METRIC SCALE PROPAGATION NETWORK FOR ZERO-SHOT DEPTH COMPLETION

Anonymous authors
Paper under double-blind review

ABSTRACT

Zero-shot depth completion aims to recover dense metric depth (i.e., physical distance) from RGB images and sparse depth observations in unseen scenes. We identify that the key challenge of this task is to preserve the critical scene scale from the input sparse depth to the final dense prediction. Existing end-to-end methods generally normalize scene scales to facilitate generalization and stabilize training, which may distort the scene scale. Recent depth-foundation-model-based methods substantially improve generalization via relative depth priors, which lack the scale information. To mitigate this issue, we propose **MSP-Net**, a Metric Scale Propagation Network that explicitly identifies and propagates scene scale through an end-to-end process. It comprises two main modules: the Scale Identification Module (SIM) and the Scale Propagation Module (SPM). SIM identifies reliable sparse metric observations and estimates scene scale at sparse locations. SPM then propagates these scale cues from sparse locations to the whole image, producing a dense scale field. Optionally, a pretrained depth foundation model is integrated to further boost cross-scene generalization. Experiments are conducted on six unseen benchmarks with diverse sparse depth patterns. Without pretraining, our MSP-Net outperforms all recent end-to-end methods and even most pretrained methods. With pretraining, our MSP-Net outperforms all recent pretrained methods with faster inference.

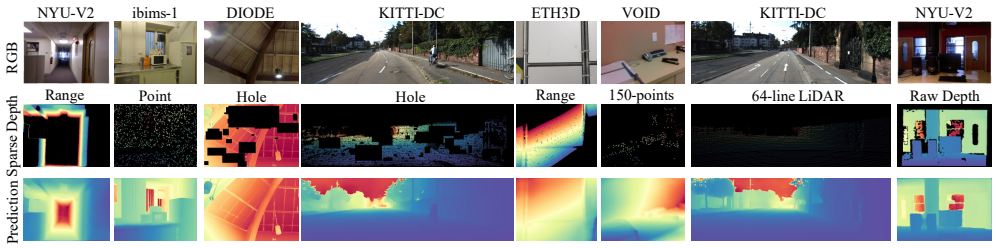


Figure 1: MSP-Net generalizes robustly across diverse unseen indoor and outdoor datasets with both simulated sparse depth inputs (Range/Point/Hole) and raw sparse depth inputs(150-points/64-line Lidar/raw depth).

1 INTRODUCTION

Depth completion aims to recover dense metric depth (i.e., physical distance) from RGB images and sparse depth observations (Ma & Karaman, 2018; Uhrig et al., 2017; Shivakumar et al., 2019). It provides reliable 3D geometry for autonomous driving (Philion & Fidler, 2020; Li et al., 2023), robotic navigation (Zhu et al., 2022), augmented reality (Watson et al., 2023), and 3D reconstruction (Mildenhall et al., 2021). While existing methods have achieved remarkable performance under in-domain settings (Park et al., 2020a; Zhang et al., 2023; Tang et al., 2024), their performance often degrades significantly in unseen scenes. Zero-shot depth completion has recently emerged as a new paradigm, aiming to recover metric depth in unseen scenes without target-domain training.

We identify that, in zero-shot depth completion, the key challenge is to recover the critical scale of unseen scenes. Scene scale refers to the conversion factor between a dimensionless 3D representation and physical distances. It bridges the gap between relative geometry and metric depth. Unlike in-domain depth completion, in which the scales of training and test data are often the same, the

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.

scale of unseen scenes is often unknown in zero-shot depth completion. Fortunately, sparse depth observations partly provide the scale information. Therefore, it is crucial to preserve the scene scale from input sparse depth observations to final dense prediction.

In recent years, end-to-end depth completion methods have been proposed to improve cross-scene generalization, such as G2-Mono (Wang et al., 2023a), OGNI-DC (Zuo & Deng, 2024), and OMNI-DC (Zuo et al., 2025). To improve generalization and stabilize training, these methods commonly normalize scene scales during training. However, this normalization causes scene scales to be gradually distorted through normalization and feature learning. They often struggle to preserve the scene scale of sparse depth throughout the network.

More recently, depth foundation models have emerged to predict relative depth through large-scale training, such as Depth Anything (Yang et al., 2024a), Depth Anything V2 (Yang et al., 2024b), and Marigold (Ke et al., 2024). Depth-foundation-model-based methods have substantially improved the generalization of depth completion by leveraging transferable geometric and semantic priors, such as DPrompt (Park et al., 2024), PromptDA (Lin et al., 2025), PriorDA (Wang et al., 2025b), and Mari-DC (Viola et al., 2025). However, since depth foundation models estimate only relative depth, they inherently lack the scene scale information. These methods still need to preserve scene scale from the input sparse depth to the final dense prediction.

To mitigate this issue, we propose the Metric Scale Propagation Network (MSP-Net), which is designed to preserve scene scale throughout the end-to-end depth completion process. MSP-Net mainly consists of two modules: the Scale Identification Module (SIM) and the Scale Propagation Module (SPM). SIM determines which sparse depth points are more reliable and extracts scene-level scale information. SPM propagates the reliable sparse scale cues to the whole image to recover the final metric depth.

Our SIM fuses RGB and sparse depth features to estimate the reliability of each sparse depth point. Sparse points on stable surfaces and consistent structures are assigned higher reliability, while sparse points near boundaries or noisy regions are down-weighted. SIM outputs a reliability map, a global scale anchor, and multi-level scale tokens, which inform the following network about which sparse depth points are more trustworthy and what the overall scene scale is.

Our SPM uses reliable sparse points to propagate scale information into regions without depth observations. Scale information is preferentially propagated between regions that are visually similar and spatially adjacent, while propagation across distinct object or structural boundaries is suppressed. SPM ultimately produces a dense scale map, which guides the subsequent depth completion process and serves as the scale basis for recovering the final metric depth.

Experiments are conducted on six unseen benchmark datasets with diverse patterns of raw and simulated sparse depth. Without pretraining, our MSP-Net outperforms all recent end-to-end depth completion methods and even most pretrained depth-foundation-model-based methods. With pretraining, our MSP-Net outperforms all recent pretrained methods. Meanwhile, our MSP-Net requires fewer parameters and achieves faster inference. Figure 1 further illustrates the robust generalization of MSP-Net across diverse unseen datasets and sparse depth patterns.

Our main contributions are summarized as follows:

- We identify that the devil of zero-shot depth completion is to preserve the critical scene scale from input sparse depth observations to final dense prediction.
- We propose MSP-Net to preserve scale throughout the end-to-end depth completion process. It comprises two main modules: SIM identifies reliable scale anchors from sparse depth and SPM propagates scale to the whole image.
- Our MSP-Net achieves state-of-the-art performance with faster inference than both end-to-end methods and pretrained depth-foundation-model-based methods.

2 Related Work

2.1 In-domain Depth Completion

Depth completion has achieved remarkable progress in recent years. Early methods mainly focused on improving prediction accuracy under in-domain settings. GuideNet (Tang et al., 2020) introduces guided convolution to effectively fuse RGB images with sparse depth observations. PENet (Hu et al., 2021) enhances depth completion by progressively exploiting geometric relationships between RGB

and sparse depth. NLSPN (Park et al., 2020b) performs non-local spatial propagation to refine dense depth predictions. LRRU (Wang et al., 2023b) adopts a lightweight residual refinement framework to progressively recover accurate dense depth.

Because these methods are trained and evaluated on the same dataset, they achieve high in-domain accuracy. However, when deployed to unseen scenes, the scene scale can no longer be reliably inferred, resulting in poor generalization.

2.2 Zero-shot Depth Completion

To improve cross-scene generalization, recent studies have shifted toward zero-shot depth completion through end-to-end training. G2-Mono (Wang et al., 2023a) adopts a unified multi-dataset training strategy to enhance generalization across heterogeneous sensing modalities. OGNI-DC (Zuo & Deng, 2024) formulates depth completion as optimization-guided neural iterations for robust refinement under unseen domains. SPNet (Wang et al., 2024) preserves scale-related feature magnitudes through scale-preserving normalization, while OMNI-DC (Zuo et al., 2025) further improves robustness via multi-resolution integration and large-scale multi-dataset training. However, to improve generalization and stabilize optimization, these methods commonly normalize scene scales during training. Consequently, they often fail to preserve the scene scale of sparse depth observations throughout the network.

2.3 Depth-Foundation-Model-Based Methods

Pretrained depth foundation models have substantially improved the generalization ability of depth completion by providing transferable geometric and semantic priors. Existing depth-foundation-model-based methods mainly differ in how sparse metric observations are integrated into these pretrained models. DPrompt (Park et al., 2024) and PromptDA (Lin et al., 2025) inject sparse depth as prompts to adapt pretrained representations. PriorDA (Wang et al., 2025b) progressively fuses dense relative predictions with sparse metric observations for depth refinement. TestDC (Jeong et al., 2025) and Mari-DC (Viola et al., 2025) exploit sparse depth during inference through prompt optimization or diffusion guidance without retraining the backbone. DMD^{3}C (Liang et al., 2025) transfers knowledge from a large foundation model to a lightweight depth completion network through distillation.

However, since depth foundation models estimate only relative depth, they inherently lack scene scale information. Consequently, depth-foundation-model-based methods still need to preserve the scene scale from the input sparse depth to the final dense prediction.

3 Method

3.1 Overview

Given an RGB image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$, a sparse depth map $\mathbf{D}_s \in \mathbb{R}^{H\times W\times 1}$, and a mask of valid pixels $\mathbf{M} \in \{0,1\}^{H\times W\times 1}$, our goal is to predict a dense metric depth map $\widehat{\mathbf{D}} \in \mathbb{R}^{H\times W\times 1}$. Figure 2 (a) illustrates the framework of our MSP-Net. It performs end-to-end depth completion through four steps:

(I) **Scale Identification Module (SIM)**. SIM takes the RGB image $\mathbf{I}$, sparse depth $\mathbf{D}_s$, and the mask $\mathbf{M}$ as input to predict a reliability map $\mathbf{A} \in [0,1]^{H\times W\times 1}$, a global scale anchor $s_g \in \mathbb{R}_+$, and multi-level scale tokens $\{\mathbf{t}_l\}_{l=1}^{L}$, where $\mathbf{t}_l \in \mathbb{R}^{C_l}$ denotes the scale token for the l-th encoder stage.

(II) **Scale Propagation Module (SPM)**. SPM then propagates reliable sparse metric observations to produce a dense scale field $\mathbf{S} \in \mathbb{R}_+^{H\times W\times 1}$. Next, $\mathbf{S}$ is used to normalize the sparse depth, yielding the normalized sparse depth $\mathbf{D}_n \in \mathbb{R}^{H\times W\times 1}$, and is further normalized by s_g to obtain the normalized scale field $\mathbf{S}_n \in \mathbb{R}^{H\times W\times 1}$.

(III) **Residual Depth Prediction**. The residual depth prediction network takes $\mathbf{I}$, $\mathbf{D}_n$, $\mathbf{M}$, and $\mathbf{S}_n$ as input, while injecting the scale tokens into the corresponding encoder stages to predict a normalized residual depth map $\mathbf{R}_\Delta \in \mathbb{R}^{H\times W\times 1}$.

(IV) **Metric Depth Restoration**. The predicted residual depth is combined with the propagated scale field $\mathbf{S}$ and the global scale anchor s_g to restore the final dense depth $\widehat{\mathbf{D}}$.

3.2 Scale Identification Module

In Figure 2 (b), sparse depth provides direct metric observations, but not every sparse measurement is equally reliable. Measurements near object boundaries, depth discontinuities, or noisy regions often

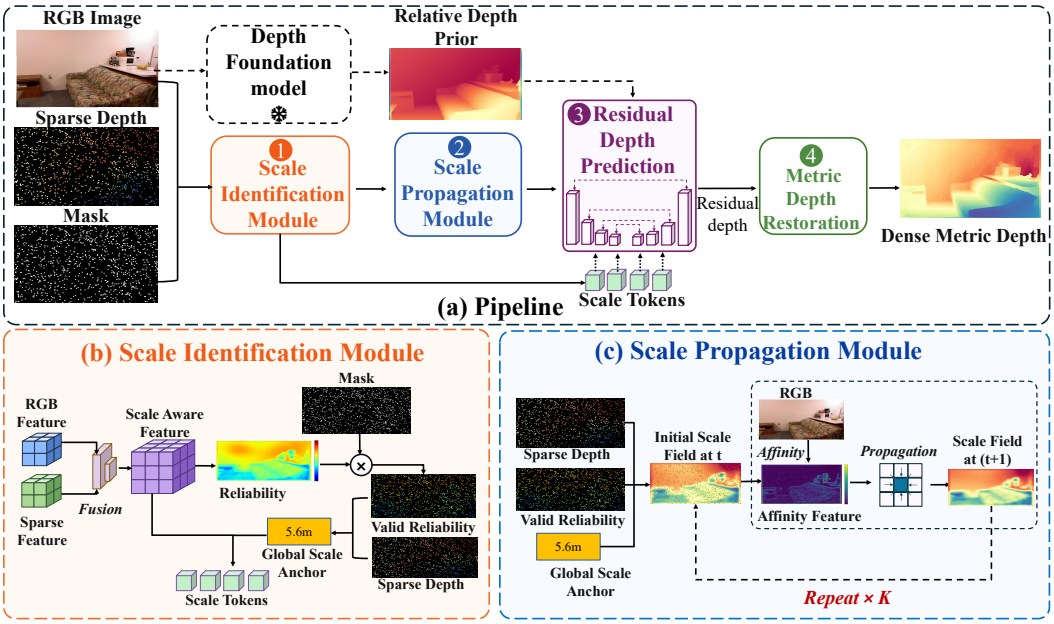


Figure 2: Overview of our MSP-Net. MSP-Net performs metric depth completion in four steps: (1) Scale Identification Module (SIM), (2) Scale Propagation Module (SPM), (3) Residual Depth Prediction, and (4) Metric Depth Restoration. Optionally, a pretrained depth foundation model can be incorporated to further boost cross-scene generalization.

provide inaccurate scale cues. Therefore, SIM first predicts a pixel-wise reliability map from the RGB image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$, sparse depth $\mathbf{D}_s \in \mathbb{R}^{H\times W\times 1}$, and validity mask $\mathbf{M} \in \{0,1\}^{H\times W\times 1}$:

$$\mathbf{A} = \sigma(\mathrm{Conv}_r([\mathrm{Conv}_{rgb}(\mathbf{I}), \mathrm{Conv}_d([\mathbf{D}_s, \mathbf{M}])])) \odot \mathbf{M} \tag{1}$$

Here, $\sigma(\cdot)$ denotes the sigmoid activation function; Conv_{rgb} and Conv_d denote the convolutional encoders for the RGB and sparse-depth branches, respectively; Conv_r denotes the convolutional layer for reliability prediction; and $\odot$ denotes element-wise multiplication.

SIM then summarizes reliable sparse observations into a global scene-scale anchor:

$$s_g = \frac{\sum_i A_i D_{s,i}}{\sum_i A_i + \epsilon} \tag{2}$$

Here, $A_i \in [0,1]$ denotes the reliability at pixel i, $D_{s,i} \in \mathbb{R}_+$ is the corresponding sparse depth value, and ϵ is a small constant for numerical stability. The resulting scalar s_g represents the overall scene scale of the current scene estimated from trustworthy sparse observations.

Besides the reliability map and global scale anchor, SIM further extracts a compact scene-level scale representation from reliability-weighted RGB and sparse depth features. The representation is projected into multi-level scale tokens $\{\mathbf{t}_l\}_{l=1}^{L}$, where $\mathbf{t}_l \in \mathbb{R}^{C_l}$. Each token is injected into the corresponding encoder stage, enabling scene scale information to guide multi-scale feature learning throughout the network.

3.3 Scale Propagation Module

In Figure 2 (c), although SIM identifies reliable metric observations, these observations remain spatially sparse and cannot directly provide complete scene-level scale guidance. SPM therefore propagates reliable scale cues into unobserved regions and produces a dense scale field $\mathbf{S} \in \mathbb{R}_+^{H\times W\times 1}$.

We first initialize the scale field using the reliability map $\mathbf{A} \in [0,1]^{H\times W\times 1}$, sparse metric depth $\mathbf{D}_s \in \mathbb{R}^{H\times W\times 1}$, and global scale anchor $s_g \in \mathbb{R}_+$:

$$\mathbf{S}^0 = \mathbf{A} \odot \mathbf{D}_s + (1-\mathbf{A}) \odot s_g \tag{3}$$

Here, $\mathbf{S}^0 \in \mathbb{R}_+^{H\times W\times 1}$ denotes the initial scale field. At reliable sparse depth locations, the initialization mainly preserves the observed metric values. At unreliable or missing locations, the global scale anchor provides an initial estimate.

To guide scale propagation, SPM extracts an affinity feature map $\mathbf{F}_a \in \mathbb{R}^{H\times W\times C_a}$ from the RGB image and normalizes it along the channel dimension. For each pixel, SPM compares its affinity feature with those of neighboring pixels and combines the resulting appearance similarity with a spatial distance penalty. The normalized propagation weight from pixel j to pixel i is denoted by $w_{ij} \in [0, 1]$, satisfying $\sum_{j\in\mathcal{N}(i)} w_{ij} = 1$, where $\mathcal{N}(i)$ denotes the local neighborhood of pixel i.

The scale field is then iteratively propagated as

$$S_i^{k+1} = A_i D_{s,i} + (1 - A_i) \sum_{j\in\mathcal{N}(i)} w_{ij} S_j^k \tag{4}$$

Here, $S_i^{k+1} \in \mathbb{R}_+$ denotes the scale value at pixel i after the $(k + 1)$-th propagation iteration. The first term preserves reliable sparse metric observations, while the second term aggregates scale information from visually similar and spatially nearby pixels. In this way, scale propagation across clear object or structural boundaries is suppressed.

After K propagation iterations, the final dense scale field is $\mathbf{S} = \mathbf{S}^K \in \mathbb{R}_+^{H\times W\times 1}$. It provides spatially dense metric-scale guidance for the subsequent depth completion process.

3.4 Residual Depth Prediction

The propagated scale field $\mathbf{S} \in \mathbb{R}_+^{H\times W\times 1}$ serves as an explicit carrier of scene scale. Before depth completion, we normalize the sparse depth and dense scale field as

$$\mathbf{D}_n = \frac{\mathbf{D}_s}{\mathbf{S} + \epsilon}, \qquad \mathbf{S}_n = \frac{\mathbf{S}}{s_g + \epsilon} \tag{5}$$

Here, $\mathbf{D}_n \in \mathbb{R}^{H\times W\times 1}$ denotes the scale-normalized sparse depth and $\mathbf{S}_n \in \mathbb{R}^{H\times W\times 1}$ denotes the normalized dense scale field. This normalization allows the network to perform geometry completion in a more scale-stable space while preserving an explicit path for scene scale restoration.

The completion network takes the RGB image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$, normalized sparse depth $\mathbf{D}_n \in \mathbb{R}^{H\times W\times 1}$, validity mask $\mathbf{M} \in \{0, 1\}^{H\times W\times 1}$, and normalized scale field $\mathbf{S}_n \in \mathbb{R}^{H\times W\times 1}$ as input. These inputs provide appearance information, normalized sparse geometry, measurement validity, and spatially dense scale guidance, respectively.

At the l-th encoder stage, the scale token $\mathbf{t}_l \in \mathbb{R}^{C_l}$ is transformed into channel-wise modulation parameters and injected into the encoder feature $\mathbf{E}_l \in \mathbb{R}^{H_l\times W_l\times C_l}$ through ScaleFiLM. This enables the scene-level scale representation identified by SIM to condition multi-scale feature learning throughout the network. The decoder predicts a residual depth map $\mathbf{R}_\Delta \in \mathbb{R}^{H\times W\times 1}$.

3.5 Metric Depth Restoration

The final dense metric depth is restored by

$$\widehat{\mathbf{D}} = \mathbf{S} \odot (1 + \mathbf{R}_\Delta) \tag{6}$$

Here, $\widehat{\mathbf{D}} \in \mathbb{R}_+^{H\times W\times 1}$ is the final metric depth prediction. In this formulation, the propagated scale field $\mathbf{S}$ provides the pixel-wise metric baseline, while $\mathbf{R}_\Delta$ refines local geometry in the normalized completion space.

3.6 Pretraining with Depth Foundation Model

To further improve cross-domain generalization, MSP-Net can optionally incorporate a pretrained monocular depth foundation model. Given the RGB image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$, the frozen depth foundation model predicts a dense depth prior $\mathbf{D}_f \in \mathbb{R}^{H\times W\times 1}$.

The dense depth prior $\mathbf{D}_f$ is concatenated with the RGB image, normalized sparse depth, validity mask, and normalized scale field as an additional input to the completion network. In this way, the

propagated scene scale from SPM provides explicit metric constraints, while the depth-foundation-model prior supplies dense structural guidance for regions without sparse measurements.

3.7 Training Objective

Let $\mathbf{D}_{gt} \in \mathbb{R}^{H\times W\times 1}$ denote the ground-truth dense depth and $\mathbf{M}_{gt} \in \{0,1\}^{H\times W\times 1}$ its validity mask. MSP-Net is optimized using an absolute depth loss, a standardized relative depth loss, and a multi-scale gradient loss.

The absolute depth loss supervises metric depth prediction at valid sparse observation locations:

$$\mathcal{L}_{\text{abs}} = \frac{\left\| \mathbf{M} \odot (\widehat{\mathbf{D}} - \mathbf{D}_{gt}) \right\|_1}{\sum_i M_i + \epsilon} \tag{7}$$

To constrain scene geometry independently of the global scale, we compute the relative loss by comparing the standardized prediction with the standardized GT depth over valid pixels. Let $\widetilde{\mathbf{D}} = \text{Std}(\widehat{\mathbf{D}}, \mathbf{M}_{gt})$ and $\widetilde{\mathbf{D}}_{gt} = \text{Std}(\mathbf{D}_{gt}, \mathbf{M}_{gt})$ denote the standardized prediction and GT depth, respectively. The relative loss is defined as

$$\mathcal{L}_{\text{rel}} = \frac{\left\| \mathbf{M}_{gt} \odot (\widetilde{\mathbf{D}} - \widetilde{\mathbf{D}}_{gt}) \right\|_1}{\sum_i M_{gt,i} + \epsilon} \tag{8}$$

To preserve fine structures and sharp depth discontinuities, we further introduce a multi-scale gradient loss:

$$\mathcal{L}_{\text{grad}} = \frac{\sum_{k=0}^{K_g-1} \left\| \nabla \mathcal{D}_{2^k} \left[\mathbf{M}_{gt} \odot (\widetilde{\mathbf{D}} - \widetilde{\mathbf{D}}_{gt}) \right] \right\|_1}{\sum_i M_{gt,i} + \epsilon} \tag{9}$$

Here, ∇ denotes the Sobel gradient operator, $\mathcal{D}_{2^k}(\cdot)$ denotes downsampling by a factor of 2^k, and K_g is the number of gradient scales.

The overall training loss is:

$$\mathcal{L} = \mathcal{L}_{\text{abs}} + \mathcal{L}_{\text{rel}} + \lambda \mathcal{L}_{\text{grad}} \tag{10}$$

Here, λ is the weighting coefficient for the gradient loss, and is set to 0.5 following MegaDepth (Li & Snavely, 2018) and MiDaS(Ranftl et al., 2020).

4 Experiments

4.1 Experiment Setup

4.1.1 Training datasets

We train the proposed method on a mixture of four datasets, including Virtual KITTI (Gaidon et al., 2016), HRWSI (Xian et al., 2020), Matterport3D (Chang et al., 2017), and UnrealCV (Qiu & Yuille, 2016). These datasets provide complementary indoor and outdoor scenes with diverse semantics, depth ranges, and geometric structures. We follow the sparse depth synthesis and data augmentation strategies used in G2-Mono (Wang et al., 2023a) during training.

4.1.2 Evaluation Settings

We evaluate our method under two representative settings of sparse depth patterns.

(I) *Simulated sparse depth patterns*. We consider three simulated sparse depth patterns: Hole, Range, and Point, which respectively remove spatially contiguous regions, retain depth values within the 20%–80% valid depth range, and randomly sample 1,000 valid ground-truth depth points. We conduct experiments on five unseen datasets: NYU-V2 (Silberman et al., 2012), iBims-1 (Koch et al., 2018), DIODE (Vasiljevic et al., 2019), ETH3D (Schops et al., 2017), and KITTI-DC (Uhrig et al., 2017).

Table 1: Comparison on simulated sparse depth patterns. "Rk" denotes the average rank in each setting (without/with pretraining). "OA Rk" denotes the average rank among all methods. The best and second-best results are in bold and underlined, respectively. Without pretraining, our MSP-Net outperforms all end-to-end methods and most pretrained methods. With pretraining, our MSP-Net outperforms all pretrained methods. Meanwhile, our MSP-Net remains moderately low complexity in each setting.

Method	Pre	Para (M)	DIODE		ETH3D		iBims-1		KITTI-DC		NYU-V2		Rk	OA Rk
			MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE		
CFormer^{K} (CVPR'23)	–	83.5	1847	3555	839	1711	1857	3198	1844	4406	1835	3198	7.9	14.1
CFormer^{N} (CVPR'23)	–	83.5	2781	6029	2115	2945	816	1040	4547	19440	667	842	8.5	15.3
Stereo-DC (3DV'24)	–	11.1	1640	3362	1177	2406	265	729	1520	4079	235	4464	6.7	11.6
OGNI-DC^{K} (ECCV'24)	–	84.4	1387	2856	286	761	374	788	1181	3284	216	455	4.8	8.7
OGNI-DC^{N} (ECCV'24)	–	84.4	1393	2934	649	1381	431	831	2736	6599	349	657	6.6	11.6
G2-Mono (TPAMI'24)	–	18.2	909	2784	**101**	576	142	459	1079	3271	122	353	2.4	5.2
SPNet (TPAMI'25)	–	35.0	**803**	2729	132	671	**112**	432	**858**	3565	101	398	2.4	5.2
OMNI-DC (ICCV'25)	–	85.5	1088	3246	121	636	180	567	1294	3925	149	362	4.0	7.8
Ours	–	36.7	921	**2118**	138	**355**	118	**301**	958	**3042**	**91**	**195**	**1.7**	3.6
PromptDA (CVPR'25)	✓	340.0	5060	6982	1889	2683	1291	1784	9571	14624	1056	1431	8.0	16.2
DPrompt^{K} (CVPR'24)	✓	327.0	2098	3335	1278	1676	1005	1422	1160	**2264**	985	1369	5.7	11.9
DPrompt^{N} (CVPR'24)	✓	327.0	2084	3908	1225	2524	406	774	4017	8844	272	501	6.1	13.1
Mari-DC (ICCV'25)	✓	1290	2187	3631	327	580	317	525	6594	12402	241	383	5.4	11.1
PriorDA (ICLR'26)	✓	195.1	1663	2952	367	588	271	477	1436	3462	217	352	4.3	8.4
TestDC (ICCV'25)	✓	363.2	1012	2915	**78**	466	142	458	996	3016	128	295	2.8	4.7
PacGDC (ICCV'25)	✓	254.4	**742**	2506	92	867	93	394	**807**	2660	73	217	2.2	3.0
Ours-DA2	✓	132.5	764	**2059**	96	**264**	**92**	**249**	876	2660	**70**	**158**	**1.6**	**1.7**

Table 2: Comparison on raw sparse depth patterns. Without pretraining, our MSP-Net even outperforms all end-to-end methods and pretrained methods. With pretraining, our MSP-Net further improves its performance and outperforms all methods.

Method	Pre	Para (M)	KITTI-64		NYU-Raw		VOID-150		VOID-500		VOID-1500		Rk	OA Rk
			MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE		
CFormer (CVPR'23)	–	83.5	2360	25758	730	2265	335	456	251	377	119	218	6.9	13.1
Stereo-DC (3DV'24)	–	11.1	548	1884	334	12841	120	237	81	188	49	138	5.5	9.9
OGNI-DC (ECCV'24)	–	84.4	679	2571	162	534	125	205	82	155	48	105	5.0	9.0
G2-Mono (TPAMI'24)	–	18.2	359	1588	**70**	340	123	251	72	175	40	116	3.7	6.7
SPNet (TPAMI'25)	–	35.0	346	1653	132	1122	74	187	48	143	31	102	2.9	6.1
OMNI-DC (ICCV'25)	–	85.5	**262**	**1223**	78	**236**	77	197	50	149	28	99	2.1	3.8
Ours	–	36.7	367	1602	106	349	**72**	**151**	**47**	**114**	**27**	**75**	**1.9**	3.6
PromptDA (CVPR'25)	✓	340.0	10984	15795	611	1086	640	786	668	812	690	831	6.8	13.3
DPrompt (CVPR'24)	✓	327.0	1825	5379	776	1097	273	408	179	311	82	172	6.1	12.0
Mari-DC (ICCV'25)	✓	1290	1262	2491	194	391	113	191	114	192	97	172	5.1	9.9
PriorDA (ICLR'26)	✓	195.1	492	**1459**	222	348	65	142	52	120	39	105	3.1	5.3
TestDC (ICCV'25)	✓	363.2	853	2275	**101**	**285**	**51**	157	**40**	124	31	101	2.3	4.4
PacGDC (ICCV'25)	✓	254.4	**371**	1471	108	387	65	185	45	165	29	114	3.1	5.3
Ours-DA2	✓	132.5	372	1630	102	320	59	**135**	43	**109**	**28**	**81**	**1.7**	**3.0**

(II) *Raw sparse depth patterns*. Raw sparse-depth inputs include three types: the official 64-line LiDAR sparse depth from KITTI-DC (Uhrig et al., 2017), the official raw depth from NYU-V2 (Silberman et al., 2012), and the three official sparse-depth settings (150 points, 500 points, and 1500 points) from VOID (Wong et al., 2020).

4.1.3 METRICS

We use mean absolute error (MAE) and root mean squared error (RMSE). MAE and RMSE are reported in millimeters.

4.1.4 BASELINES

We compare the proposed method with twelve representative depth completion approaches, including six end-to-end methods, CFormer (Zhang et al., 2023), Stereo-DC (Bartolomei et al., 2024), OGNI-DC (Zuo & Deng, 2024), G2-Mono (Wang et al., 2023a), SPNet (Wang et al., 2024), and OMNI-DC (Zuo et al., 2025), and six depth-foundation-model-based methods, PromptDA (Lin et al., 2025), DPrompt (Park et al., 2024), Mari-DC (Viola et al., 2025), PriorDA (Wang et al., 2025b), TestDC (Jeong et al., 2025), and PacGDC (Wang et al., 2025a). All baseline results are obtained using the officially released models.

4.1.5 IMPLEMENTATION DETAILS

We adopt ConvNeXt-V2-Tiny (Woo et al., 2023) as the backbone of the trainable depth completion network following SPNet (Wang et al., 2024). The network is initialized using Xavier normal initialization (Glorot & Bengio, 2010) and optimized with AdamW (Loshchilov & Hutter, 2017) using an initial learning rate of 3×10^{-4} and a weight decay of 0.05. The batch size is set to 65, and the model is trained for 300 epochs on four NVIDIA RTX 4090 GPUs, each with 24 GB memory, using a cosine learning-rate schedule.

4.1.6 PRETRAINING

We employ Depth Anything V2 (Yang et al., 2024b) to generate relative depth priors. The model is frozen throughout training and inference, and its prediction is converted from the inverse depth domain to the depth domain before being used by the completion network.

4.2 QUANTITATIVE RESULTS

4.2.1 SIMULATED SPARSE DEPTH PATTERNS

Table 1 presents the comparison with end-to-end and depth-foundation-model-based methods under simulated sparse depth patterns. Without incorporating a depth foundation model, our MSP-Net achieves the best rank of 1.7 among non-pretrained methods and an overall rank of 3.6. It consistently outperforms all competing end-to-end methods and most depth-foundation-model-based methods. After incorporating the relative depth prior from Depth Anything V2, Ours-DA2 further improves the rank to 1.6 and achieves the best overall rank of 1.7 among all compared methods. These results demonstrate that our MSP-Net effectively improves zero-shot generalization across diverse simulated sparse observation patterns.

4.2.2 RAW SPARSE DEPTH PATTERNS

Table 2 reports the comparison under raw sparse depth settings using official sensor observations from KITTI-DC with 64-line LiDAR, NYU-Depth V2 with raw depth measurements, and VOID with 150, 500, and 1500 sparse points. Without incorporating a depth foundation model, our MSP-Net achieves the best rank of 1.9 among non-pretrained methods and an overall rank of 3.6, outperforming all end-to-end baselines and all competing depth-foundation-model-based methods. By incorporating the Depth Anything V2 prior, Ours-DA2 further improves the rank to 1.7 and achieves the best overall rank of 3.0 among all methods. These results validate the effectiveness and robustness of our MSP-Net under diverse sensor configurations and raw sparse observation patterns.

4.2.3 ACCURACY–EFFICIENCY TRADE-OFF

Figure 3 compares RMSE and inference speed on iBims-1 and VOID. Without pretraining, Ours achieves the lowest RMSE and fastest inference on iBims-1. With pretraining, Ours-DA2 still achieves the best accuracy–efficiency trade-off on VOID.

4.3 QUALITATIVE RESULTS

Figure 1 has presented several qualitative results. Additional qualitative results can be found in supplementary material.

4.4 ABLATION STUDY

Main ablation In Table 3, we study the role of the two main modules, SIM and SPM, as well as DA2 pretraining. The ablation study is conducted under two representative settings: NYUv2 with

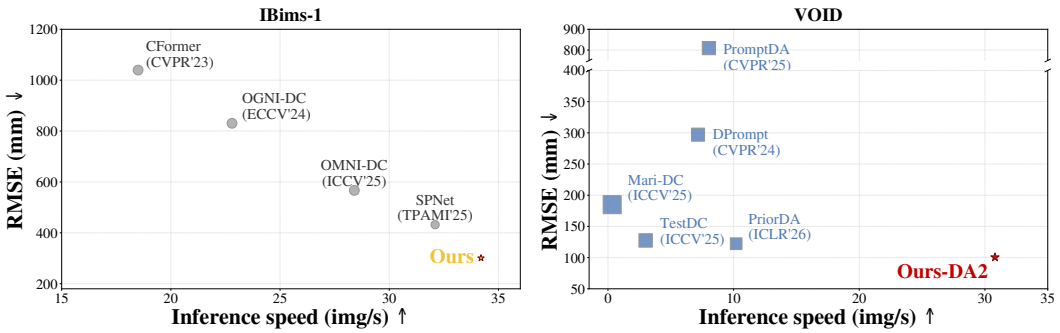


Figure 3: Accuracy–efficiency trade-off using RMSE averaged over three simulated patterns on iBims-1 (left, without pretraining) and three sparsity patterns on VOID (right, with pretraining). Ours and Ours-DA2 achieve the best trade-offs in their respective settings.

Hole/Range/Point sparse inputs and VOID with 150/500/1500 sparse samples. The baseline does not use our scale identification and propagation. Adding SIM improves the estimation by identifying reliable sparse depth observations and extracting scene-level scale cues. Adding SPM further propagates sparse scale cues to missing regions. Introducing DA2 provides additional guidance.

Table 3: Ablation on main modules.

Module			NYUv2			VOID		
SIM	SPM	DA2	AbsRel	MAE	RMSE	AbsRel	MAE	RMSE
			3.9	100.8	398.4	3.3	50.7	144.1
✓			3.7	96.8	201.3	3.2	51.1	117.8
✓	✓		3.4	90.6	194.7	3.0	48.4	113.0
✓	✓	✓	**2.6**	**69.9**	**158.1**	**2.6**	**43.4**	**108.5**

Ablation on SIM. In Table 4, we study the internal designs of SIM on the DIODE dataset. Removing the reliability map means that all valid sparse depth points are treated as equally reliable. Removing the global scale anchor disables the scene scale reference estimated from reliable sparse depth. Removing Scale Token Injection (STI) means scale information cannot be directly injected into hierarchical features. The full SIM achieves the best performance, demonstrating the effectiveness of all important components in SIM.

Table 4: Ablation on SIM components.

SIM Components			Metrics		
Reliability	Anchor	STI	AbsRel	MAE	RMSE
	✓	✓	19.1	1964	3512
✓		✓	15.5	1703	2634
✓	✓		15.7	1673	2578
✓	✓	✓	**14.1**	**921**	**2118**

Table 5: Ablation on SPM components.

SPM Components			Metrics		
Dense Prop	RGB Aff	Iter Prop	AbsRel	MAE	RMSE
	✓	✓	3.9	613	2420
✓		✓	3.6	661	1935
✓	✓		2.7	498	1795
✓	✓	✓	**2.5**	**367**	**1602**

Ablation on SPM. In Table 5, we study the internal designs of SPM on KITTI/64-line lidar. Without dense scale propagation, the model cannot spread reliable sparse scale cues to unobserved regions. Without RGB-guided affinity, scale propagation degenerates into structure-agnostic smoothing. Without iterative propagation, scale information could not reach larger missing regions. The full SPM leads to better performance.

5 Conclusion

We identify that existing depth completion methods struggle to preserve scene scale from sparse inputs to dense predictions. We propose MSP-Net to explicitly preserve scene scale throughout the end-to-end depth completion process. Our method outperforms both recent end-to-end methods and pretrained depth-foundation-model-based methods while achieving faster inference. One limitation is that MSP-Net with a pretrained depth foundation model shows limited performance under certain raw sparse depth patterns, such as KITTI 64-line LiDAR. Future work will focus on more effective strategies for incorporating pretrained depth foundation models to further improve cross-scene generalization.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish the writing of this paper, assist with English translation, check the consistency of reported tables, ranks, and textual descriptions, and refactor portions of the research code. We did not use generative AI to generate data, formulate hypotheses or mathematical claims, develop the core method, design experiments, or produce experimental results. All AI-assisted content was reviewed by the authors, and the refactored code was tested for correctness. The authors take full responsibility for the final content of this work, including its text, claims, code, and artifacts.

ETHICS STATEMENT

This work uses existing public datasets and does not involve the collection of new human-subject data. All datasets are used for research purposes in accordance with their respective licenses. As depth completion may be applied in safety-critical systems, real-world deployment should include appropriate validation and safety measures.

REPRODUCIBILITY STATEMENT

The main paper describes the proposed architecture, training objectives, datasets, evaluation protocols, and implementation settings. The appendix provides detailed dataset statistics, complete quantitative results, additional predictions, and the implementation of the pretrained depth prior. These materials are intended to support reproduction of the reported experiments.

REFERENCES

Luca Bartolomei, Matteo Poggi, Andrea Conti, Fabio Tosi, and Stefano Mattoccia. Revisiting depth completion from a stereo matching perspective for cross-domain generalization. In *2024 International Conference on 3D Vision (3DV)*, pp. 1360–1370. IEEE, 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.

Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4340–4349, 2016.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256. JMLR Workshop and Conference Proceedings, 2010.

Mu Hu, Shuling Wang, Bin Li, Shiyu Ning, Li Fan, and Xiaojin Gong. Penet: Towards precise and efficient image guided depth completion. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13656–13662. IEEE, 2021.

Chanhwi Jeong, Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Test-time prompt tuning for zero-shot depth completion. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9443–9454. IEEE, 2025.

Bingxin Ke, Anton Obukhov, Shengyu Huang, Nicolas Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Korner. Evaluation of cnn-based single-image depth estimation methods. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 0–0, 2018.

Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 1477–1485, 2023.

Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2041–2050, 2018.

Yingping Liang, Yutao Hu, Wenqi Shao, and Ying Fu. Distilling monocular foundation model for fine-grained depth completion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22254–22265, 2025.

Haotong Lin, Sida Peng, Jingxiao Chen, Songyou Peng, Jiaming Sun, Minghuan Liu, Hujun Bao, Jiashi Feng, Xiaowei Zhou, and Bingyi Kang. Prompting depth anything for 4k resolution accurate metric depth estimation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17070–17080, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *IEEE International Conference on Robotics and Automation*, 2018.

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

Jin-Hwi Park, Chanhwi Jeong, Junoh Lee, and Hae-Gon Jeon. Depth prompting for sensor-agnostic depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9859–9869, 2024.

Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kai Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *European Conference on Computer Vision*, 2020a.

Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *European conference on computer vision*, pp. 120–136. Springer, 2020b.

Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, 2020.

Weichao Qiu and Alan Yuille. Unrealcv: Connecting computer vision to unreal engine. In *European conference on computer vision*, pp. 909–916. Springer, 2016.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020.

Thomas Schops, Johannes L Schonberger, 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3260–3269, 2017.

Shreyas S Shivakumar, Ty Nguyen, Ian D Miller, Steven W Chen, Vijay Kumar, and Camillo J Taylor. Dfusenet: Deep fusion of rgb and sparse depth information for image guided dense depth completion. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 13–20. IEEE, 2019.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pp. 746–760. Springer, 2012.

Jie Tang, Fei-Peng Tian, Wei Feng, Jian Li, and Ping Tan. Learning guided convolutional network for depth completion. *IEEE Transactions on Image Processing*, 30:1116–1129, 2020.

Jie Tang, Fei-Peng Tian, Boshi An, Jian Li, and Ping Tan. Bilateral propagation network for depth completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9763–9772, 2024.

Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *2017 international conference on 3D Vision (3DV)*, pp. 11–20. IEEE, 2017.

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

Massimiliano Viola, Kevin Qu, Nando Metzger, Bingxin Ke, Alexander Becker, Konrad Schindler, and Anton Obukhov. Marigold-dc: Zero-shot monocular depth completion with guided diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5359–5370, 2025.

Haotian Wang, Meng Yang, and Nanning Zheng. G2-monodepth: A general framework of generalized depth inference from monocular rgb+ x data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3753–3771, 2023a.

Haotian Wang, Meng Yang, Xinhu Zheng, and Gang Hua. Scale propagation network for generalizable depth completion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3): 1908–1922, 2024.

Haotian Wang, Aoran Xiao, Xiaoqin Zhang, Meng Yang, and Shijian Lu. Pacgdc: Label-efficient generalizable depth completion with projection ambiguity and consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7709–7720, 2025a.

Yufei Wang, Bo Li, Ge Zhang, Qi Liu, Tao Gao, and Yuchao Dai. Lrru: Long-short range recurrent updating networks for depth completion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9422–9432, 2023b.

Zehan Wang, Siyu Chen, Lihe Yang, Jialei Wang, Ziang Zhang, Hengshuang Zhao, and Zhou Zhao. Depth anything with any prior. *arXiv preprint arXiv:2505.10565*, 2025b.

Jamie Watson, Mohamed Sayed, Zawar Qureshi, Gabriel J Brostow, Sara Vicente, Oisin Mac Aodha, and Michael Firman. Virtual occlusions through implicit depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9053–9064, 2023.

Alex Wong, Xiaohan Fei, Stephanie Tsuei, and Stefano Soatto. Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters*, 5(2):1899–1906, 2020.

Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16133–16142, 2023.

Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. Structure-guided ranking loss for single image depth prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 611–620, 2020.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024a.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024b.

Youmin Zhang, Xianda Guo, Matteo Poggi, Zheng Zhu, Guan Huang, and Stefano Mattoccia. Completionformer: Depth completion with convolutions and vision transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12786–12796, 2022.

Yiming Zuo and Jia Deng. Ogni-dc: Robust depth completion with optimization-guided neural iterations. In *European Conference on Computer Vision*, pp. 78–95. Springer, 2024.

Yiming Zuo, Willow Yang, Zeyu Ma, and Jia Deng. Omni-dc: Highly robust depth completion with multiresolution depth integration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9287–9297, 2025.

SUPPLEMENTARY MATERIAL

This appendix provides additional details and results. It is organized into four sections. Section A presents qualitative visualizations under both simulated sparse depth input and raw sparse depth input. Section B provides detailed results on specific patterns of simulated sparse depth input. Section C provides the details of the training and testing datasets. Section D provides the implementation of the pretrained depth foundation model.

A QUALITATIVE VISUALIZATIONS

We further provide qualitative visualizations to present the generalization ability of our method under both simulated and raw sparse depth inputs.

For simulated sparse depth inputs, Figure 4 presents qualitative comparisons under three sparse depth patterns: *Hole*, *Range*, and *Point*. Some baseline methods struggle to generalize across different sparse depth patterns in a zero-shot setting. For example, PriorDA, OGNI-DC, and DepthPrompt exhibit inconsistent performance when handling different types of sparse inputs. This results in degraded geometric structures or inaccurate depth boundaries under certain patterns. Although some baseline methods can adapt to all three sparse depth patterns, they still suffer from blurred object boundaries, distorted geometric structures, and missing fine details. In contrast, our method consistently preserves global metric scale while recovering accurate scene geometry, sharp depth discontinuities, and detailed local structures across all simulated sparse depth settings.

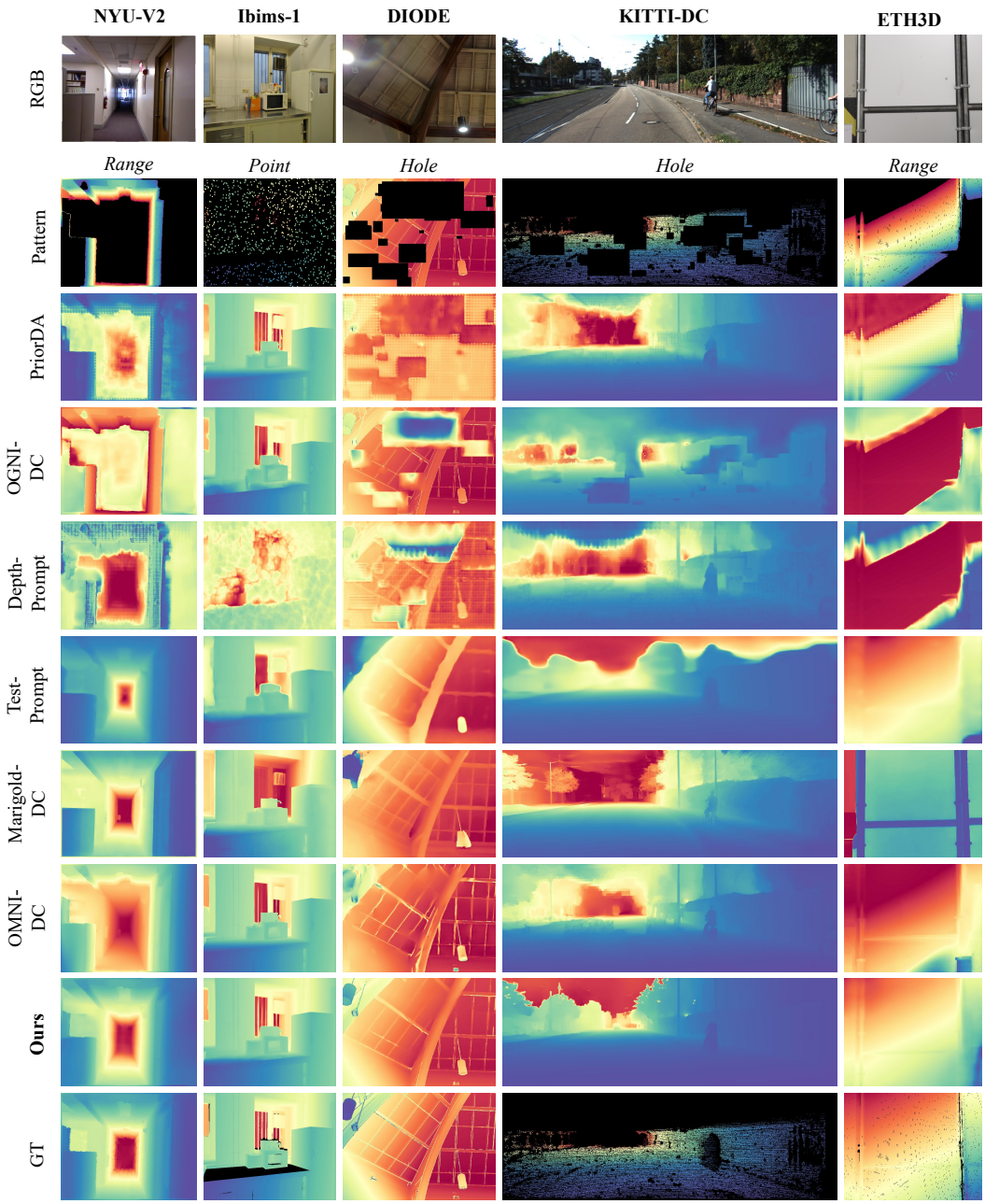


Figure 4: Qualitative comparisons on simulated sparse depth inputs.

For raw sparse depth inputs, Figure 5 presents qualitative comparisons under raw depth measurements, including VOID sparse points, KITTI-DC 64-line LiDAR, and NYU-V2 raw depth. Some baseline methods fail to generalize to real sensor inputs in a zero-shot setting. For example, CFormer and DepthPrompt produce degraded predictions when directly applied to raw sparse depth observations due to the different measurement characteristics and sparse distributions. Although other baseline methods can handle raw sparse inputs to some extent, they still exhibit inaccurate geometric reconstruction, over-smoothed structures, and insufficient preservation of fine boundaries and textures. Our method produces more accurate and visually consistent depth predictions under different types of raw sparse depth inputs.

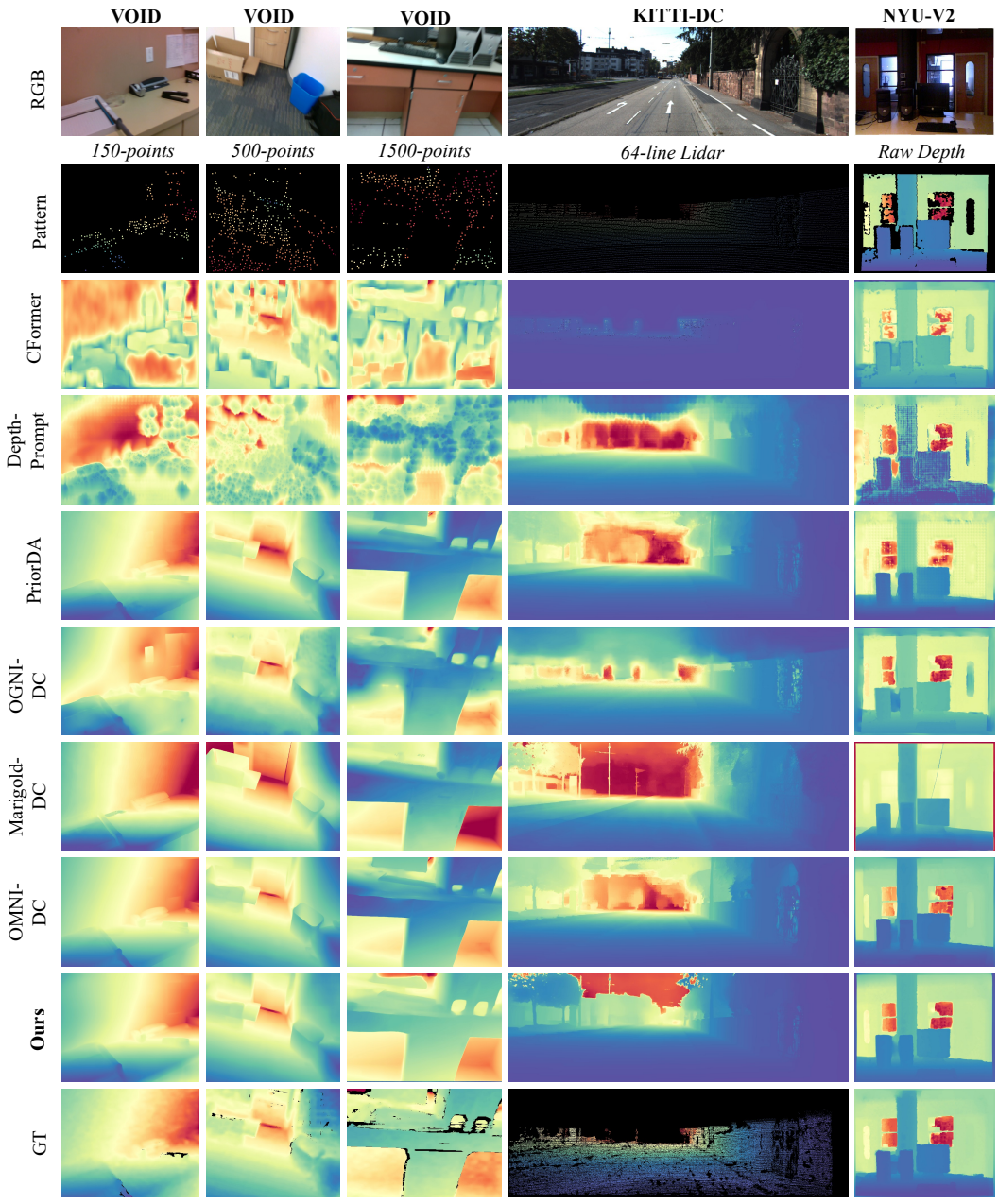


Figure 5: Qualitative comparisons on raw sparse depth inputs.

B Detailed Results on Simulated Sparse Depth

We further provide complete results on five datasets with three different sparse patterns, including DIODE, ETH3D, iBims-1, KITTI-DC, and NYU-V2. We add AbsRel as an additional evaluation metric. AbsRel is reported in percentage. All methods are grouped according to whether they leverage pretrained foundation depth priors. "Rk" refers to the rank in each setting (without/with pretraining). "Overall Rank" refers to the rank among all methods. The best and second-best results are in **bold** and underlined, respectively.

Table 6 presents the detailed results on DIODE. Without pretraining, our method achieves the second-best rank and the third-best overall rank among all methods. With pretraining, Ours-DA2 achieves the best rank and the best overall rank among all methods.

Table 6: Detailed results on the DIODE dataset with simulated sparse depth.

Method	Pre	Para (M)	*Hole* AbsRel	*Hole* MAE	*Hole* RMSE	*Range* AbsRel	*Range* MAE	*Range* RMSE	*Point* AbsRel	*Point* MAE	*Point* RMSE	Rk	OA Rk
CFormer^K (CVPR'23)	–	83.5	18.6	1328	2954	40.4	2483	4596	50.1	1729	3114	7.8	13.2
CFormer^N (CVPR'23)	–	83.5	42.0	3522	8453	46.8	3879	5872	14.9	944	3763	8.2	14.8
Stereo-DC (3DV'24)	–	11.1	25.1	1636	3366	29.6	2164	4083	26.2	1121	2638	6.9	11.4
OGNI-DC^K (ECCV'24)	–	84.4	11.9	1034	2353	33.0	2181	4033	18.3	947	2181	5.2	8.9
OGNI-DC^N (ECCV'24)	–	84.4	13.0	1158	2701	27.6	2183	4186	13.9	838	1914	5.1	8.4
G2-Mono (TPAMI'24)	–	18.2	8.9	473	1812	25.2	1567	4113	15.2	688	2426	3.8	6.9
SPNet (TPAMI'25)	–	35.0	**7.6**	**416**	1727	**20.3**	**1392**	3993	11.7	600	2465	**2.0**	4.6
OMNI-DC (ICCV'25)	–	85.5	11.9	1072	3223	20.8	1643	4387	**10.8**	**549**	2129	3.8	6.6
Ours	–	36.7	7.8	470	**1254**	21.9	1588	**3269**	12.7	703	**1830**	2.2	4.1
PromptDA (CVPR'25)	✓	340.0	48.0	4767	6900	30.9	3022	5106	71.7	7391	8941	7.7	16.0
DPrompt^K (CVPR'24)	✓	327.0	29.4	2007	3102	36.3	2553	3962	31.6	1733	2942	5.9	12.8
DPrompt^N (CVPR'24)	✓	327.0	24.0	2183	4185	33.8	2872	4977	15.5	1197	2562	5.9	13.2
Mari-DC (ICCV'25)	✓	1290.0	47.3	2589	4148	32.3	2144	3660	34.3	1828	3084	5.9	12.6
PriorDA (ICLR'26)	✓	195.1	15.7	1555	2771	22.4	2176	3834	13.3	1259	2250	3.9	8.6
TestDC (ICCV'25)	✓	363.2	13.8	752	2263	21.6	1496	4042	14.2	787	2441	3.6	7.0
PacGDC (ICCV'25)	✓	254.4	7.1	405	1558	**17.0**	**1272**	3861	**10.6**	**548**	2097	1.8	2.2
Ours-DA2	✓	132.5	**7.0**	**397**	**1211**	18.1	1314	**3140**	11.1	580	**1826**	**1.4**	**1.7**

Table 7 presents the detailed results on ETH3D. Without pretraining, our method achieves the second-best rank and the fourth-best overall rank among all methods. With pretraining, Ours-DA2 achieves the best rank and the best overall rank among all methods.

Table 7: Detailed results on the ETH3D dataset with simulated sparse depth.

Method	Pre	Para (M)	*Hole* AbsRel	*Hole* MAE	*Hole* RMSE	*Range* AbsRel	*Range* MAE	*Range* RMSE	*Point* AbsRel	*Point* MAE	*Point* RMSE	Rk	OA Rk
CFormer^K (CVPR'23)	–	83.5	2.3	155	568	25.0	1256	2834	47.7	1106	1730	7.4	13.2
CFormer^N (CVPR'23)	–	83.5	31.6	2678	3393	36.2	3420	4346	1.1	248	1098	8.3	15.2
Stereo-DC (3DV'24)	–	11.1	19.5	2237	4297	14.8	876	2075	10.6	418	846	7.3	13.7
OGNI-DC^K (ECCV'24)	–	84.4	1.0	65	282	11.8	735	1784	0.7	57	218	5.0	8.6
OGNI-DC^N (ECCV'24)	–	84.4	2.6	205	667	17.3	1657	3175	0.5	84	300	6.6	11.2
G2-Mono (TPAMI'24)	–	18.2	**0.1**	**18**	116	2.5	**257**	1455	0.3	28	158	**1.8**	3.6
SPNet (TPAMI'25)	–	35.0	0.5	41	126	**2.2**	304	1586	0.6	51	301	3.4	6.4
OMNI-DC (ICCV'25)	–	85.5	0.6	43	257	3.2	311	1514	**0.1**	**10**	136	2.8	5.0
Ours	–	36.7	0.3	39	**75**	3.6	331	**882**	0.4	45	**110**	2.3	4.4
PromptDA (CVPR'25)	✓	340.0	12.0	1424	2268	11.1	1318	2216	25.3	2927	3565	7.2	14.3
DPrompt^K (CVPR'24)	✓	327.0	21.0	1354	1552	26.4	1934	2650	7.2	548	827	7.0	14.2
DPrompt^N (CVPR'24)	✓	327.0	6.6	741	1904	24.1	2514	4446	3.2	422	1223	6.8	14.3
Mari-DC (ICCV'25)	✓	1290.0	2.3	193	330	6.2	588	1064	2.4	198	346	4.2	9.2
PriorDA (ICLR'26)	✓	195.1	3.3	421	639	5.7	610	978	0.5	70	148	3.9	8.3
TestDC (ICCV'25)	✓	363.2	0.3	28	124	1.9	**167**	1039	0.5	39	234	2.4	4.1
PacGDC (ICCV'25)	✓	254.4	**0.1**	28	497	**1.6**	191	1350	**0.2**	58	754	2.8	4.9
Ours-DA2	✓	132.5	0.2	**24**	**49**	2.5	234	**662**	**0.2**	**30**	**81**	**1.6**	**2.3**

Table 8 presents the detailed results on iBims-1. Without pretraining, our method achieves the best rank and the third-best overall rank among all methods. With pretraining, Ours-DA2 achieves the best rank and the best overall rank among all methods.

Table 8: Detailed results on the iBims-1 dataset with simulated sparse depth.

Method	Pre	Para (M)	*Hole* AbsRel	*Hole* MAE	*Hole* RMSE	*Range* AbsRel	*Range* MAE	*Range* RMSE	*Point* AbsRel	*Point* MAE	*Point* RMSE	Rk	OA Rk
CFormer^K (CVPR'23)	–	83.5	35.9	885	2250	101.3	2242	4033	141.2	2443	3311	8.9	16.6
CFormer^N (CVPR'23)	–	83.5	33.8	1116	1352	37.0	1275	1561	1.4	57	205	6.7	12.1
Stereo-DC (3DV'24)	–	11.1	3.1	131	307	15.4	543	1246	2.8	120	634	5.8	9.8
OGNI-DC^K (ECCV'24)	–	84.4	7.0	233	639	31.4	777	1424	3.4	114	302	6.7	11.6
OGNI-DC^N (ECCV'24)	–	84.4	11.3	443	906	20.5	788	1395	1.6	62	191	5.9	10.9
G2-Mono (TPAMI'24)	–	18.2	1.3	46	226	10.4	318	860	1.9	61	290	3.9	6.8
SPNet (TPAMI'25)	–	35.0	**0.9**	**32**	206	**8.3**	**262**	820	1.1	44	270	1.9	4.4
OMNI-DC (ICCV'25)	–	85.5	2.8	98	395	11.6	406	1065	**1.0**	**37**	240	3.4	5.9
Ours	–	36.7	**0.9**	34	**130**	8.5	275	**593**	1.2	45	**180**	**1.8**	3.8
PromptDA (CVPR'25)	✓	340.0	29.5	1142	1708	22.5	865	1387	47.2	1865	2258	7.6	14.9
DPrompt^K (CVPR'24)	✓	327.0	43.1	906	1380	60.8	1330	1864	37.9	778	1023	7.4	15.6
DPrompt^N (CVPR'24)	✓	327.0	10.2	412	837	17.5	678	1168	3.1	129	319	5.6	11.6
Mari-DC (ICCV'25)	✓	1290.0	7.4	258	445	16.4	512	790	4.6	180	339	4.9	10.4
PriorDA (ICLR'26)	✓	195.1	6.7	270	451	12.7	489	821	1.4	54	**159**	3.8	7.4
TestDC (ICCV'25)	✓	363.2	2.5	88	321	7.4	238	689	2.8	99	364	3.4	6.8
PacGDC (ICCV'25)	✓	254.4	0.8	30	201	**6.0**	210	739	**1.0**	**39**	242	1.9	2.7
Ours-DA2	✓	132.5	**0.7**	**27**	**105**	6.1	**207**	**473**	1.1	40	168	**1.4**	**1.7**

Table 9 presents the detailed results on KITTI-DC. Without pretraining, our method achieves the best rank and the second-best overall rank among all methods. With pretraining, Ours-DA2 achieves the second-best rank and the third-best overall rank among all methods.

Table 9: Detailed results on the KITTI-DC dataset with simulated sparse depth.

Method	Pre	Para (M)	*Hole* AbsRel	*Hole* MAE	*Hole* RMSE	*Range* AbsRel	*Range* MAE	*Range* RMSE	*Point* AbsRel	*Point* MAE	*Point* RMSE	Rk	OA Rk
CFormer^K (CVPR'23)	–	83.5	6.9	1311	3190	16.8	3432	7500	4.8	790	2527	6.8	11.4
CFormer^N (CVPR'23)	–	83.5	19.4	4413	26474	37.7	7457	14364	6.2	1770	17482	9.0	15.7
Stereo-DC (3DV'24)	–	11.1	2.8	632	2046	13.6	3346	8093	2.7	583	2098	5.3	9.0
OGNI-DC^K (ECCV'24)	–	84.4	2.9	617	2446	14.3	2437	5674	2.4	488	1731	4.3	7.6
OGNI-DC^N (ECCV'24)	–	84.4	8.1	1536	4531	25.3	5578	11831	4.4	1093	3436	7.9	12.9
G2-Mono (TPAMI'24)	–	18.2	1.6	334	1588	10.8	2418	6297	2.4	486	1929	2.9	5.7
SPNet (TPAMI'25)	–	35.0	1.6	361	2119	**6.9**	**1660**	5792	2.4	554	2783	3.5	6.4
OMNI-DC (ICCV'25)	–	85.5	2.3	530	1917	11.0	2976	8289	**1.7**	**376**	**1571**	3.3	5.6
Ours	–	36.7	**1.4**	**324**	**1460**	8.1	2074	**5543**	2.1	477	2122	**1.9**	4.1
PromptDA (CVPR'25)	✓	340.0	52.3	9989	14773	38.8	8005	13421	55.9	10718	15676	7.7	16.3
DPrompt^K (CVPR'24)	✓	327.0	7.0	1310	2302	8.2	1529	**2801**	3.3	642	1688	3.6	7.2
DPrompt^N (CVPR'24)	✓	327.0	14.3	2772	6999	34.4	6954	13510	10.7	2324	6024	6.4	14.7
Mari-DC (ICCV'25)	✓	1290.0	8.1	3384	7113	29.0	13036	23039	8.2	3364	7054	6.9	15.3
PriorDA (ICLR'26)	✓	195.1	2.6	604	1710	15.0	3186	6850	2.3	516	1826	3.7	7.3
TestDC (ICCV'25)	✓	363.2	3.5	700	2037	6.2	**1395**	4176	4.2	893	2835	3.6	7.4
PacGDC (ICCV'25)	✓	254.4	**1.3**	**268**	**1160**	**5.9**	1700	5196	**2.2**	**453**	**1624**	**1.6**	**2.2**
Ours-DA2	✓	132.5	1.5	312	1291	7.8	1818	4534	**2.2**	498	2153	2.6	4.2

Table 10 presents the detailed results on NYU Depth V2. Without pretraining, our method achieves the best rank and the third-best overall rank among all methods. With pretraining, Ours-DA2 achieves the best rank and the best overall rank among all methods.

Table 10: Detailed results on the NYU-V2 dataset with simulated sparse depth.

Method	Pre	Para (M)	*Hole* AbsRel	*Hole* MAE	*Hole* RMSE	*Range* AbsRel	*Range* MAE	*Range* RMSE	*Point* AbsRel	*Point* MAE	*Point* RMSE	Rk	OA Rk
CFormer^K (CVPR'23)	–	83.5	36.6	804	2061	133.4	2637	4731	130.5	2065	2801	8.8	16.4
CFormer^N (CVPR'23)	–	83.5	33.7	919	1089	37.1	1024	1210	2.2	59	228	7.3	12.9
Stereo-DC (3DV'24)	–	11.1	3.6	106	465	16.2	388	3378	8.2	212	9549	6.6	11.9
OGNI-DC^K (ECCV'24)	–	84.4	4.6	124	316	19.1	432	798	3.6	91	250	6.0	10.5
OGNI-DC^N (ECCV'24)	–	84.4	11.4	365	742	19.5	634	1075	1.8	50	154	5.9	10.6
G2-Mono (TPAMI'24)	–	18.2	1.5	40	161	11.8	274	752	2.2	53	146	3.5	6.1
SPNet (TPAMI'25)	–	35.0	**1.0**	**29**	153	9.2	234	843	1.4	40	198	2.8	5.7
OMNI-DC (ICCV'25)	–	85.5	3.4	89	256	12.7	325	709	**1.2**	**34**	121	2.9	5.6
Ours	–	36.7	**1.0**	**29**	**89**	**7.8**	**208**	**396**	**1.2**	35	**100**	**1.3**	2.7
PromptDA (CVPR'25)	✓	340.0	27.2	862	1294	24.0	734	1136	49.6	1573	1861	7.3	14.7
DPrompt^K (CVPR'24)	✓	327.0	45.4	913	1348	61.6	1335	1843	33.0	708	917	7.7	15.6
DPrompt^N (CVPR'24)	✓	327.0	9.2	274	550	15.2	455	762	3.0	88	190	5.1	10.3
Mari-DC (ICCV'25)	✓	1290.0	5.6	148	263	17.5	433	625	5.4	142	261	5.1	10.6
PriorDA (ICLR'26)	✓	195.1	7.0	219	342	12.4	355	522	2.6	78	192	4.1	8.7
TestDC (ICCV'25)	✓	363.2	3.1	84	190	8.5	210	475	3.4	91	220	3.7	6.9
PacGDC (ICCV'25)	✓	254.4	**0.9**	**26**	115	6.0	158	417	**1.2**	**35**	120	1.7	2.3
Ours-DA2	✓	132.5	**0.9**	27	**79**	**5.7**	**147**	**296**	**1.2**	**35**	**99**	**1.3**	**1.6**

C Train and Test Datasets

Table 11 presents the training datasets used in our experiments. We use four datasets for training, including HRWSI, Matterport3D, UnrealCV, and Virtual KITTI. These datasets contain a total of 235,612 samples and cover diverse indoor and outdoor scenes collected with various sensing modalities, including structured light, multiview, and synthetic sensors.

Table 11 also presents the testing datasets used for evaluation. We evaluate on six datasets, including DIODE, ETH3D, iBims-1, KITTI-DC, NYU-Depth V2, and VOID. These datasets cover indoor, outdoor, and mixed indoor-outdoor environments with different sensing modalities, including LiDAR, multiview, structured light, and active stereo. In addition, we provide the valid depth ranges of each testing dataset (in meters), computed from valid depth pixels.

Table 11: Training and testing datasets used in our method.

Training datasets			
Dataset	**Scene Type**	**Samples**	**Sensor Type**
HRWSI	Indoor+Outdoor	19,113	Structured light
Matterport3D	Indoor	168,980	Multiview
UnrealCV	Indoor	4,999	Synthetic
Virtual KITTI	Outdoor	42,520	Synthetic

Testing datasets				
Dataset	**Scene Type**	**Samples**	**Depth Range (m)**	**Sensor Type**
DIODE	Indoor+Outdoor	771	0.2–65.5	LiDAR
ETH3D	Indoor+Outdoor	454	0.7–68.9	Multiview
iBims-1	Indoor	100	0.6–34.7	Structured light
KITTI-DC	Outdoor	1,000	2.8–85.5	LiDAR
NYU-Depth V2	Indoor	654	0.7–9.9	Structured light
VOID	Indoor+Outdoor	2,400	0.2–5.0	Active stereo

D Implementation of Pretrained Depth Foundation Model

We optionally incorporate priors from a pretrained depth foundation model. The implementation is described below.

First, given an RGB image, we feed it into the frozen Depth Anything V2 model with a ViT-Base backbone to obtain a raw inverse relative depth map, where only positive values are retained.

Second, we convert the raw inverse relative depth map into a relative depth map. The valid values are first transformed into the logarithmic domain and centered using their median value, followed by an exponential transformation to reverse the inverse relationship between the raw inverse relative depth and the relative depth.

Third, we perform robust percentile normalization on the relative depth map. The 1st and 99th percentile values are used to normalize the valid relative depth values, reducing the influence of extreme predictions. The normalized relative depth is then clipped to the range of $[0, 1]$.

Fourth, invalid pixels are assigned zero values to obtain the final normalized relative depth prior. Algorithm 1 summarizes the complete generation procedure.

During both training and testing, the relative depth prior is jointly used with RGB images and sparse metric depth inputs to predict the residual depth map. The pretrained Depth Anything V2 model remains frozen.

This implementation is applicable to any depth foundation model. It allows our framework to potentially incorporate other depth foundation models for further improvement.

Algorithm 1 Implementation of the pretrained depth foundation model.

Require: RGB image I, frozen Depth Anything V2 model F

Step 1: Generate raw inverse relative depth map

$r \leftarrow \mathrm{float32}(F(I))$

$\Omega \leftarrow \{x \mid r(x) \text{ is finite}, \ r(x) > 0\}$

Step 2: Convert inverse relative depth to relative depth

$z(x) \leftarrow \log(r(x) + \epsilon)$ for $x \in \Omega$

$z(x) \leftarrow z(x) - \mathrm{median}(z_\Omega)$ for $x \in \Omega$

$d(x) \leftarrow \exp(-z(x))$ for $x \in \Omega$

Step 3: Normalize relative depth map

$l, h \leftarrow \mathrm{percentile}(d_\Omega, \{1, 99\})$

$\hat{d}(x) \leftarrow \mathrm{clip}((d(x) - l)/(h - l), 0, 1)$ for $x \in \Omega$

Step 4: Handle invalid pixels

$\hat{d}(x) \leftarrow 0$ for $x \notin \Omega$

return $\hat{d}$