The Midas Touch for Metric Depth

Yu Ma[1] Zizhan Guo[1] Zuyi Xiong[1] Haoran Zhang[1]
Yi Feng[1] Hongbo Zhao[1] Hanli Wang[1,2] Rui Fan[1,2,3]✉

[1]College of Electronic and Information Engineering, Tongji University
[2]Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University
[3]National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, Xi'an Jiaotong University

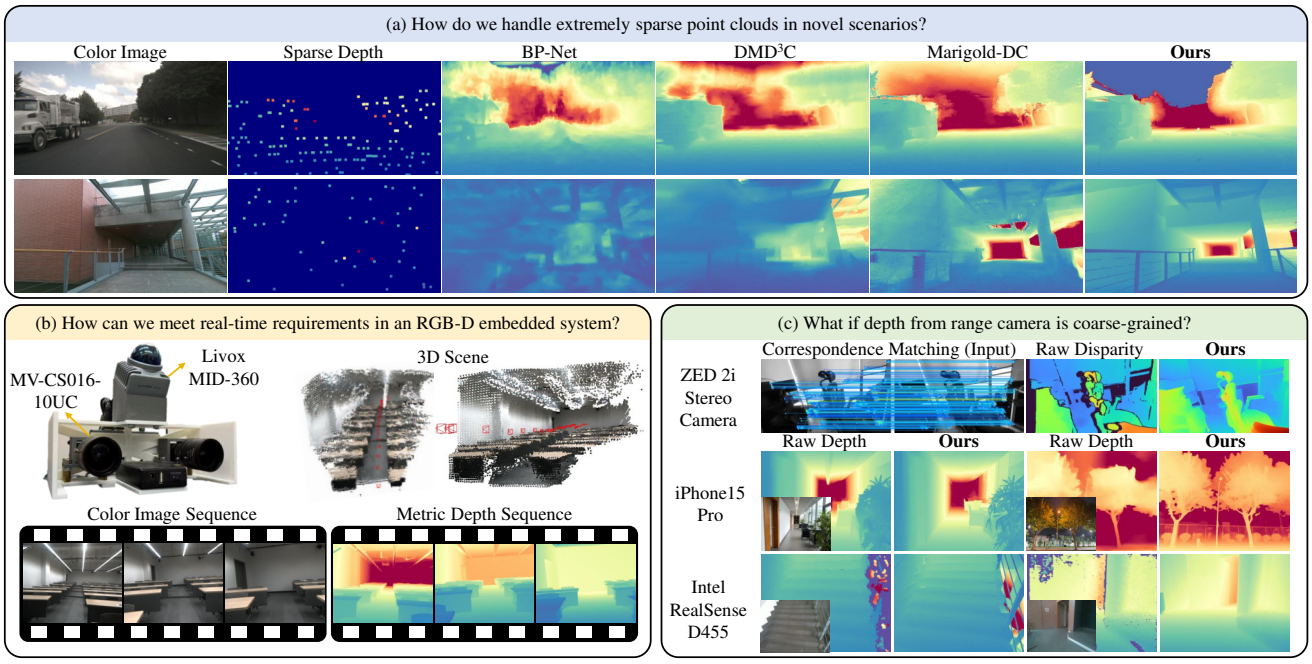

Figure 1. Application versatility of MTD in metric depth perception. (a) For novel scenes with extremely sparse point clouds, our method achieves precise depth completion and outperforms existing state-of-the-art methods. (b) To eliminate offline LiDAR point cloud aggregation, our method achieves real-time, online predictions on handheld edge devices, thanks to its low inference time and high accuracy. (c) For commonly used range cameras, our method can also serve as a plug-and-play module to enhance the quality of low-cost depth data.

Abstract

*Recent advances have markedly improved the cross-scene generalization of relative depth estimation, yet its practical applicability remains limited by the absence of metric scale, local inconsistencies, and low computational efficiency. To address these issues, we present **M**idas **T**ouch for **D**epth (MTD), a mathematically interpretable approach that converts relative depth into metric depth using only extremely sparse 3D data. To eliminate local scale inconsistencies, it applies a segment-wise recovery strategy via sparse graph optimization, followed by a pixel-wise refinement strategy using a discontinuity-aware geodesic cost. MTD exhibits strong generalization and achieves substantial accuracy improvements over previous depth completion and depth estimation methods. Moreover, its lightweight, plug-and-play design facilitates deployment and integration on diverse downstream 3D tasks. Project page is available at* https://mias.group/MTD.

1. Introduction

"In Greek mythology, King Midas's ability to turn everything into gold is known as the 'Midas touch'. Gold from a touch; meters from a hint. With extremely sparse 3D cues, relative depth crystallizes into metric measurement."

In recent years, monocular depth perception has emerged as a pivotal research focus in digital entertainment, computational photography, and 3D modeling [10, 50]. Driven

✉Corresponding author.

by the growing demand for strong generalization capabilities, including zero-shot performance, numerous efforts, such as the MiDaS series [2, 30] and the DepthAnything series [49, 50], have focused on developing depth foundation models for relative depth estimation. Despite significant advancements in generalization ability, their practical applicability remains limited. This limitation arises not only from their large parameter counts and inference latency, but more fundamentally from the inherent scale ambiguity [52], which hinders accurate metric depth prediction.

To resolve this scale ambiguity problem, a straightforward approach in previous studies [15, 19, 30, 49] leverages 3D point clouds to perform least-squares optimization for global scale recovery. Nevertheless, this solution often suffers from limited precision, as local regions may exhibit scale inconsistencies: different instances or segments exhibit distinct scale ratios and shift biases utilized for recovering metric depth. Therefore, a single global rescaling cannot accommodate these variations, thereby degrading metric depth accuracy and downstream task performance. Another line of research investigates incorporating captured 3D data into the metric depth estimation framework via networks [23, 24, 38]. However, these approaches are typically trained in domain-specific environments, which limits their ability to generalize. Addressing this issue requires large-scale training datasets, which in turn significantly increase the cost and complexity of data collection.

Motivated by these limitations, we introduce the ***Midas Touch for Depth*** (MTD), a universal paradigm that leverages available 3D data to efficiently and accurately convert relative depth to metric depth. We avoid fine-tuning the depth foundation models, thereby preventing potential performance degradation. To maintain interpretability and improve efficiency, we develop a reliable nonparametric method with a clear mathematical foundation. Specifically, we adopt a parallel, segment-wise strategy that optimizes a sparse segment graph to correct local scale inconsistencies. To further compensate for pixel-level errors, we perform a pixel-wise refinement by reformulating depth propagation as a discontinuity-aware geodesic problem, thereby enabling an efficient dynamic-programming solution.

To validate the effectiveness and real-world applicability of our method, we conduct large-scale experiments across indoor and outdoor depth perception benchmarks. Our method achieves strong accuracy and generalization, outperforming state-of-the-art (SoTA) approaches, as shown in Fig. 1. The results show that even under conditions of extremely sparse 3D data, our algorithm remains stable. To underscore the challenges posed by highly sparse, multi-source inputs (*e.g.*, depth, disparity, or correspondence matches), we refer to them metaphorically as ***3D seeds***, a term used throughout the paper. Moreover, our approach still recovers accurate metric depth, even when the relative depth predictions are only of moderate quality due to the reductions in model size. This advancement opens up new possibilities for improved computational efficiency. Our contributions are summarized as follows:

1. An efficient, effective, and universal paradigm for converting relative depth to metric depth, grounded in interpretable mathematical foundations;
2. A segment-wise scale recovery strategy via sparse graph optimization, complemented by a pixel-wise refinement strategy that uses a discontinuity-aware geodesic cost;
3. A SoTA, highly practical method which can handle different types of extremely sparse 3D seeds and support various downstream 3D tasks.

2. Related Work

2.1. Depth prediction

Recent studies [10, 50] build depth foundation models on large, diverse datasets, achieving strong cross-dataset generalization. Representative discriminative approaches include MiDaS [2, 30] and DepthAnything [49, 50]. To leverage data with varying scales, these methods typically adopt inverse-depth representations and affine-invariant losses, yielding relative depth predictions. In parallel, generative approaches [9, 15, 19, 37] repurpose diffusion models to infer dense relative depth, often improving boundary fidelity at higher inference cost. Despite strong generalization and visual quality, both categories lack absolute scale, limiting applications that require metric accuracy or spatiotemporal consistency. A complementary line of work studies metric monocular depth estimation [52], including Metric3D [16, 52] and UniDepth [28, 29]. However, these models still face challenges in cross-domain generalization and exhibit edge-fattening issues at object boundaries compared with SoTA relative depth estimators. We instead pursue a fast, interpretable scale recovery paradigm that preserves the generalization strength of relative depth models while yielding metric depth.

2.2. Depth Completion

Depth completion fuses the color image with sparse range measurements (*e.g.*, LiDAR) to recover dense metric depth [44, 56]. Earlier propagation-based methods [25, 27] depend on learned affinities via additional training, making their behavior less interpretable and potentially more sensitive to domain shifts. Recent SoTA methods, such as BP-Net [38] and DMD^{3}C [23], achieve strong results but tend to optimize accuracy gains on specific datasets at the expense of cross-domain robustness. Recent studies [41, 57] have trended toward stronger generalization. OGNI-DC [57] introduces optimization-guided neural iterations to improve generalization. Marigold-DC [41] leverages the progressive denoising of diffusion models to refine depth comple-

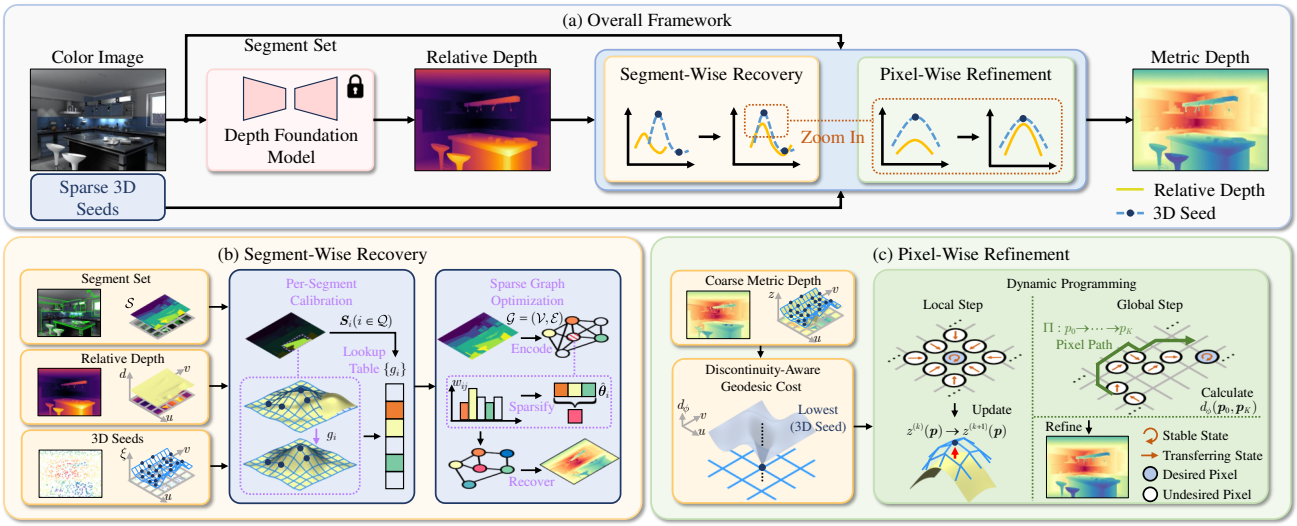

Figure 2. MTD takes relative depth, sparse 3D seeds, and a superpixel segment set as inputs and outputs reliable metric depth. (a) In the overall framework, segment-wise recovery followed by pixel-wise refinement forms a coarse-to-fine pipeline. (b) Per-segment calibration first recovers scale for segments containing projected 3D seeds; we then propagate these calibration parameters to unseeded segments via an optimization on a segment graph. (c) Based on coarse depth, we derive pixel-wise discontinuities to guide the paths of depth propagation. We formulate the geodesic problem as a path-integral optimization and solve it efficiently via dynamic programming, progressing from local updates to a global solution.

tion outputs. Nevertheless, these methods [23, 38, 41] exhibit limited flexibility and low runtime efficiency. Furthermore, achieving highly generalizable models requires collecting large-scale depth completion datasets and training for days [23], which further increases the cost of supervised learning. In contrast, our Midas Touch approach requires no additional training, is mathematically interpretable, uses only extremely sparse 3D seed points, and flexibly accommodates multiple input modalities.

3. Methodology

We present a universal, coarse-to-fine paradigm for converting relative depth to metric depth using sparse 3D seeds in a mathematically interpretable manner, as shown in Fig. 2. Our approach capitalizes on the strong generalization of depth foundation models without any fine-tuning, avoiding the risk of performance degradation. Beginning with a model's relative depth output, we first introduce a coarse, segment-wise scale recovery strategy to reduce local scale inconsistencies (Sect. 3.1). We then propose a fine, pixel-wise refinement strategy by casting the task as a discontinuity-aware geodesic problem (Sect. 3.2). Utilizing the above strategies, our method recovers accurate metric depth even when the depth foundation model is lightweight or only moderately accurate, thereby improving overall computational efficiency (Sect. 3.3).

3.1. Segment-Wise Recovery

When converting relative depth to metric depth, the scale often varies across local regions, so a single global rescaling is insufficient. We therefore partition the image into segments. Due to the sparsity of 3D seeds, some segments receive no seed projections and lack a reliable scale. To address this, we construct a segment graph and propagate recovered scales from seeded to unseeded segments.

3.1.1. Per-Segment Calibration

Given a color image $\boldsymbol{I}$ and a set of 3D seeds $\mathcal{X}$, we project $\mathcal{X}$ onto the image plane at the same resolution as $\boldsymbol{I}$. Applying superpixel segmentation [1, 7, 54] to $\boldsymbol{I}$ produces segments $\mathcal{S} = \{\boldsymbol{S}_i\}$, where $1 \le i \le |\mathcal{S}|$ and $|\cdot|$ denotes set cardinality. Let $\mathcal{Q}$ index the superpixels containing projected seed priors; then $\mathcal{S} = \{\boldsymbol{S}_i\}_{i\in\mathcal{Q}} \cup \{\boldsymbol{S}_i\}_{i\notin\mathcal{Q}}$.

For each $i \in \mathcal{Q}$, let $\mathcal{X}_i = \{\boldsymbol{X}_i^j\}$ denote the set of 3D seeds that projected on $\mathcal{S}_i$, where $1 \le j \le |\mathcal{X}_i|$. The 3D seeds originate from different sources, such as LiDAR or multi-view stereo, and each seed $\boldsymbol{X}_i^j$ can provide a scalar proxy ξ_i^j that is equivalent to depth z_i^j via a monotonic bijection (details provided in the supplementary material). Let d_i^j be the relative depth, at the pixel corresponding to $\boldsymbol{X}_i^j$, predicted by a depth foundation model. We estimate a per-segment calibration function $g_i : d \mapsto \xi$ that maps relative depth to the scalar proxies by aligning the empirical distributions of $\{d_i^j\}$ and $\{\xi_i^j\}$ within $\boldsymbol{S}_i$ $(i \in \mathcal{Q})$. In practice, g_i can be obtained via least squares or median matching. To enable efficient parallel computation, we store the calibrated parameters of g_i in a lookup table, allowing us to propagate these parameters to the uncalibrated segments in the subsequent sections.

3.1.2. Sparse Graph Optimization

We extend the per-segment calibration from the supported set $\{\boldsymbol{S}_i\}_{i\in\mathcal{Q}}$ to the full segmentation $\mathcal{S}$ in a manner that is metrically faithful and spatially consistent. We encode this set $\mathcal{S}$ with a superpixel graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ whose vertices $\mathcal{V}$ correspond to segments and whose edges capture geometric proximity. For the vertex in $\mathcal{V}$, let $\boldsymbol{\theta}_i$ denote the parameters of the local calibration function g_i for segment $\boldsymbol{S}_i$. $\forall i \in \mathcal{Q}$, $\hat{\boldsymbol{\theta}}_i$ is the per-segment fit obtained in Sect. 3.1.1; $\forall i \notin \mathcal{Q}$, no seeds are available and $\boldsymbol{\theta}_i$ must be inferred from the context. For each edge (i, j) in $\mathcal{E}$, edge weight w_{ij} is defined by a decaying kernel of the inter-centroid distance. We also use an adaptive, median-based scale parameter that normalizes the dynamic range for numerical stability.

To reduce memory and improve computational efficiency, we sparsify $\mathcal{G}$ by retaining, for each node v_i, only its N nearest neighbors under the distance induced by w_{ij}. Estimating the full set of calibration parameters $\{\boldsymbol{\theta}_i\}$ is then posed as a graph-regularized quadratic problem:

$$\min_{\{\boldsymbol{\theta}_i\}} \sum_{i\in\mathcal{Q}} \left\|\boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i\right\|^2 + \sum_{(i,j)\in\mathcal{E}} w_{ij} \left\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\right\|^2. \quad (1)$$

The objective encourages neighboring segments to share similar transfer parameters while remaining faithful to the per-segment anchors where available. To solve the problem efficiently, we adopt a closed-form approximation to (1), thereby propagating reliable calibration from $\mathcal{Q}$ into $i \notin \mathcal{Q}$.

Finally, the learned graph $\mathcal{G}$ is lifted back to the image domain by assigning g_i to all pixels $\boldsymbol{p} \in \mathcal{S}_i$ and applying it to the foundation model's relative depth: $\xi(\boldsymbol{p}) = g_i\big(d(\boldsymbol{p})\big)$. This provides an estimate of metric depth, based on the known monotone bijection between ξ and z. The segment-wise, graph-regularized transformation aligns relative depth predictions with available metric seeds and propagates a coherent, edge-aware calibration into unseeded segments.

3.2. Pixel-Wise Refinement

Section 3.1 provides coarse metric depth estimations via segment-wise recovery, but per-segment matching leaves residual pixel-level errors. A simple indication is that the real depth values on the 3D seed projections still deviate from the coarse depth. Moreover, these pixels and their neighbors often lie on the same physical surface and thus tend to share similar errors, leading to local depth inaccuracies. We therefore introduce a mathematically interpretable, pixel-wise refinement to correct these residuals.

3.2.1. Discontinuity-Aware Geodesic Cost

Our goal is to refine the coarse metric depth while discouraging wrong propagation across 3D discontinuities. We formulate the coarse metric depth $z : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ as a function of image coordinates $(u, v)^\top$. For a pixel $\boldsymbol{p} \in \Omega$, we define its neighbor $\mathcal{N}_{\boldsymbol{p}} = \{\boldsymbol{q} : \|\boldsymbol{q} - \boldsymbol{p}\|_\infty \le 1\}$, where $\|\cdot\|_\infty$ represents the infinity norm.

Proposition 1 (Pathwise bound on the remainder)**.** *For any $\boldsymbol{p}$ and $\boldsymbol{q}$, define the first-order remainder*

$$R(\boldsymbol{p}, \boldsymbol{q}) \triangleq z(\boldsymbol{q}) - z(\boldsymbol{p}) - \tfrac{1}{2}\big(\nabla z(\boldsymbol{p}) + \nabla z(\boldsymbol{q})\big)^\top (\boldsymbol{q} - \boldsymbol{p}).^1 \quad (2)$$

Assume that the second partial derivatives z_{uu} and z_{vv} exist and are integrable.[2] *Then there exists an axis-parallel polygonal path $\mathcal{L}_{\boldsymbol{p}\to\boldsymbol{q}}$ connecting $\boldsymbol{p}$ and $\boldsymbol{q}$ such that*

$$\big|R(\boldsymbol{p}, \boldsymbol{q})\big| \le \int_{\mathcal{L}_{\boldsymbol{p}\to\boldsymbol{q}}} \phi(u, v)\, ds, \qquad \phi(u, v) \triangleq \sqrt{z_{uu}^2(u, v) + z_{vv}^2(u, v)}. \quad (3)$$

Using the Cauchy-Schwarz inequality, Proposition 1 can be proved (details provided in the supplementary material). Proposition 1 guarantees the existence of at least one admissible curve; let $\mathscr{L}_{\boldsymbol{p}\to\boldsymbol{q}}$ be the family of admissible curves, which is therefore nonempty. For any $\mathcal{L} \in \mathscr{L}_{\boldsymbol{p}\to\boldsymbol{q}}$, the minimum accumulated quantity

$$d_\phi(\boldsymbol{p}, \boldsymbol{q}) = \inf_{\mathcal{L}\in\mathscr{L}_{\boldsymbol{p}\to\boldsymbol{q}}} \int_{\mathcal{L}} \phi(u, v)\, ds \quad (4)$$

is the ***discontinuity-aware geodesic cost***, where inf denotes the infimum. d_ϕ is exactly the geodesic distance under the conformal Riemannian metric $\phi^2 \boldsymbol{I}_2$, where $\boldsymbol{I}_2$ denotes the identity matrix. A rigorous proof of this statement is provided in the supplementary material.

In practice, we treat ϕ as a local discontinuity density and discretize the line integral in (3) by a Riemann-sum over single-pixel moves. For a pixel path $\Pi : \boldsymbol{p}_0 \to \boldsymbol{p}_1 \to \cdots \to \boldsymbol{p}_K$ and its polygonal chain $\mathcal{L}_\Pi = \bigcup_{k=0}^{K-1} \mathcal{L}_k$, the path cost can be approximated as follows:

$$\int_{\mathcal{L}_\Pi} \phi\, ds = \sum_{k=0}^{K-1} \int_{\mathcal{L}_k} \phi\, ds \approx \sum_{k=0}^{K-1} \underbrace{\ell(\boldsymbol{p}_k, \boldsymbol{p}_{k+1})\, \phi(\boldsymbol{p}_{k+1})}_{W(\boldsymbol{p}_k \to \boldsymbol{p}_{k+1})}, \quad (5)$$

where $\ell(\cdot, \cdot)$ is the step length. When the depth map exhibits discontinuities at object boundaries, d_ϕ does not violate the conditions under which (3), (4), and (5) hold. On the contrary, from a discretized computational perspective, such discontinuities result in a large value of ϕ, so any curve that crosses such regions accumulates a large cost. Taking the minimum over curves yields a path-independent quantity d_ϕ that penalizes discontinuity crossings, thereby confining depth propagation to a reliable spatial extent.

[1] Compared with the common first-order remainder $r(\boldsymbol{p}, \boldsymbol{q}) = z(\boldsymbol{q}) - z(\boldsymbol{p}) - \nabla z(\boldsymbol{p})^\top (\boldsymbol{q} - \boldsymbol{p})$, the remainder in (2) is antisymmetric, $R(\boldsymbol{p}, \boldsymbol{q}) + R(\boldsymbol{q}, \boldsymbol{p}) = 0$, hence $|R(\boldsymbol{p}, \boldsymbol{q})| = |R(\boldsymbol{q}, \boldsymbol{p})|$. This removes orientation bias and isolates second-order variation; in particular, $R(\boldsymbol{p}, \boldsymbol{q}) = 0$ for any affine z.

[2] It suffices that z_{uu} and z_{vv} be integrable along an axis-parallel path between $\boldsymbol{p}$ and $\boldsymbol{q}$.

3.2.2. Dynamic Programming via Path Integrals

Dynamic programming iteratively minimizes a cost (or energy) function using a reliable update rule, thereby converging to a stable solution. Since the geodesic cost can be approximated by (5), therefore, we rewrite it as

$$d_\phi(\boldsymbol{p}_0, \boldsymbol{p}_K) \leq \inf(W(\boldsymbol{p}_{K-1} \rightarrow \boldsymbol{p}_K)) + d_\phi(\boldsymbol{p}_0, \boldsymbol{p}_{K-1}), \tag{6}$$

which serves as the cost function at iteration K. The scheme in (6) establishes a correspondence between computing the line integral along a path and the cost-function optimization of dynamic programming. Furthermore, it ensures that interactions between pixels are not confined to a local neighborhood, thereby expanding the effective receptive field. We initialize the costs at the reliable projected 3D seed pixels to the minimum by default. After the dynamic-programming iterations, (6) yields the smoothest discrete path among all paths from these seed pixels to $\boldsymbol{p}$ with at most K moves. To update the value at a point $\boldsymbol{p}$ using $\boldsymbol{q}$ within this path, we use the following expression:

$$z^{(k+1)}(\boldsymbol{p}) = \left(1 - \tfrac{1}{k+1}\right) z^{(k)}(\boldsymbol{p}) + \tfrac{1}{k+1} \hat{z}^{(k)}(\boldsymbol{p} \mid \boldsymbol{q}, \Delta\boldsymbol{p}), \tag{7}$$

where $\hat{z}^{(k)}(\boldsymbol{p} \mid \boldsymbol{q}, \Delta\boldsymbol{p}) = \boldsymbol{\alpha}^{(k)}(\boldsymbol{q})^\top \boldsymbol{\Psi}(\Delta\boldsymbol{p})$, $\boldsymbol{\alpha}^{(k)}(\boldsymbol{q})$ denotes local coefficients estimated from data around $\boldsymbol{q}$ at iteration k, $\Delta\boldsymbol{p} = \boldsymbol{p} - \boldsymbol{q}$ denotes the one-step displacement, and $\boldsymbol{\Psi}$ is a chosen set of basis functions on the step domain. Using the harmonic step-size sequence $\frac{1}{k+1}$, the overall update forms a convex combination of the previous estimate and current prediction, which enhances stability while incorporating new local information.

3.3. Computational Efficiency Improvements

To improve computational efficiency and practical applicability, we reduce the parameter count of the depth foundation model via knowledge distillation. According to previous studies [10, 19], high-quality datasets are crucial for distilling vision foundation models. Therefore, we leverage qualified raw data collected from both real-world and simulated environments. Inspired by the studies [48, 54], we adopt TinyViT [47] and EfficientViT [26] as backbones and use DepthAnythingV2 as the teacher network. We employ both feature distillation and logit distillation objectives. As shown in Sect. 4.3, despite a substantial reduction in parameters, the resulting metric depth accuracy remains comparable, and runtime efficiency improves markedly, which enables practical deployment in downstream autonomous driving and robotics applications.

4. Experiments

4.1. Experimental Setup

To compare our method with existing approaches, we conduct zero-shot generalization evaluations on nuScenes [5], DDAD [14], Make3D [33], DIODE [40], ETH3D [34], ScanNet [6], VOID [46], SUN-RGBD [36], HAMMER [18], IBims-1 [21], KITTI [11], and NYU-Depth V2 [35] datasets. Detailed descriptions of the datasets' split and statistics can be found in the supplementary material. The datasets used for distillation are VKITTI2 [4], Hypersim [32], TartanAir [43], and SA-1B [20]. The performance of depth is evaluated using several standard metrics, including root mean squared error (RMSE), mean absolute error (MAE), absolute relative error (AbsRel), squared relative error (SqRel), the accuracy metric (δ_i) under thresholds of 1.25^i, and the scale-invariant error in log-scale ($\mathrm{SI}_{\log}$).

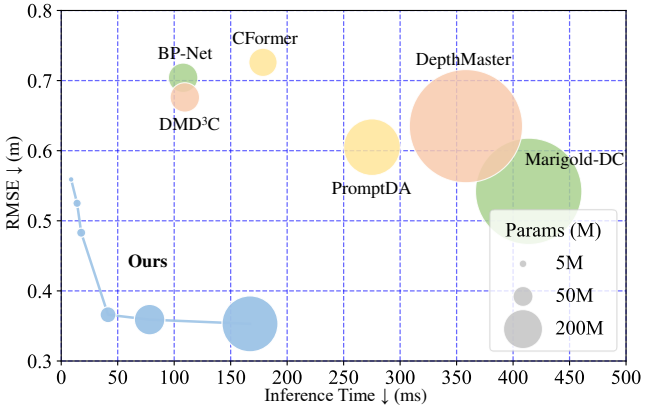

Figure 3. Evaluation results on VOID1500 at input resolution of 480×640 pixels using an RTX 3090. For Ours, the left-to-right ordering corresponds to the following backbones: EfficientViT-B0, EfficientViT-B1, TinyViT, DepthAnythingV2-S, DepthAnythingV2-B, and DepthAnythingV2-L.

4.2. Comparison with State-of-the-Art Methods

Depth Completion. Table 1 shows that our method consistently outperforms previous SoTA depth completion approaches, with improvements in terms of MAE and RMSE. Since KITTI and NYUv2 are widely used for training, we exclude them from our zero-shot evaluation to avoid potential train-test overlap. Outdoor scenes are generally more challenging to generalize to than indoor scenes. For instance, in datasets such as nuScenes, where point clouds are sparse and depth values are relatively high, nearly all methods struggle. Nevertheless, our approach surpasses the SoTA method Marigold-DC [41], reducing MAE and RMSE by 0.418 and 0.537, respectively.

Depth Prediction. In Table 2, we report results for both discriminative and generative relative depth estimation methods, as well as metric depth estimation methods. For relative depth estimation, the original evaluations already require 3D data as input for global scale recovery; we preserve these input conditions and only sample 1.0%-1.5% of the available 3D data per scene (on KITTI, we use the official

Table 1. Quantitative comparison with SoTA depth completion methods. All methods are evaluated in a zero-shot setting. ↓ lower is better.

Method	nuScenes		DDAD		Make3D		DIODE		ETH3D	
	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓
CFormer [56] (CVPR'23)	15.672	10.528	9.606	3.328	10.749	5.035	2.297	0.976	2.796	1.940
LRRU [44] (ICCV'23)	13.660	8.472	9.164	2.738	13.023	5.893	2.795	1.947	3.016	2.337
BP-Net [38] (CVPR'24)	15.092	10.592	8.903	2.712	12.034	5.353	2.724	1.831	3.365	2.665
DMD^{3}C [23] (CVPR'25)	5.556	3.112	7.766	2.498	12.019	5.575	2.262	1.127	0.935	0.285
PromptDA [24] (CVPR'25)	9.072	5.325	8.487	2.891	12.392	5.803	2.139	1.287	1.172	0.479
Marigold-DC [41] (ICCV'25)	4.924	2.595	6.449	2.364	8.926	4.932	1.987	0.887	0.706	0.245
MTD (Ours)	**4.387**	**2.177**	**5.252**	**1.834**	**8.581**	**4.776**	**1.736**	**0.761**	**0.662**	**0.177**
Method	ScanNet		VOID1500		SUN-RGBD		HAMMER		IBims-1	
	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓
CFormer [56] (CVPR'23)	0.223	0.118	0.726	0.261	0.442	0.218	0.101	0.055	0.177	0.040
LRRU [44] (ICCV'23)	1.175	0.773	0.698	0.232	1.304	0.808	1.418	1.137	0.298	0.107
BP-Net [38] (CVPR'24)	1.326	1.055	0.704	0.230	1.327	1.003	1.620	1.415	0.302	0.119
DMD^{3}C [23] (CVPR'25)	0.152	0.070	0.676	0.225	0.423	0.106	0.098	0.047	0.286	0.083
PromptDA [24] (CVPR'25)	0.179	0.072	0.605	0.191	0.264	0.095	0.106	0.070	0.308	0.122
Marigold-DC [41] (ICCV'25)	0.145	0.059	0.505	0.151	0.238	0.067	**0.054**	0.037	**0.176**	**0.038**
MTD (Ours)	**0.129**	**0.049**	**0.366**	**0.138**	**0.220**	**0.050**	0.093	**0.034**	0.190	0.072

Table 2. Quantitative comparison with SoTA zero-shot monocular depth estimation methods. The upper part lists data-driven relative depth estimation methods, the middle part presents relative depth estimation methods based on diffusion models, and the lower part represents the metric depth estimation methods.

Method	with **Ours**	KITTI		NYUv2		ETH3D		ScanNet		DIODE	
		AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑
MiDaS [2]		0.183	0.711	0.095	0.915	0.190	0.884	0.099	0.907	0.266	0.713
MiDaS [2]	✓	0.069	0.929	0.048	0.949	0.055	0.944	0.015	0.991	0.113	0.864
LeReS [51]		0.149	0.784	0.090	0.916	0.171	0.777	0.091	0.917	0.271	0.766
LeReS [51]	✓	0.035	0.974	0.014	0.994	0.022	0.984	0.013	0.994	0.097	0.885
DPT [31]		0.111	0.881	0.091	0.919	0.115	0.929	0.084	0.932	0.269	0.730
DPT [31]	✓	0.032	0.976	0.016	0.994	0.019	0.985	0.014	0.994	0.096	0.887
Depth Pro [3]		0.077	0.949	0.044	0.975	0.060	0.965	0.042	0.980	0.321	0.752
Depth Pro [3]	✓	0.034	0.975	**0.012**	**0.995**	0.028	0.974	0.020	0.986	0.093	0.889
DepthAnythingV2 [50]		0.080	0.946	0.043	0.980	0.062	0.980	0.043	0.981	0.260	0.759
DepthAnythingV2 [50]	✓	**0.022**	**0.987**	0.013	0.995	**0.017**	**0.988**	0.016	0.991	**0.093**	**0.917**
Marigold [19]		0.099	0.916	0.055	0.964	0.065	0.960	0.064	0.951	0.308	0.773
Marigold [19]	✓	0.041	0.966	0.017	0.991	0.025	0.978	0.014	0.991	0.094	0.896
GeoWizard [9]		0.097	0.921	0.052	0.966	0.064	0.961	0.061	0.953	0.297	0.792
GeoWizard [9]	✓	0.047	0.962	0.014	0.994	0.021	0.984	**0.013**	**0.994**	0.097	0.885
Lotus [15]		0.093	0.928	0.053	0.967	0.068	0.953	0.060	0.963	0.228	0.738
Lotus [15]	✓	0.032	0.975	0.015	0.993	0.022	0.985	0.015	0.992	0.108	0.868
DepthMaster [37]		0.082	0.937	0.050	0.972	0.053	0.974	0.055	0.967	0.215	0.776
DepthMaster [37]	✓	0.043	0.965	0.016	0.993	0.021	0.984	0.016	0.992	0.095	0.888
Metric3Dv2 [16]		0.051	0.976	0.067	0.973	0.137	0.825	0.047	0.990	0.246	0.823
Metric3Dv2 [16]	✓	0.027	0.980	0.014	0.994	0.019	0.986	0.015	0.992	0.099	0.881
UniDepthV2 [29]		0.076	0.952	0.062	0.970	0.150	0.865	0.058	0.975	0.251	0.795
UniDepthV2 [29]	✓	0.032	0.977	0.017	0.993	0.023	0.980	0.014	0.993	0.096	0.887

sparse data). As shown in Table 2, MTD serves as a plug-and-play component for depth foundation models, utilizing their relative depth output and producing accurate metric depth, which consistently reduces AbsRel and increases δ_1. Moreover, Table 2 compares MTD across different depth foundation models. When paired with DepthAnythingV2, MTD achieves the best overall performance. Accordingly, we adopt DepthAnythingV2 as the primary backbone in our subsequent hardware and application experiments.

Inference Time. In Fig. 3, we compare our method with SoTA baselines in terms of RMSE and inference time. Benefiting from MTD's flexibility, our framework offers greater latitude to balance accuracy and efficiency. Notably, the dominant runtime cost lies in the front-end depth foundation model; the back end adds negligible overhead. For example, for an input image with a resolution of 480×640 pixels on an RTX 3090, our back end requires only 1.9 ms, with memory usage below 1.8 GB and GPU utilization under 4%. This underscores the importance of efficient relative depth backbones for lightweight frameworks.

4.3. Ablation Studies

We validate the rationality and efficacy of our method through extensive ablation studies, specifically focusing on

Table 3. Unified ablation on KITTI (outdoor) and VOID (indoor) for segment-wise recovery and pixel-wise refinement strategy.

Module	Factor	KITTI		VOID	
		RMSE↓	MAE↓	RMSE↓	MAE↓
Per-Segment Calibration	median	10.891	2.169	0.898	0.358
	least squares	7.013	1.802	0.791	0.307
	Domain: z^{-1}	6.782	1.794	0.614	0.238
Sparse Graph Optimization	global-based	2.521	0.687	0.554	0.169
	graph-based	2.232	0.608	0.459	0.150
Dynamic Programming	w/o d_ϕ	2.618	0.661	0.482	0.158
	B-spline	2.112	0.578	0.442	0.147
	polynomial	2.049	0.551	0.429	0.145
	k=3	2.028	0.532	**0.366**	**0.138**
	k=5	**1.913**	**0.498**	0.391	0.141

the segment-wise recovery strategy, the pixel-wise refinement strategy, the selection of the depth foundation models, and the 3D seed sparsity.

Segment-Wise Recovery Strategy. As shown in Table 3, we evaluate both on the outdoor and indoor datasets. We first validate our per-segment calibration to determine both the proxy domain and the fitting strategy. In particular, we compare two standard alignment schemes for monocular depth estimation [12, 13, 19, 49, 55]: median alignment and least-squares fitting. We further adopt inverse depth z^{-1} as the proxy representation, yielding a measurable improvement in accuracy. Moreover, after constructing a sparse graph over segments and optimizing it, we observe consistent improvements over the global baseline.

Pixel-Wise Refinement Strategy. We solve the geodesic problem using dynamic programming and, therefore, perform ablation studies on the proposed discontinuity-aware geodesic cost and the dynamic programming jointly. As shown in Table 3, without our geodesic cost, standard edge extractors fail to represent discontinuities effectively. For the basis functions in (7), polynomial functions outperform B-splines. We also find that increasing k in (7) expands the receptive field and improves accuracy.

Foundation Model's Capacity. To assess our method's practical applicability, we conduct ablation studies across different depth foundation models. In Fig. 3, RMSE remains stable until the parameter count drops below ∼20M. Although RMSE increases for TinyViT, EfficientViT-B0, and EfficientViT-B1, these backbones are extremely lightweight, and their RMSE remains within an acceptable range relative to other SoTA methods. In Fig. 4(a), we evaluate with two mainstream depth foundation models, DepthAnythingV2 and MiDaS. We observe that under the global least-squares baseline, the MAE gap between DepthAnythingV2 and MiDaS is substantial; after incorporating our algorithm, this gap is markedly reduced. This demonstrates that our method can bridge the performance gaps across different depth foundation models and does not rely on high-capacity foundation models.

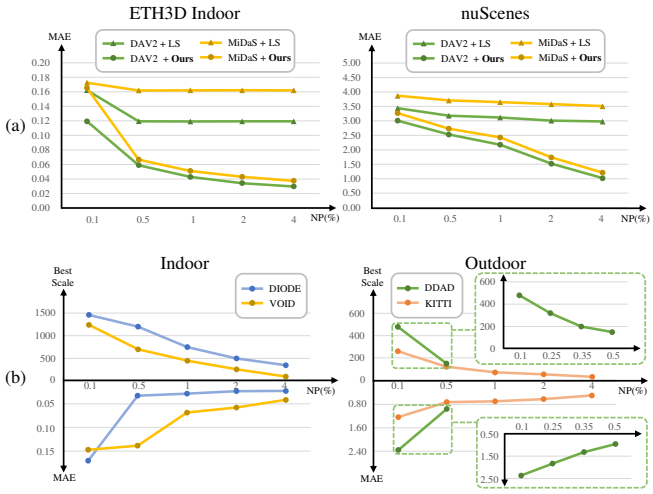

Figure 4. Ablation on 3D Seed Sparsity and Segment Scale. (a) **Effect of the number of 3D seed points (NP)**. We compare the MAE obtained with a global least-squares baseline and with our method as NP varies. The left panel reports results on ETH3D (indoor); the right panel reports results on nuScenes (outdoor). "LS" represents global least-squares method, and "DAV2" denotes the DepthAnythingV2. (b) **Best segment scale at fixed NP**. For both indoor and outdoor datasets, we conduct a hyperparameter search over the segment scale at each fixed NP, selecting the value that minimizes MAE.

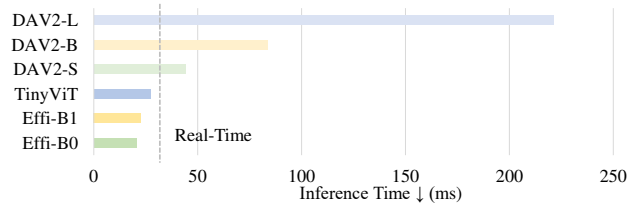

Figure 5. Inference time on embedded system (NVIDIA Jetson AGX Orin Platform) after acceleration. "DAV2" denotes the DepthAnythingV2, and "Effi" represents the EfficientViT.

3D Seed Sparsity and Segment Scale. In Fig. 4(a), we compare the MAE as the number of 3D seed points (NP) varies, using both global least squares and our method. When the 3D seeds are extremely sparse, our method degenerates to the least-squares baseline, serving as a built-in safeguard. As the point cloud becomes denser, however, our approach yields substantial MAE reductions, whereas global least squares shows little further improvement. In Fig. 4(b), for multiple datasets and NP settings, we perform a hyperparameter search over the segment-scale parameter to minimize MAE. As NP increases, the optimal segment scale consistently decreases, and the best MAE likewise drops, since additional 3D seeds enable finer metric depth detail. These trends provide practical guidance for selecting hyperparameters under different NP situations.

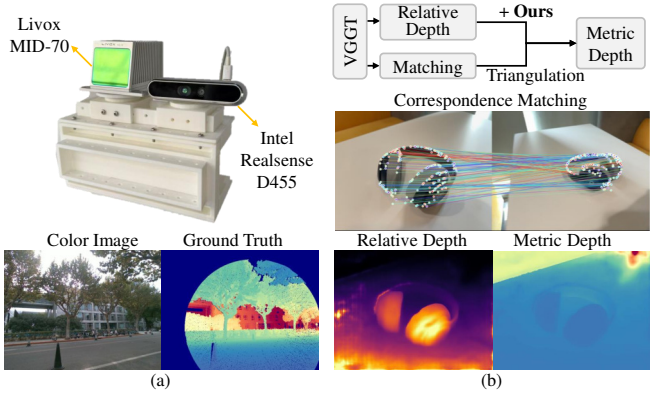

Figure 6. Evaluation setup and visualization. (a) We use a Livox MID-70 LiDAR as ground truth to evaluate the depth rectification of the Intel RealSense D455. (b) Our method plugs into the back end of VGGT to produce metric 3D reconstructions.

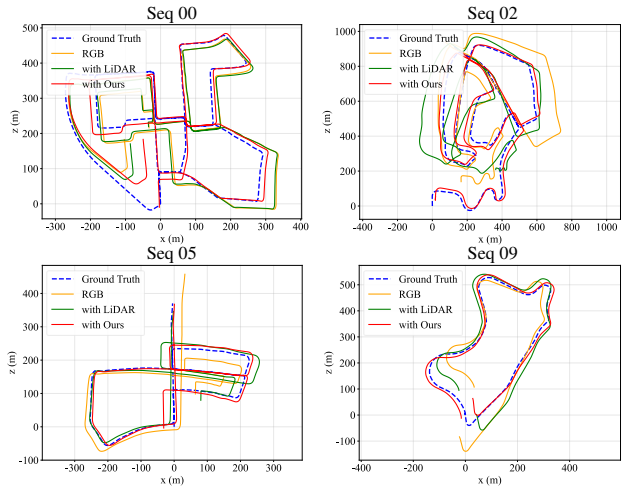

Figure 7. Qualitative results on the KITTI Odometry benchmark.

4.4. Downstream Applications

In Fig. 5, we further report inference time after TensorRT and multi-thread acceleration on the NVIDIA Jetson AGX Orin Platform. With our nonparametric lightweight algorithm, the embedded system achieves real-time performance. Fig. 6 and Table 4 demonstrate the effectiveness of our method across multiple downstream applications. Thanks to the plug-and-play design, our approach integrates easily into existing pipelines. Compared with raw depth from commonly used range cameras, MTD substantially improves data quality in both indoor and outdoor settings. In addition, when combined with large models such as VGGT [42], our method enables multi-view stereo that surpasses the SoTA MVSAnywhere [17]. Specifically, VGGT outputs relative depth and correspondence matches. Utilizing the collected extrinsic, we triangulate these matches to obtain 3D seeds with known scale, which yields metric depth. Compared with a least-squares alignment, our scale-recovery strategy markedly improves the accuracy of the resulting metric depth. In SLAM experiments, augmenting LiDAR with our method significantly reduces trajectory error compared to leveraging LiDAR alone; qualitative results are shown in Fig. 7. For occupancy prediction, following the pipeline and evaluation protocol in studies [8, 22, 45], we observe a consistent improvement in accuracy.

Table 4. Applications of MTD. (a)-(b) Experimental setups are shown in Fig. 6. (c) We report the RMSE of the absolute trajectory error (ATE-RMSE↓). (d) Quantitative results on KITTI-360; the evaluation metrics and the two-decimal rounding convention follow those in the study [45].

(a) Depth Rectification in Commonly Used Range Camera

Method	Scene	RMSE↓	MAE↓	AbsRel↓	SqRel↓	SI_{log} ↓	δ_1 ↑
Raw	Indoor	0.935	0.387	0.101	0.209	0.579	0.942
Raw + **Ours**		**0.345**	**0.243**	**0.065**	**0.032**	**0.090**	**0.967**
Raw	Outdoor	2.408	1.248	0.190	0.753	0.971	0.845
Raw + **Ours**		**1.287**	**0.889**	**0.150**	**0.289**	**0.219**	**0.886**

(b) Multi-View Stereo

Method	RMSE↓	MAE↓	AbsRel↓	SqRel↓	SI_{log} ↓	δ_1 ↑
MVSAnywhere[17]	0.088	0.041	**0.054**	0.011	0.099	0.957
VGGT[42] + Align	0.107	0.067	0.108	0.015	0.147	0.889
VGGT[42] + **Ours**	**0.079**	**0.034**	0.054	**0.010**	**0.088**	**0.971**

(c) SLAM on KITTI Odometry

Method	Seq 00	Seq 02	Seq 05	Seq 06
Droid [39]	66.562	84.828	41.973	72.402
Droid + LiDAR	63.295	77.610	31.641	104.552
Droid + **Ours**	**25.017**	**25.698**	**12.233**	**12.194**
Method	Seq 07	Seq 08	Seq 09	Seq 10
Droid [39]	13.785	64.552	60.414	11.394
Droid + LiDAR	9.867	60.201	42.493	46.627
Droid + **Ours**	**5.994**	**20.738**	**23.077**	**10.871**

(d) Occupancy Predicition

Method	Scene		Object	
	O^s_{acc} ↑	IE^s_{acc} ↑	O^o_{acc} ↑	IE^o_{acc} ↑
Monodepth2 [13]	0.90	N/A	0.69	N/A
Monodepth2 + 4m	0.90	0.59	0.70	0.53
PixelNeRF [53]	0.89	0.62	0.67	0.53
BTS [45]	0.92	0.69	0.79	0.69
KYN [22]	0.92	0.70	0.79	0.69
ViPOcc [8]	0.93	0.71	0.79	0.69
BTS + **Ours**	**0.94**	**0.72**	**0.80**	**0.70**

5. Conclusion

In this paper, we introduced a novel paradigm for recovering depth scale, comprising a segment-wise scale recovery strategy and a pixel-wise refinement strategy. Extensive experiments demonstrate strong cross-domain generalization and high accuracy of our method, and we further show that it can be flexibly integrated into a variety of downstream tasks. As future work, we will investigate using 3D seeds only for initialization and, within a continual-learning framework, gradually eliminate this requirement.

Acknowledgements

This research was supported by the National Natural Science Foundation of China under Grant 62473288, Grant 62371343, Grant 62233013, and Grant 62333017, the National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, Xi'an Jiaotong University (No. HMHAI-202406), the Fundamental Research Funds for the Central Universities, NIO University Programme (NIO UP), and the Xiaomi Young Talents Program.

References

[1] Radhakrishna Achanta et al. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012. 3

[2] Reiner Birkl et al. MiDaS v3.1 – A Model Zoo for Robust Monocular Relative Depth Estimation. *arXiv preprint arXiv:2307.14460*, 2023. 2, 6

[3] Aleksei Bochkovskii et al. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 6

[4] Yohann Cabon et al. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 5

[5] Holger Caesar et al. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5

[6] Angela Dai et al. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 5

[7] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004. 3

[8] Yi Feng et al. ViPOcc: leveraging visual priors from vision foundation models for single-view 3d occupancy prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 3004–3012, 2025. 8

[9] Xiao Fu et al. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 241–258. Springer, 2024. 2, 6

[10] Yongtao Ge et al. GeoBench: Benchmarking and Analyzing Monocular Geometry Estimation Models. *arXiv preprint arXiv:2406.12671*, 2024. 1, 2, 5

[11] Andreas Geiger et al. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. IEEE, 2012. 5

[12] Clement Godard et al. Unsupervised Monocular Depth Estimation With Left-Right Consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 7

[13] Clement Godard et al. Digging Into Self-Supervised Monocular Depth Estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 7, 8

[14] Vitor Guizilini et al. 3D Packing for Self-Supervised Monocular Depth Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5

[15] Jing He et al. Lotus: Diffusion-based visual foundation model for high-quality dense prediction. *arXiv preprint arXiv:2409.18124*, 2024. 2, 6

[16] Mu Hu et al. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2, 6

[17] Sergio Izquierdo et al. MVSAnywhere: Zero-Shot Multi-View Stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11493–11504, 2025. 8

[18] HyunJun Jung et al. Is my Depth Ground-Truth Good Enough? HAMMER–Highly Accurate Multi-Modal Dataset for DEnse 3D Scene Regression. *arXiv preprint arXiv:2205.04565*, 2022. 5

[19] Bingxin Ke et al. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9492–9502, 2024. 2, 5, 6, 7

[20] Alexander Kirillov et al. Segment Anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 5

[21] Tobias Koch et al. Comparison of monocular depth estimation methods using geometrically relevant metrics on the IBims-1 dataset. *Computer Vision and Image Understanding*, 191:102877, 2020. 5

[22] Rui Li et al. Know Your Neighbors: Improving Single-View Reconstruction via Spatial Vision-Language Reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9848–9858, 2024. 8

[23] Yingping Liang et al. Distilling Monocular Foundation Model for Fine-grained Depth Completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22254–22265, 2025. 2, 3, 6

[24] Haotong Lin et al. Prompting Depth Anything for 4K Resolution Accurate Metric Depth Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17070–17080, 2025. 2, 6

[25] Yuankai Lin et al. Dynamic spatial propagation network for depth completion. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 1638–1646, 2022. 2

[26] Xinyu Liu et al. EfficientViT: Memory Efficient Vision Transformer With Cascaded Group Attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14420–14430, 2023. 5

[27] Jinsun Park et al. Non-local spatial propagation network for depth completion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 120–136. Springer, 2020. 2

[28] Luigi Piccinelli et al. Unidepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10106–10116, 2024. 2

[29] Luigi Piccinelli et al. UniDepthV2: Universal Monocular Metric Depth Estimation Made Simpler. *arXiv preprint arXiv:2502.20110*, 2025. 2, 6

[30] René Ranftl et al. 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. 2

[31] René Ranftl et al. Vision Transformers for Dense Prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12179–12188, 2021. 6

[32] Mike Roberts et al. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10912–10922, 2021. 5

[33] Ashutosh Saxena et al. Make3D: Learning 3D Scene Structure from a Single Still Image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):824–840, 2009. 5

[34] Thomas Schops et al. 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 (CVPR)*, 2017. 5

[35] Nathan Silberman et al. Indoor segmentation and support inference from rgbd images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 746–760. Springer, 2012. 5

[36] Shuran Song et al. SUN RGB-D: A RGB-D Scene Understanding Benchmark Suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 5

[37] Ziyang Song, Zerong Wang, Bo Li, Hao Zhang, Ruijie Zhu, Li Liu, Peng-Tao Jiang, and Tianzhu Zhang. Depthmaster: Taming diffusion models for monocular depth estimation. *arXiv preprint arXiv:2501.02576*, 2025. 2, 6

[38] Jie Tang et al. Bilateral propagation network for depth completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9763–9772, 2024. 2, 3, 6

[39] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in neural information processing systems (NeurIPS)*, 34:16558–16569, 2021. 8

[40] Igor Vasiljevic et al. Diode: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019. 5

[41] Massimiliano Viola et al. Marigold-DC: Zero-Shot Monocular Depth Completion with Guided Diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5359–5370, 2025. 2, 3, 5, 6

[42] Jianyuan Wang et al. VGGT: Visual Geometry Grounded Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5294–5306, 2025. 8

[43] Wenshan Wang et al. TartanAir: A Dataset to Push the Limits of Visual SLAM. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4909–4916, 2020. 5

[44] Yufei Wang et al. LRRU: Long-short Range Recurrent Updating Networks for Depth Completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9422–9432, 2023. 2, 6

[45] Felix Wimbauer et al. Behind the scenes: Density fields for single view reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9076–9086, 2023. 8

[46] Alex Wong et al. Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters*, 5 (2):1899–1906, 2020. 5

[47] Kan Wu et al. Tinyvit: Fast pretraining distillation for small vision transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 68–85. Springer, 2022. 5

[48] Yunyang Xiong et al. EfficientSAM: Leveraged Masked Image Pretraining for Efficient Segment Anything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16111–16121, 2024. 5

[49] Lihe Yang et al. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10371–10381, 2024. 2, 7

[50] Lihe Yang et al. Depth anything v2. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:21875–21911, 2024. 1, 2, 6

[51] Wei Yin et al. Learning To Recover 3D Scene Shape From a Single Image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 204–213, 2021. 6

[52] Wei Yin et al. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9043–9053, 2023. 2

[53] Yu, Alex and others. pixelNeRF: Neural Radiance Fields From One or Few Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4578–4587, 2021. 8

[54] Chaoning Zhang et al. Faster segment anything: Towards lightweight sam for mobile applications. *arXiv preprint arXiv:2306.14289*, 2023. 3, 5

[55] Mengtan Zhang et al. DCPI-Depth: Explicitly Infusing Dense Correspondence Prior to Unsupervised Monocular Depth Estimation. *IEEE Transactions on Image Processing*, 34:4258–4272, 2025. 7

[56] Youmin Zhang et al. CompletionFormer: Depth Completion With Convolutions and Vision Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18527–18536, 2023. 2, 6

[57] Yiming Zuo and Jia Deng. OGNI-DC: Robust depth completion with optimization-guided neural iterations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 78–95. Springer, 2024. 2