapting uav monocular metric depth estimation in the real world. In *ECCV*, 2026. 3

Bin Tan, Changjiang Sun, Xiage Qin, Hanat Adai, Zelin Fu, Tianxiang Zhou, Han Zhang, Yinghao Xu, Xing Zhu, Yujun Shen, and Nan Xue. Masked depth modeling for spatial perception. In *ECCV*, 2026. 6, 16

Fabio Tosi, Yiyi Liao, Carolin Schmitt, and Andreas Geiger. Smd-nets: Stereo mixture density networks. In *CVPR*, 2021. 6, 16

Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *ECCV*, 2024. 6, 16

Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z Dai, Andrea F Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R Walter, and Gregory Shakhnarovich. Diode: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019. 7, 20

Kaixuan Wang and Shaojie Shen. Flow-motion and depth network for monocular stereo and beyond. *RA-L*, 2020. 6, 16

Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *CVPR*, 2025a. 2, 3, 6, 7

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. In *NeurIPS*, 2025b. 2, 3, 4, 6, 7, 8, 9, 14, 20

Shuai Wang, Ziteng Gao, Chenhui Zhu, Weilin Huang, and Limin Wang. Pixnerd: Pixel neural field diffusion. In *ICLR*, 2026. 2

Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *IROS*, 2020. 6, 16

Magnus Wrenninge and Jonas Unger. Synscapes: A photorealistic synthetic dataset for street scene parsing. *arXiv preprint arXiv:1810.08705*, 2018. 6, 16

Gangwei Xu, Haotong Lin, Hongcheng Luo, Xianqi Wang, Jingfeng Yao, Lianghui Zhu, Yuechuan Pu, Cheng Chi, Haiyang Sun, Bing Wang, Guang Chen, Hangjun Ye, Sida Peng, and Xin Yang. Pixel-perfect depth with semantics-prompted diffusion transformers. In *NeurIPS*, 2025. 1, 2, 3, 4, 7, 14, 20

Haofei Xu, Rundi Wu, Philipp Henzler, Nikolai Kalischek, Michael Oechsle, Fabian Manhardt, Marc Pollefeys, Andreas Geiger, Federico Tombari, and Michael Niemeyer. Pointdit: Pixel-space diffusion for monocular geometry estimation. In *ICML*, 2026. 2, 3, 4, 7, 20

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024a. 2, 3

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. In *NeurIPS*, 2024b. 2, 3, 7, 18, 19, 20

Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *ICCV*, 2023. 3

Hao Yu, Haotong Lin, Jiawei Wang, Jiaxin Li, Yida Wang, Xueyang Zhang, Yue Wang, Xiaowei Zhou, Ruizhen Hu, and Sida Peng. Infinidepth: Arbitrary-resolution and fine-grained depth estimation with neural implicit fields. In *CVPR*, 2026a. 1, 3, 7, 14, 19, 20, 22

Yongsheng Yu, Wei Xiong, Weili Nie, Yichen Sheng, Shiqiu Liu, and Jiebo Luo. Pixeldit: Pixel diffusion transformers for image generation. In *CVPR*, 2026b. 2, 5, 6

Zaiwei Zhang, Marc Mapeke, Wei Ye, Rakesh Ranjan, and JQ Huang. Hyden: A hybrid dual-path encoder for monocular geometry of high-resolution images. In *ICLR*, 2026. 3

Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *ECCV*, 2020. 6, 16

APPENDIX

A Supplementary Video **14**

B More comparisons with recent structure-aware depth prediction methods **15**

B.1 Comparison with MoGe-3 15

B.2 Comparison with SurGe 16

C Additional Implementation Details **16**

C.1 Training Data and Configuration 16

C.2 Training Objective 18

D More Architecture Details **18**

D.1 Model Configuration 18

D.2 Pixel Token Compaction 19

D.3 Alternative Guidance Strategies 19

D.4 Context-Guided Adaptive Normalization 19

E Evaluation Protocol Details **19**

E.1 Baseline Model Configurations 19

E.2 Evaluation Datasets and Resolution Handling 20

E.3 Global Depth Accuracy Evaluation Protocol 20

E.4 Boundary Evaluation Protocol 20

F Additional Experimental Results **21**

F.1 Ablation on the Number of CM-PiT Blocks 21

F.2 Center-Crop Comparison with PointDiT 21

F.3 Global Depth Evaluation on Synth4K 22

F.4 Boundary Evaluation on iBims-1 and HAMMER 22

F.5 Additional Qualitative Results 22

A SUPPLEMENTARY VIDEO

We provide a supplementary video of reconstructed point clouds. It compares PXDEPTH with MoGe-2 (Wang et al., 2025b), InfiniDepth (Yu et al., 2026a), and PPD (Xu et al., 2025). The video also includes comparisons with MoGe-3 (Kong et al., 2026) using one, two, and three refinement steps, as well as with SurGe (Knaebel et al., 2026). These visualizations complement the comparisons in the paper and allow closer inspection of local structures and boundary details.

B MORE COMPARISONS WITH RECENT STRUCTURE-AWARE DEPTH PREDICTION METHODS

Recent works such as MoGe-3 (Kong et al., 2026) and SurGe (Knaebel et al., 2026) specifically study structural distortions. MoGe-3 attributes these distortions to decoding 3D geometry in a 2D parameterization and addresses them through Self-Guided Sparse Volumetric Refinement in 3D, while SurGe focuses on the limitations of convolutional decoding. We therefore further compare with these two recent methods to examine the effectiveness of our approach.

B.1 COMPARISON WITH MOGE-3

Table 4 reports the quantitative comparison with three-step MoGe-3 on the MDA Benchmark (Bian et al., 2026). Despite MoGe-3 benefiting from substantially larger and more diverse training data, our method achieves comparable global depth accuracy while yielding better boundary fidelity. Moreover, our single-pass design achieves lower inference latency. Fig. 7 further shows that MoGe-3 is faster with one or two refinement steps, but structural distortions remain. With three steps, its structures become qualitatively close to ours, but it is slower than PXDEPTH.

Table 4. **Quantitative comparison with iterative methods.** Acc and CD are in millimeters.

Method	Rel↓	δ_1 ↑	Acc↓	CD↓	Time(ms)↓
MoGe-3 (3 steps)	**3.4**	**97.0**	86.4	88.6	60.4
Ours	3.6	**97.0**	**66.5**	**70.6**	**48.9**

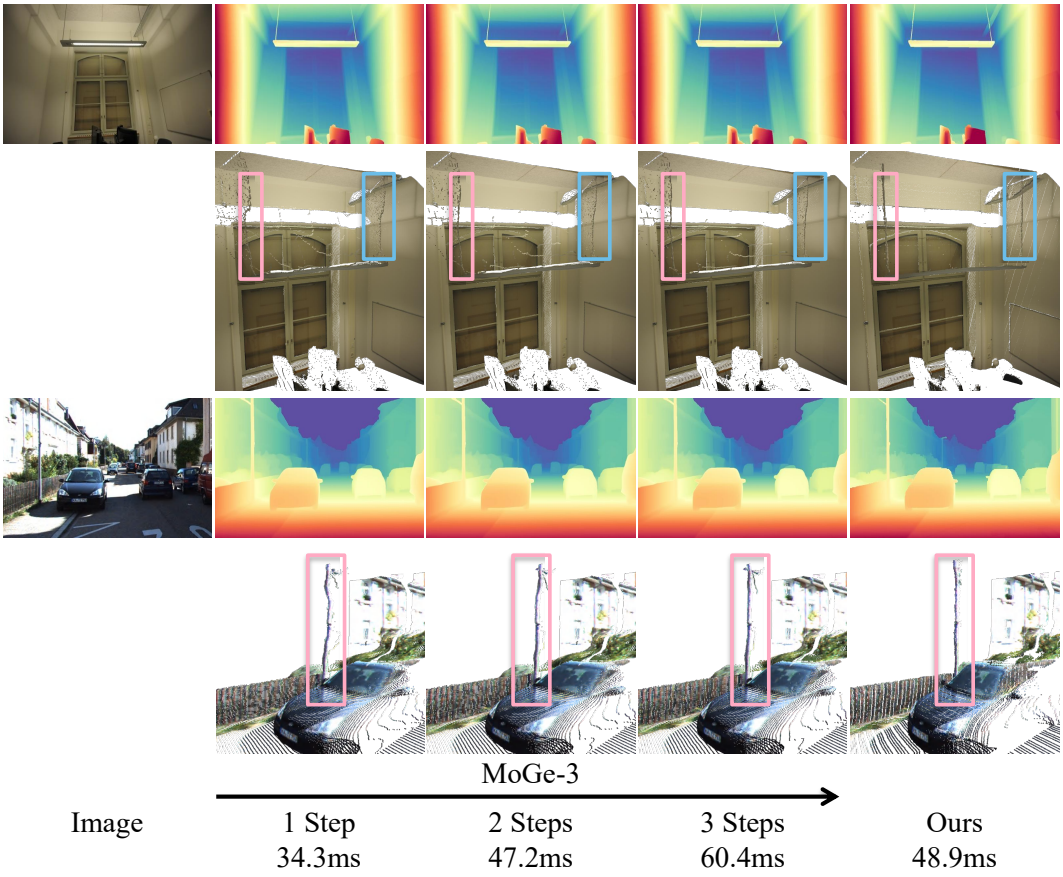


Figure 7. **Qualitative comparison with MoGe-3.** MoGe-3 progressively improves structural fidelity with more refinement steps, but this also increases inference latency. In contrast, PXDEPTH preserves fine structures in a single feed-forward pass.

B.2 Comparison with SurGe

SurGe (Knaebel et al., 2026) also studies local geometric distortions and focuses on the limitations of convolutional decoding. This makes it a particularly relevant comparison, since our analysis also identifies convolutional decoding as one source of structural distortion. We therefore compare with SurGe to further examine the sources of these distortions.

Table 5. **Quantitative comparison with SurGe.** Acc and CD are in millimeters.

Method	Rel↓	δ_1 ↑	Acc↓	CD↓
SurGe	**3.3**	**97.2**	79.1	78.8
Ours	3.6	97.0	**66.5**	**70.6**

Table 5 compares mean results on the MDA Benchmark. SurGe achieves slightly better global depth accuracy, while PXDepth yields substantially better boundary fidelity. The qualitative comparison in Fig. 8 further shows that, although SurGe already recovers well-preserved structures, PXDepth reconstructs small objects more faithfully. SurGe shows that fixed convolutional kernels struggle to model high-frequency and high-amplitude signals, which supports our analysis of convolution-based decoding. However, SurGe still reconstructs geometry from a low-resolution DINOv2 token grid. Our analysis considers both sources of distortion. Large-patch ViTs weaken fine-grained image cues, while convolutional decoding with spatially shared kernels struggles to handle low-frequency regions and high-frequency details simultaneously. In contrast, PXDepth builds pixel-space features directly from the input image and refines them under global context guidance. This design preserves more fine-grained information and recovers structural details more faithfully.

C Additional Implementation Details

This appendix provides additional details about the training datasets, optimization settings, and training objective.

C.1 Training Data and Configuration

Training Datasets. We train our model exclusively on synthetic RGB-D data. Stage 1 denotes low-resolution pretraining, while Stage 2 denotes high-resolution fine-tuning. Table 6 summarizes the data sources and their stage-specific sampling weights. These weights are relative factors used to construct each data mixture rather than percentages or dataset sizes.

Table 6. **Training dataset summary.** Relative sampling weights for both training stages, native RGB resolutions, and scene distributions of the synthetic datasets used for training.

Dataset	Scene distribution	Resolution	Sampling weight	
			Stage 1	Stage 2
Hypersim (Roberts et al., 2021)	Indoor	1024 × 768	6.0	10.0
MVS-Synth (Huang et al., 2018)	Urban	1920 × 1080	1.2	1.2
TartanAir (Wang et al., 2020)	Indoor / outdoor	640 × 480	3.0	3.0
TartanGround (Patel et al., 2025)	Indoor / outdoor, ground robot	640 × 640	4.8	4.8
UnrealStereo4K (Tosi et al., 2021)	Indoor / outdoor	3840 × 2160	2.4	2.4
GTA-SfM (Wang & Shen, 2020)	Urban / rural	640 × 480	2.8	2.8
Structured3D (Zheng et al., 2020)	Indoor	1280 × 720	4.8	4.8
UrbanSyn (Gómez et al., 2025)	Urban driving	2048 × 1024	2.1	2.1
ParallelDomain-4D (Van Hoorick et al., 2024)	Driving	640 × 480	3.5	3.5
VKITTI2 (Cabon et al., 2020)	Driving	1242 × 375	2.5	2.5
SceneNetRGBD (McCormac et al., 2017)	Indoor	320 × 240	3.6	–
RobbySim (Tan et al., 2026)	Indoor	1280 × 960	–	5.4
Synscapes (Wrenninge & Unger, 2018)	Urban driving	1440 × 720	4.0	4.0

Optimization Details. Training on NVIDIA RTX PRO 6000 GPUs requires approximately 1,120 GPU hours for low-resolution pretraining and 880 GPU hours for high-resolution fine-tuning. We configure AdamW with $\boldsymbol{\beta} = (0.9, 0.95)$ and a weight decay of 0.01. The learning rate of the Pixel-Space Depth Predictor is 1×10^{-4} in Stage 1 and 5×10^{-5} in Stage 2, while the Global Context Encoder is optimized with a learning rate of 5×10^{-6} throughout training. Both stages follow a cosine OneCycle schedule with a 2% warm-up, and the global gradient norm is clipped to 1.0.

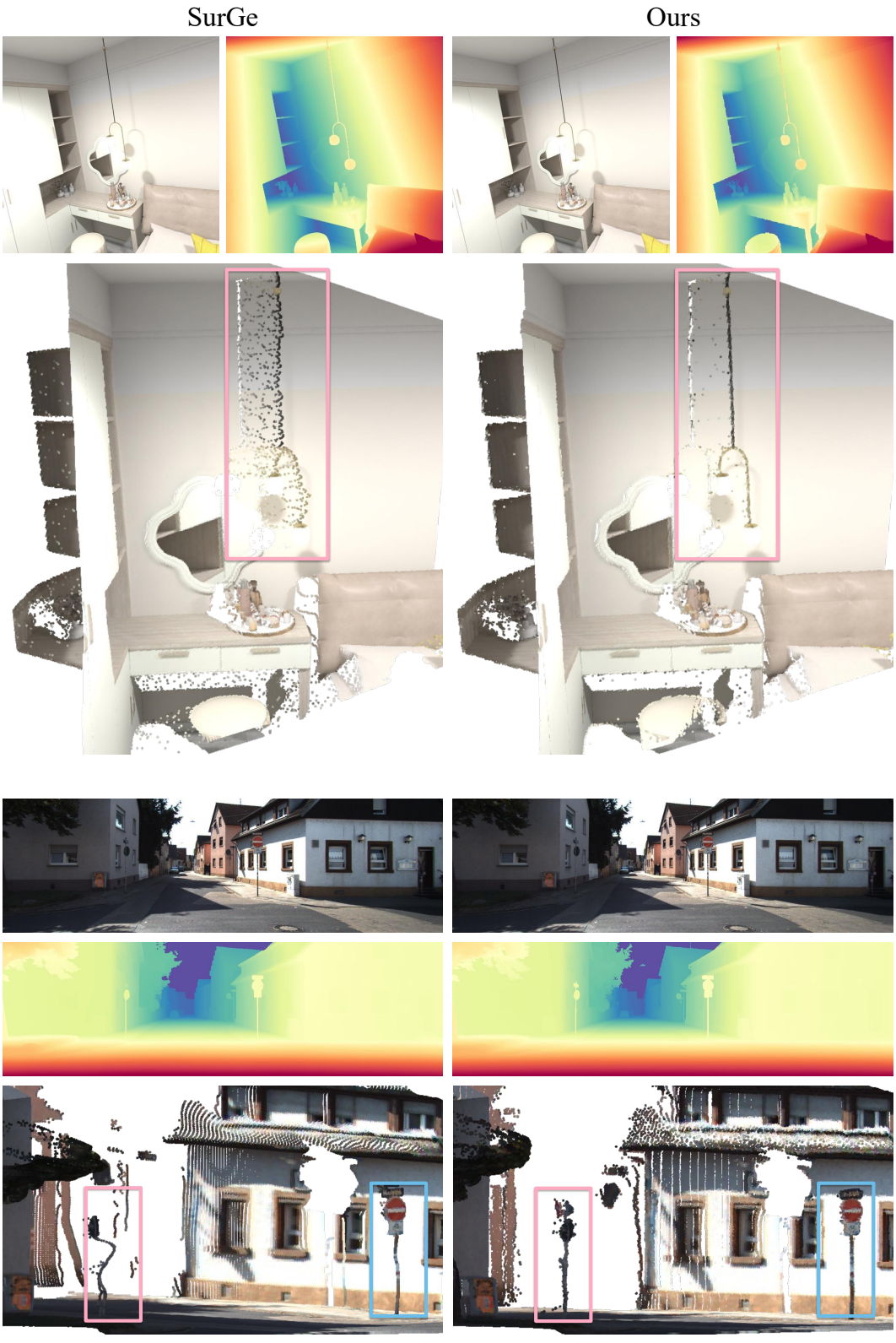


Figure 8. **Qualitative comparison with SurGe.** We show the input image together with the predicted depth map and reconstructed point map from SurGe and PXDEPTH. PXDEPTH recovers fine-grained structures more faithfully.

RGB images and depth maps undergo the same geometric transformations to maintain pixel-aligned supervision. We apply horizontal flipping with probability 0.5.

C.2 Training Objective

Depth Normalization. We normalize the ground-truth depth to reduce its dynamic range across scenes. Given a valid ground-truth depth value d, we first convert it to log depth and then normalize each depth map as

$$d_{\log} = \log(d+1), \qquad d_{\text{norm}} = \frac{d_{\log} - d_{0.02}}{d_{0.98} - d_{0.02}}, \tag{8}$$

where $d_{0.02}$ and $d_{0.98}$ denote the 2% and 98% percentiles of the valid log-depth values, respectively.

Normalized Depth Loss. Let $\hat{d}$ be the predicted normalized depth and $\mathcal{V}$ the set of valid depth pixels. We supervise the prediction with an ℓ_1 loss,

$$\mathcal{L}_{\text{depth}} = \frac{1}{|\mathcal{V}|} \sum_{u \in \mathcal{V}} \left| \hat{d}(u) - d_{\text{norm}}(u) \right|. \tag{9}$$

Multi-Scale Depth-Gradient Loss. Following prior work (Ranftl et al., 2022; Yang et al., 2024b; Bochkovskii et al., 2025), we adopt a standard multi-scale depth-gradient loss. To preserve local depth variations and sharp discontinuities, we supervise the gradients of the depth error $e = \hat{d} - d_{\text{norm}}$. At strides $s \in \{1, 2, 4, 8\}$, the loss is

$$\mathcal{L}_{\text{grad}} = \sum_{s \in \{1,2,4,8\}} \frac{1}{|\mathcal{V}_s|} \sum_{u \in \mathcal{V}_s} \left(|\nabla_x e_s(u)| + |\nabla_y e_s(u)| \right), \tag{10}$$

where e_s and $\mathcal{V}_s$ denote the error map and valid pixels sampled at stride s, respectively.

Binary Mask Loss. The mask head produces a logit for each pixel, which is converted by a sigmoid into the probability $\hat{m}(u) \in [0, 1]$ that the pixel belongs to the valid mask. Given the binary target $m(u)$ over the labeled pixel set $\mathcal{M}$, we use binary cross-entropy,

$$\mathcal{L}_{\text{mask}} = -\frac{1}{|\mathcal{M}|} \sum_{u \in \mathcal{M}} \left[m(u) \log \hat{m}(u) + \big(1 - m(u)\big) \log\big(1 - \hat{m}(u)\big) \right]. \tag{11}$$

At inference time, pixels with $\hat{m}(u) > 0.5$ are retained as valid.

Final Loss. The three losses are combined as

$$\mathcal{L} = \lambda_d \mathcal{L}_{\text{depth}} + \lambda_g \mathcal{L}_{\text{grad}} + \lambda_m \mathcal{L}_{\text{mask}}. \tag{12}$$

We set $\lambda_d = 1$, $\lambda_g = 0.5$, and $\lambda_m = 0.5$.

D More Architecture Details

D.1 Model Configuration

Global Context Encoder. We implement the Global Context Encoder with ViT-L/14, using a patch size of $P = 14$ and weights initialized from MoGe-2. Features from layers $\{5, 11, 17, 23\}$ are separately projected to $d_{\text{ctx}} = 1024$ channels with 1×1 convolutions. The projected feature maps are summed on the same $\frac{H}{P} \times \frac{W}{P}$ grid to obtain F_{ctx}. This aggregates context from different encoder layers without changing the spatial resolution of the token grid.

Pixel-Space Depth Predictor. The pixel embedding layer uses a 1×1 convolution to map each RGB pixel to a $d_{\text{pix}} = 16$-dimensional feature. The depth prediction path contains $N = 8$ CM-PiT blocks that refine these features under global context guidance from F_{ctx}. In our coarse-to-fine pixel compaction design, the first four blocks use $p = 14$ for efficient coarse-structure modeling, while the remaining four use $p = 7$ for fine-detail refinement. A final 1×1 convolution maps F^8_{pix} to the depth map. The mask head shares the pixel embedding and the first four blocks with the depth prediction path. It takes F^4_{pix} as input and applies two additional CM-PiT blocks with $p = 14$, followed by a 1×1 convolution that produces a validity logit for each pixel. We retain $p = 14$ in both mask blocks because using $p = 7$ provides little improvement in mask prediction while increasing computation.

D.2 Pixel Token Compaction

We implement the compaction and expansion operators $\mathcal{C}_p$ and $\mathcal{U}_p$ within $\mathcal{A}(\cdot)$ using learned linear projections. Given a pixel feature map, $\mathcal{C}_p$ partitions it into non-overlapping $p \times p$ regions. The p^2 pixel features in each region are flattened into a vector of dimension $p^2 d_{\text{pix}}$ and projected to an attention dimension of $d_{\text{attn}} = 1536$. This produces $(H/p)(W/p)$ compacted tokens. Self-attention operates on these tokens using 24 heads with 64 channels per head, query/key normalization, and 2D RoPE. The expansion operator $\mathcal{U}_p$ projects each attended token back to $p^2 d_{\text{pix}}$ dimensions and reshapes it into a $p \times p$ region at the corresponding pixel locations.

D.3 Alternative Guidance Strategies

We describe the direct-addition and cross-attention variants evaluated in Table 3 (c). Following the notation in the main paper, the equations below focus on context injection in the attention branch, with normalization omitted for clarity. For both alternatives, the feed-forward branch uses a standard residual update without context-generated modulation.

Direct Addition. This variant adds context features to compacted pixel tokens before self-attention. We formulate it as

$$\begin{aligned} T^i &= \mathcal{C}_p(F^i_{\text{pix}}) + \text{MLP}(F_{\text{ctx}}), \\ \bar{F}^i_{\text{pix}} &= F^i_{\text{pix}} + \mathcal{U}_p\big(\text{Self-Attn}(T^i, \text{RoPE})\big). \end{aligned} \tag{13}$$

Here, MLP maps each context token from d_{ctx} to d_{attn} channels, and T^i denotes the fused tokens. Context is combined with pixel features by element-wise addition on the compacted grid, and the attended tokens are expanded back to pixel space for the residual update.

Cross-Attention. This variant introduces an additional cross-attention operation before the self-attention update. Compacted pixel tokens provide the queries, while context features provide the keys and values through learned projections. The update is

$$\begin{aligned} \tilde{F}^i_{\text{pix}} &= \mathcal{U}_p(\text{Cross-Attn}(\mathcal{C}_p(F^i_{\text{pix}}), F_{\text{ctx}})), \\ \bar{F}^i_{\text{pix}} &= F^i_{\text{pix}} + \mathcal{A}(\tilde{F}^i_{\text{pix}}). \end{aligned} \tag{14}$$

Each compacted query token aggregates context values according to query–key similarities. The attended tokens are expanded back to pixel space to obtain $\tilde{F}^i_{\text{pix}}$, which is then processed by $\mathcal{A}(\cdot)$ for the residual update.

D.4 Context-Guided Adaptive Normalization

The context feature F_{ctx} is passed through a SiLU activation and a token-wise linear projection to generate modulation parameters. The projected values are reshaped into six pixel-aligned maps, each of size $H \times W \times d_{\text{pix}}$. Three maps provide the residual gate, shift, and scale for the attention branch, while the other three provide the corresponding parameters for the feed-forward branch.

We denote the attention parameters by α_a, β_a, and γ_a, which correspond to α, β, and γ in the main paper. The scale and shift modulate the normalized pixel features before token compaction. After self-attention and expansion, the residual gate weights the pixel-space update.

The feed-forward branch uses its own modulation maps, denoted by α_f, β_f, and γ_f, and follows the same update form:

$$F^{i+1}_{\text{pix}} = \bar{F}^i_{\text{pix}} + \alpha_f(F_{\text{ctx}}) \odot \text{FFN}\big(\text{RMSNorm}(\bar{F}^i_{\text{pix}}) \odot \gamma_f(F_{\text{ctx}}) + \beta_f(F_{\text{ctx}})\big), \tag{15}$$

where FFN denotes the pixel-wise SwiGLU network.

E Evaluation Protocol Details

E.1 Baseline Model Configurations

We evaluate all baselines using their official implementations and publicly released checkpoints. For DA V2 (Yang et al., 2024b), we adopt the relative-depth model with a ViT-L backbone. Depth-Pro (Bochkovskii et al., 2025) and InfiniDepth (Yu et al., 2026a) are evaluated with their publicly

released monocular models. For MoGe-2 (Wang et al., 2025b), we select the ViT-L variant that jointly predicts geometry and surface normals. PPD (Xu et al., 2025) is evaluated with the variant conditioned on semantic features from DA V2 and runs for 4 sampling steps. For PointDiT (Xu et al., 2026), we adopt the large model with 3 sampling steps. For MDA (Bian et al., 2026), we adopt the authors' primary DA3-Giant configuration with a Gaussian mixture-density head, a dedicated sky component, and an ℓ_2 objective in log-depth space.

E.2 Evaluation Datasets and Resolution Handling

Table 7 summarizes the datasets and dataset-specific evaluation resolutions used in our experiments. For each method, we first resize the input image to its target inference resolution. We then restore the predicted depth map to the corresponding evaluation resolution using nearest-neighbor interpolation before computing metrics or generating visualizations. This ensures that all methods are evaluated at the same resolution on each dataset, enabling a fair comparison. Nearest-neighbor interpolation avoids blending depth values across discontinuities, which can distort object boundaries.

Table 7. **Evaluation benchmark summary.** Configured evaluation resolutions and scene distributions of the zero-shot benchmarks.

Dataset	Type	Scene distribution	Resolution
7Scenes (Shotton et al., 2013)	Real	Indoor	504×378
NRGBD (Azinović et al., 2022)	Synthetic	Indoor	504×378
HiRoom (Lin et al., 2025)	Synthetic	Indoor	504×504
NYUv2 (Silberman et al., 2012)	Real	Indoor	640×480
KITTI (Geiger et al., 2012)	Real	Driving	1242×375
ETH3D (Schöps et al., 2017)	Real	Indoor / outdoor	1024×686
iBims-1 (Koch et al., 2018)	Real	Indoor	640×480
Sintel (Butler et al., 2012)	Synthetic	Animated scenes	872×436
DDAD (Guizilini et al., 2020)	Real	Urban driving	1400×700
DIODE (Vasiljevic et al., 2019)	Real	Indoor / outdoor	1024×768
HAMMER (Jung et al., 2023)	Real	Indoor tabletop	1664×832
Synth4K (Yu et al., 2026a)	Synthetic	Indoor / outdoor games	896×504

E.3 Global Depth Accuracy Evaluation Protocol

Different monocular depth estimators parameterize their predictions in different domains, e.g., depth, disparity, or log-depth. We therefore perform affine alignment in each method's native prediction domain, following the representation used by the corresponding model, rather than first converting all predictions to a common domain and imposing an additional nonlinear transformation. Specifically, PXDepth and PPD (Xu et al., 2025) are aligned in log-depth space, DA V2 (Yang et al., 2024b) and InfiniDepth (Yu et al., 2026a) in disparity space, and DepthPro (Bochkovskii et al., 2025), MoGe-2 (Wang et al., 2025b), PointDiT-L (Xu et al., 2026), MDA (Bian et al., 2026), MoGe-3 (Kong et al., 2026), and SurGe (Knaebel et al., 2026) in depth space. After estimating the optimal scale and shift, all predictions are converted to depth, where Rel and δ_1 are computed using the same valid pixels.

E.4 Boundary Evaluation Protocol

Following the MDA protocol (Bian et al., 2026), ground-truth depth is clipped to $[0.1, 65]$ meters, min–max normalized to an 8-bit image, and processed by a Canny detector with thresholds 100 and 200. Pixels adjacent to invalid depth are removed by one 2×2 dilation of the invalid mask. For PXDepth, we first align the prediction to the ground truth by a scale and shift in log-depth space, then convert it back to linear depth. Predicted and ground-truth depths at the remaining edge pixels are back-projected with the ground-truth intrinsics. Samples with fewer than ten valid boundary points are excluded from the 3D boundary metrics.

The predicted boundary point cloud is registered to the ground truth by the same point-to-point iterative closest point (ICP) registration as MDA with a correspondence threshold of 0.1 m. Accuracy (Acc) is the mean nearest-neighbor distance from predicted to ground-truth boundary points,

completeness is the reverse distance, and Chamfer distance (CD) is their average. Algorithm 1 summarizes the procedure.

Algorithm 1 **Boundary evaluation protocol.** Computation of 3D boundary metrics.

Input. Predicted depth, ground-truth depth, and camera intrinsics.
function EVALUATEBOUNDARY()
  Detect valid ground-truth depth boundaries.
  For PXDEPTH, align by a scale and shift in log-depth space and convert back to linear depth.
  Convert predicted and ground-truth depths at these boundaries into 3D point sets.
  Discard samples without enough valid boundary points.
  Align the predicted points to the ground truth using point-to-point ICP.
  Average predicted-to-ground-truth distances as Acc.
  Average ground-truth-to-predicted distances as completeness.
  Average Acc and completeness as CD.
  return Acc and CD in millimeters.
end function

F ADDITIONAL EXPERIMENTAL RESULTS

F.1 ABLATION ON THE NUMBER OF CM-PIT BLOCKS

We compare $N = 4$, 8, and 12 CM-PiT blocks on HiRoom in Table 8. Increasing N from 4 to 8 improves boundary fidelity at the cost of inference latency. Further increasing N to 12 increases latency without improving global depth accuracy or boundary fidelity. We therefore adopt $N = 8$ to balance prediction quality and inference efficiency.

Table 8. **Effect of the number of CM-PiT blocks on HiRoom.** Acc and CD are in millimeters. Params denotes the total parameter count. Memory denotes peak GPU memory. Time denotes median inference latency over 100 runs after 100 warmup runs, using FP16 and a batch size of 1.

N	Rel↓	δ_1 ↑	Acc↓	CD↓	Params (M)↓	Memory (GB)↓		Time (ms)↓	
						518×518	1022×770	518×518	1022×770
4	2.78	98.97	75.5	73.2	**497.8**	**2.35**	**3.01**	**30.6**	**122.0**
8 (Ours)	**2.74**	**99.01**	**70.3**	**72.5**	687.0	3.12	3.77	48.9	193.1
12	2.77	**99.01**	73.0	74.1	876.2	3.88	4.54	68.4	287.5

F.2 CENTER-CROP COMPARISON WITH POINTDIT

PointDiT-L is trained only at 512×512. In the main comparison, we directly resize its input images to the resolution of 512×512, which changes their original aspect ratios and may disadvantage PointDiT-L. To ensure a fair comparison, we therefore conduct an additional evaluation following PointDiT's official center-crop protocol. Each image is resized without changing its aspect ratio and then center-cropped to 512×512 for both methods. As shown in Tables 9 and 10, our method achieves comparable performance on the MoGe Benchmark and outperforms PointDiT-L in both global accuracy and boundary fidelity on the MDA Benchmark. These results further validate the effectiveness of our pixel-space feature learning.

Table 9. **Center-crop comparison with PointDiT on the MoGe Benchmark.** Both methods receive the same 512×512 center crop obtained after aspect-ratio-preserving resizing. PointDiT is trained only at 512×512, whereas our method supports images with arbitrary aspect ratios. We report this comparison for completeness.

Method	NYUv2		KITTI		ETH3D		iBims-1		Sintel		DDAD		DIODE		HAMMER		Mean	
	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑
PointDiT-L	3.0	98.4	4.0	**97.7**	**2.7**	**98.6**	2.4	**98.8**	**12.8**	**84.3**	**10.7**	87.8	**3.4**	**97.3**	**2.4**	**99.3**	**5.2**	**95.3**
Ours	**2.9**	**98.6**	**3.9**	**97.7**	2.9	98.4	**2.4**	**98.8**	13.6	82.9	11.1	**89.2**	4.1	96.3	3.0	98.0	5.5	95.0

Table 10. **Center-crop comparison with PointDiT on the MDA Benchmark.** Both methods receive the same 512×512 center crop obtained after aspect-ratio-preserving resizing.

Method	Global Accuracy								Boundary Fidelity							
	NRGBD		7Scenes		HiRoom		Mean		NRGBD		7Scenes		HiRoom		Mean	
	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Acc↓	CD↓	Acc↓	CD↓	Acc↓	CD↓	Acc↓	CD↓
PointDiT-L	1.8	98.6	6.3	93.7	2.0	99.0	3.4	97.1	102.0	88.3	63.1	75.4	68.7	62.6	77.9	75.4
Ours	**1.6**	**99.1**	**5.7**	**94.0**	**1.9**	**99.3**	**3.0**	**97.5**	**54.0**	**59.1**	**57.3**	**72.3**	**58.4**	**58.4**	**56.6**	**63.3**

F.3 Global Depth Evaluation on Synth4K

To complement the main evaluation, we provide additional quantitative results on Synth4K (Yu et al., 2026a) to probe zero-shot generalization across synthetic scenes. Introduced by InfiniDepth, Synth4K comprises five subsets of indoor and outdoor game scenes with dense synthetic depth. The results are reported in Table 11.

Table 11. **Quantitative comparison on the Synth4K Benchmark.** Zero-shot affine-invariant depth estimation across its five subsets. * indicates direct resizing to the required input resolution.

Category	Method	Synth4K-1		Synth4K-2		Synth4K-3		Synth4K-4		Synth4K-5		Mean	
		Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑	Rel↓	δ_1 ↑
Global Prediction	DA V2	17.9	80.1	11.9	87.1	17.7	83.5	8.4	93.7	8.2	92.7	12.8	87.4
	DepthPro*	13.0	81.8	9.0	87.3	10.0	86.2	7.3	92.6	6.7	92.7	9.2	88.1
	MoGe-2	**10.0**	**86.4**	**7.6**	**89.9**	**7.6**	**91.3**	**5.4**	**94.8**	**4.9**	**95.5**	**7.1**	**91.6**
Structure Aware	PPD	14.5	78.7	13.0	79.8	15.4	76.0	7.6	92.4	10.2	85.2	12.2	82.4
	InfiniDepth	13.6	85.4	10.4	87.4	13.0	85.0	7.9	94.5	6.2	95.1	10.2	89.5
	PointDiT-L*	11.2	**87.0**	8.5	**89.8**	7.9	**92.1**	6.4	93.2	5.4	95.0	7.9	91.4
	MDA	22.6	69.2	16.3	80.7	13.9	85.1	14.3	84.4	14.9	81.2	16.4	80.1
	Ours	**9.9**	85.7	**7.7**	89.7	**7.6**	91.0	**5.2**	**94.9**	**4.4**	**96.0**	**7.0**	**91.5**

F.4 Boundary Evaluation on iBims-1 and HAMMER

We further report boundary metrics on iBims-1 (Koch et al., 2018) and HAMMER (Jung et al., 2023), two real-world datasets included in the MoGe Benchmark that cover indoor and tabletop scenes. The results are reported in Table 12.

Table 12. **Quantitative comparison on iBims-1 and HAMMER.** Acc and CD are reported in millimeters. * indicates direct resizing to the required input resolution.

Category	Method	iBims-1		HAMMER		Mean	
		Acc↓	CD↓	Acc↓	CD↓	Acc↓	CD↓
Global Prediction	DA V2	145.9	203.2	35.9	43.3	90.9	123.3
	DepthPro*	138.6	196.6	32.9	40.3	85.8	118.4
	MoGe-2	**116.3**	**175.2**	**29.5**	**36.7**	**72.9**	**105.9**
Structure Aware	PPD	126.4	273.1	23.6	33.8	75.0	153.4
	InfiniDepth	157.7	247.4	28.5	38.6	93.1	143.0
	PointDiT-L*	126.0	235.1	39.6	41.1	82.8	138.1
	MDA	121.7	172.7	22.3	32.5	72.0	102.6
	Ours	**112.5**	**162.3**	**20.8**	**29.4**	**66.7**	**95.9**

F.5 Additional Qualitative Results

Figures 9 & 10 extend the qualitative evaluation to diverse scenes. Across both indoor and outdoor examples, our method more faithfully preserves thin structures and object boundaries, producing cleaner point-cloud reconstructions with fewer flying points than competing methods.

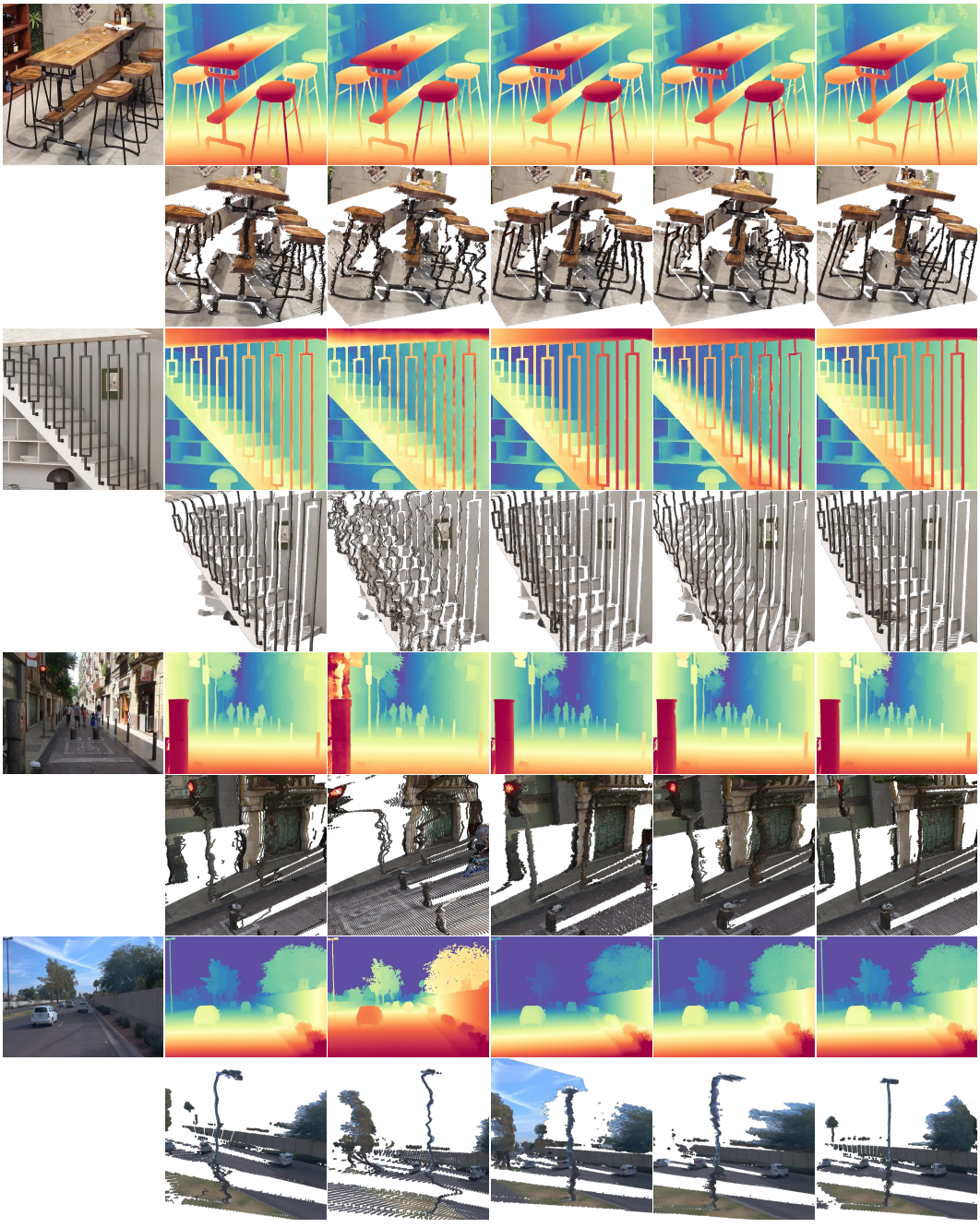


Figure 9. **Additional qualitative comparisons.** Our method preserves fine-grained structures.

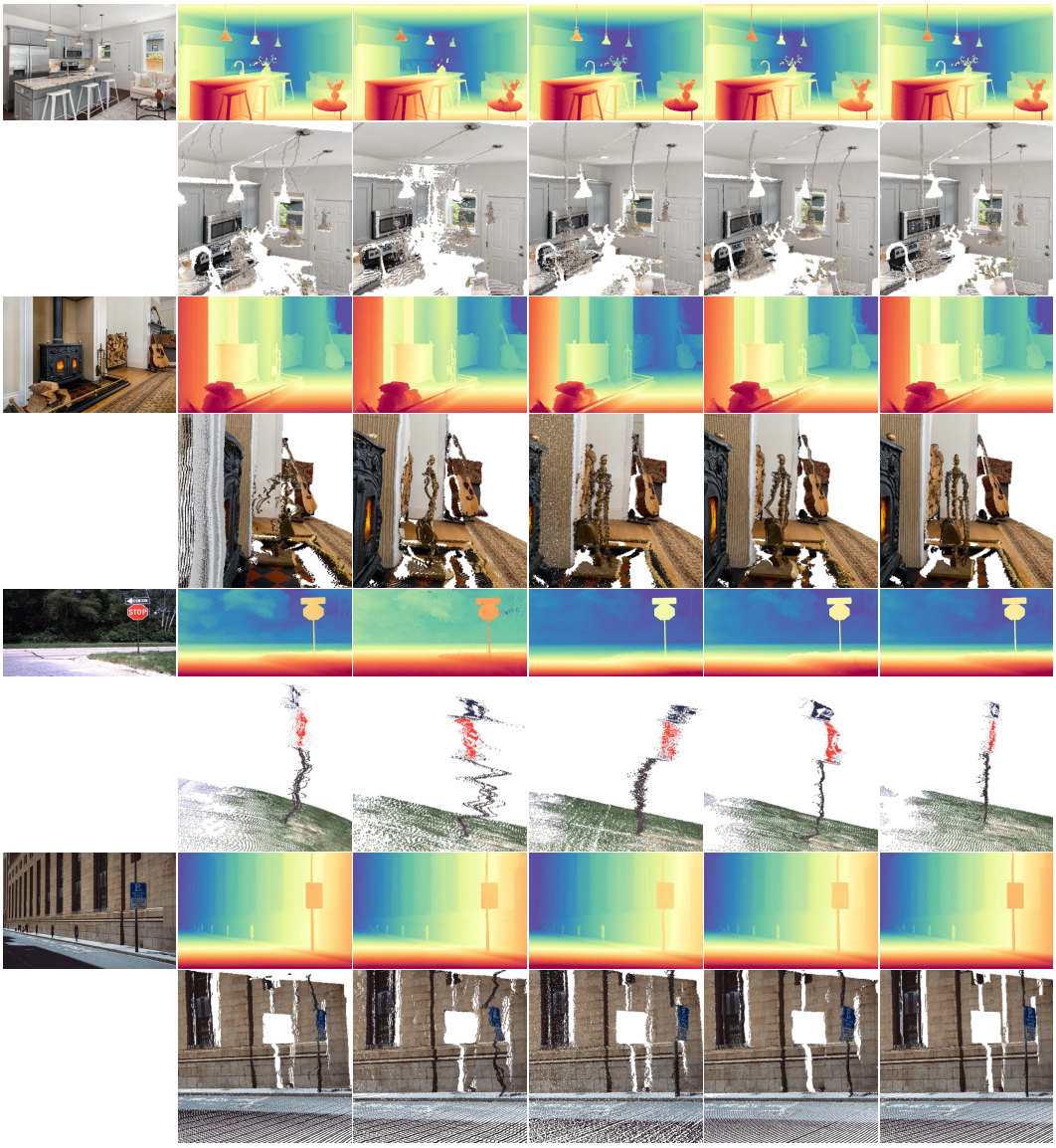


Figure 10. **Additional qualitative comparisons.** Our method preserves fine-grained structures.