2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Depth Completion in Unseen Field Robotics Environments Using Extremely Sparse Depth Measurements

Marco Job, Thomas Stastny, Eleni Kelasidi, Roland Siegwart, Michael Pantic

***Abstract*— Autonomous field robots operating in unstructured environments require robust perception to ensure safe and reliable operations. Recent advances in monocular depth estimation have demonstrated the potential of low-cost cameras as depth sensors; however, their adoption in field robotics remains limited due to the absence of reliable scale cues, ambiguous or low-texture conditions, and the scarcity of large-scale datasets. To address these challenges, we propose a depth completion model that trains on synthetic data and uses extremely sparse measurements from depth sensors to predict dense metric depth in unseen field robotics environments. A synthetic dataset generation pipeline tailored to field robotics enables the creation of multiple realistic datasets for training purposes. This dataset generation approach utilizes textured 3D meshes from Structure from Motion and photorealistic rendering with novel viewpoint synthesis to simulate diverse field robotics scenarios. Our approach achieves an end-to-end latency of 53 ms per frame on an Nvidia Jetson AGX Orin, enabling real-time deployment on embedded platforms. Extensive evaluation demonstrates competitive performance across diverse real-world field robotics scenarios.**

I. Introduction

Performing autonomous operations in challenging field robotic applications requires a comprehensive understanding of the surroundings, particularly in unknown and unstructured environments. This understanding improves the efficiency, safety, and effectiveness of operations, supporting the development of robots capable of performing complex tasks in diverse and demanding settings. Its importance is particularly evident in navigation, where robots must perceive the geometric properties of the scene. Beyond navigation, it is crucial for obstacle avoidance, mapping, and informed decision-making, all of which are key enabling factors to ensure autonomy and reliability in field scenarios. To fulfill these requirements, robots commonly employ depth sensors from a range of existing modalities, such as Light Detection And Rangings (LiDARs), Time of Flight (ToF) cameras and sensors, radars, and stereo cameras [1, 2]. Recent advances in metric monocular depth estimation (MDE) have motivated the use of simple and inexpensive monocular cameras as depth sensors [3–7]. These works have focused on generalization capabilities in structured domains, utilizing large-scale datasets to build robust models. In general, MDE approaches estimate metric depth using scale cues present in the images, as the problem is inherently ill-posed [3].

The authors MJ, TS, RS and MP are with the Autonomous Systems Lab, ETH Zürich, Switzerland. The corresponding author MJ (marco.job@ntnu.no) and EK are with the Field Robotics Lab, NTNU, Norway. This work has been supported by a Swiss Polar Institute Technograntt, the Armasuisse Research Grant No 4780002580, and by an ETH RobotX research grant funded through the ETH Zurich Foundation.

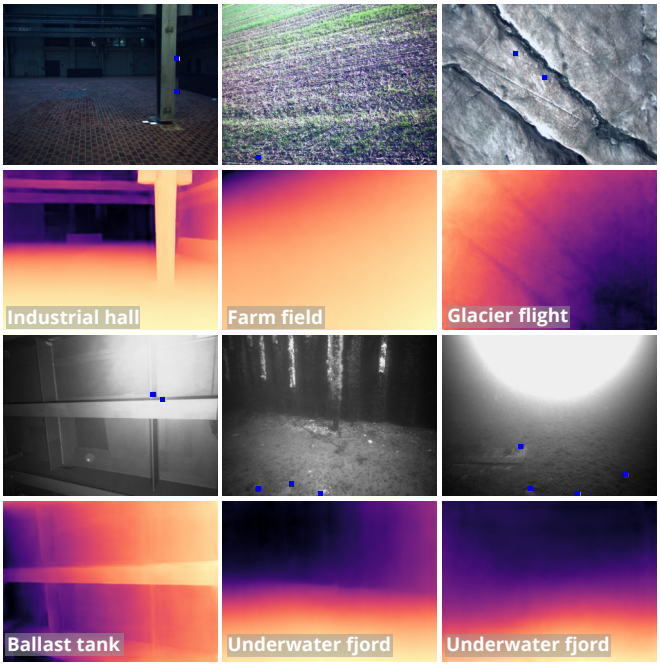

Fig. 1: Our depth completion (DC) approach in five unseen, real-world field robotics environments. The input is a three-channel image and a fourth channel for extremely sparse depth measurements. The blue squares visualize the measurements in this figure. The output is dense metric depth.

However, the direct applicability of MDE methods for metric depth estimation in real-world field robotics environments remains challenging for vision-based approaches due to the frequent absence of scale features and the prevalence of self-similar, ambiguous, or low-texture conditions [1, 2] (see Fig. 1). In addition, few large-scale datasets dedicated to field robotics exist, reflecting the inherent difficulty of collecting data under such challenging conditions. A potential solution to overcome the limitations of metric MDE is to use additional depth measurements to disambiguate the scene. This approach, known as depth completion (DC), commonly uses a set of sparse depth measurements and fills in the gaps to obtain a dense depth image [8]. Depending on the specific constraints of the application, domain or robotic platform, the order of magnitude of depth points can be as low as 10^1 per frame. This is common with commercially available, low-cost, and small-size sensors, such as frequency-modulated continuous-wave (FMCW) radars, single- or few-point ToF sensors, and visually tracked sparse 3D landmarks. Moreover, LiDARs can also produce sparse and noisy depth

measurements if the sensor operates in a degraded mode [2].

Few works have explored the use of such sparse, noisy, and degraded measurements for DC [9–13]. Even fewer studies have investigated this sparsity level combined with generalization to unseen real-world field robotics environments [14–17]. To this end, motivated by this research gap and the lack of field robotics-focused datasets, the main contributions of this paper are summarized as follows:

- A real-time depth completion (DC) approach that predicts metric dense depth from input images and extremely sparse depth measurements (see Fig. 1). Our approach architecturally extends a state-of-the-art (SOTA) MDE method, incorporating the sparse depth measurements, while retaining the MDE's generalization capabilities.
- A synthetic training dataset generation approach tailored to field robotics that utilizes 3D meshes obtained from Structure from Motion (SfM), photorealistic rendering, and novel viewpoint synthesis.
- The release of the four synthetic training datasets, code, trained model, and real-world evaluation datasets to support future research in this domain.

II. Related Work

Eigen et al. [18] strongly influenced the field of MDE with one of the first end-to-end metric depth estimation networks. Since then, the field has seen numerous works [19–24], introducing various architectures and performance improvements. Yet, an inhibitor for wide-scale adaptation in field robotics is the lack of generalization to domains outside the training set.

The research challenge of generalizing to unseen environments is known as zero-shot depth estimation [25]. In this domain, considerable research efforts have focused on training with large-scale datasets and affine-invariant depth estimation [3, 5, 25, 26]. For field robotics, metric depth estimation is more crucial, and recent works have shown increasing performance primarily in structured environments [4, 6, 7, 27, 28]. Notably, [28] encodes camera intrinsics in the input space and [4] employs a canonical image transformation, both to improve generalization with varying cameras. The authors of [6] propose a method that generalizes to a wide range of camera models and intrinsics. In the absence of scale cues, these methods still attempt to solve an ill-posed problem, which is an inherent limitation of MDE rather than the models themselves [3], motivating the incorporation of additional depth information.

To overcome metric MDE limitations, depth completion (DC) methods have been investigated. Initial DC works utilized only LiDAR scans and filled in the gaps in the depth image [8]. Later works began incorporating RGB images to improve the performance. These methods often utilize LiDAR or automotive radar scans, or randomly subsample the ground truth to simulate a sparse sensor [9–13]. Nevertheless, these works are typically trained on constrained domains and thus demonstrate restricted applicability to unseen environments outside the training set [15, 16].

Recently, a direction of generalizable DC has emerged, with notable works including [14–17]. The authors of [14] propose a depth data augmentation framework, and a U-Net-based architecture for metric depth prediction. The approach in [16] uses a custom layer that solves a linear Least Squares (LS) problem to incorporate multi-resolution depth gradients and the sparse depth measurements as constraints into a dense depth map. In [17], the authors utilize a depth foundation model with a prompting module to guide the prediction to the metric domain. These works still utilize approximately one order of magnitude more sparse depth points than what can be available in challenging field robotics environments. Furthermore, since the focus is not field robotics, the training and evaluation datasets in these works are often not representative of this domain. Notably, [15] proposes a diffusion-based DC approach that combines a pretrained affine-invariant MDE model with very sparse depth measurements for metric depth. Due to the diffusion-based architecture, real-time capabilities are a concern, which is particularly relevant for field robotics applications.

Another challenge for increasing adoption in the field robotics domain is the limited availability of training datasets. Several depth datasets exist for MDE and DC, focused on automotive, indoor and outdoor [8, 29, 30] environments. These datasets often contain numerous scale cues, such as vehicles, furniture, or man-made structures, and do not cover the diverse range of natural environments in the field. Therefore, these widely used datasets may not be fully representative of challenging field robotic environments.

The acquisition of large-scale data in-the-wild is extremely labor-intensive and generally challenging. The work in [31] is notable for employing SfM to scale up the collection of metric depth maps from publicly available images. However, due to the limited availability of field robotic datasets, these environments are scarcely represented in this pool. To overcome this, a widely adapted approach is to rely on purely synthetic datasets [32–34], which might not accurately represent the real world.

The authors of [35] propose a hybrid approach, reconstructing a 3D mesh from a set of images to render synthetic images and ground truth depth, which are blended with real images to create a realistic dataset. Yet, the images are limited to those used for the reconstruction, which restricts the dataset diversity. The authors of [36] propose a dataset for SfM in built environments, using real-world 3D meshes to render synthetic viewpoints, effectively enriching the dataset with diverse perspectives. Despite these efforts, a gap remains in datasets for MDE and DC that specifically target the field robotics domain.

III. Method

In this section, we describe our method in detail. Our primary objective is to develop a real-time capable metric DC method tailored to unseen field robotics environments. The architecture builds upon the existing SOTA MDE model, Depth Anything V2 (DAV2) [5], which originally predicts affine-invariant depth. We modify the convolutional layers

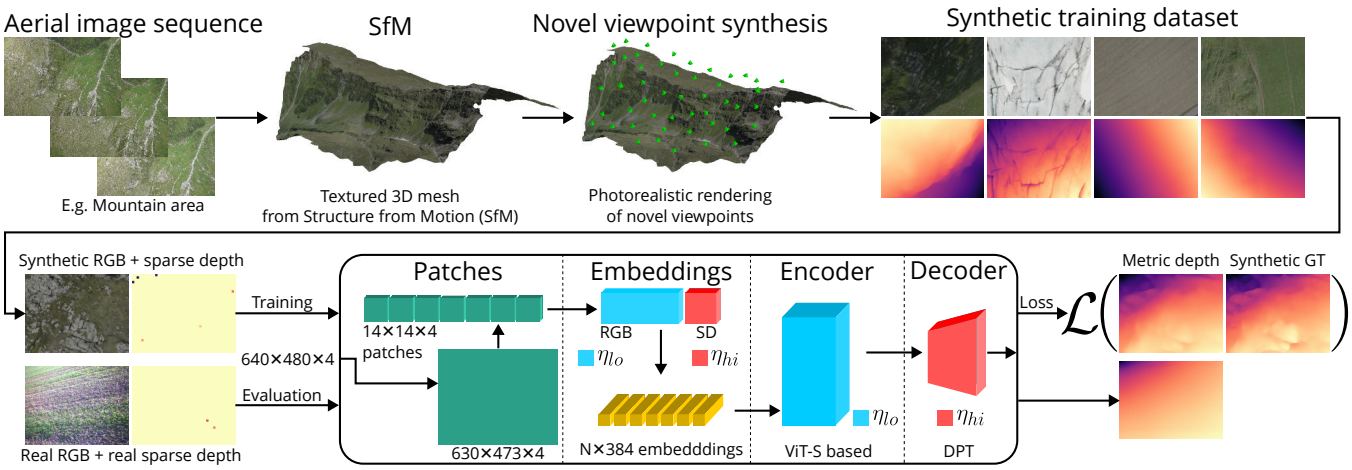

Fig. 2: From an aerial image sequence, using SfM, we obtain a textured 3D mesh of the area. With randomly sampled camera poses, we render synthetic RGB and depth ground truth for training. The input to the model is a four-channel image consisting of RGB and sparse depth (SD) information. The model operates on patches, which are transformed into an embedding vector, and through a Vision Transformer (ViT) and a dense prediction transformer (DPT) decoder, we predict metric dense depth maps.

of the pretrained encoder to accommodate a fourth input channel that represents sparse depth measurements. The newly added random weights in these layers are trained from scratch, while the pretrained weights serve as initialization for the remaining model parameters. We retrain the model on synthetic datasets derived from real-world textured 3D meshes, thereby enabling the incorporation of sparse depth measurements. Figure 2 provides an overview of the method. In the following, we detail the training dataset generation, model architecture, loss function, and the synthetic depth measurement generation used during training.

A. *Synthetic Training Dataset Generation*

The source data for the synthetic training datasets consists of multiple sequences of high-resolution RGB images captured from nadir cameras onboard Uncrewed Aerial Vehicles (UAVs). Pix4D, a commercially available SfM and photogrammetry software, is used to process the images and obtain a high-resolution, dense metric 3D mesh of the mapped region. Four specific locations are selected to obtain four relevant datasets: a mountainous area, a high-altitude alpine glacier, a road corridor, and a rural farming area. These locations cover challenging aspects of field robotics such as self-similar texture in natural environments, a wide range of scale, and few human-built structures. Such areas are not prominently represented in existing datasets, further justifying their selection. For the dataset creation, the textured 3D mesh is processed in Blender, an open-source modelling and rendering software. The goal is to generate a diverse synthetic RGB and ground truth depth dataset for training. To achieve this, we randomly generate a set of N_{frames} camera poses within the extent of the mesh in Blender. The horizontal position of the camera pose is sampled within the mesh limits, while the vertical position (along gravity) is sampled within $[z_{min}, z_{max}]$ above the mesh surface. The camera's attitude along x, and y is uniformly sampled within $[-\theta_{x,y}, \theta_{x,y}]$, while the rotation around the vertical axis is uniformly sampled over the full rotation. Additionally, Blender's sun simulation plugin is used as the main light source, randomly sampling a time of day within $[t_{min}, t_{max}]$, simulating different intensities and shadows based on the mesh. Blender then renders each camera pose, resulting in a set of high-quality, photorealistic input images and the corresponding metric depth ground truth map. Table 1 summarizes all parameters used for the dataset generation. Following this approach, the synthetic training datasets for Mountain area, Rhône glacier, Road corridor and Rural area, shown in Fig. 3, are generated.

Tab. 1: Generation parameters for the four synthetic training datasets.

Generation parameter	Value
Photogrammetry Rhône glacier	DJI Mavic Pro 12 MP
Photogrammetry all others	WingtraOne, Sony RX1RII 42 MP
N_{frames}	$4 \times 10,000$
$[z_{min}, z_{max}]$	$[1, 51]$ m
$\theta_{x,y}$	$22.5\,^\circ$
$[t_{min}, t_{max}]$	[10:00, 16:00]
Dataset resolution	640×480 px

B. *Architecture*

The proposed depth completion (DC) architecture uses the pretrained weights of the SOTA MDE model Depth Anything V2 (DAV2) [5], originally designed for three-channel RGB images and affine-invariant depth prediction. DAV2 is available in the ViT-S variant, balancing between performance and inference speed. Additionally, DAV2 is trained purely on synthetic or pseudo-labelled data, which fits well with our approach of using synthetic data for training [5].

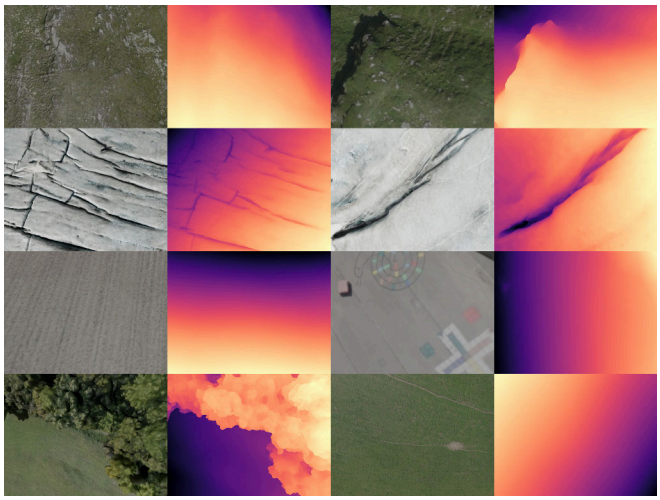

Fig. 3: From top to bottom row, samples of the Mountain area, Rhône glacier, Road corridor, and Rural area synthetic training datasets. Each pair of columns shows an RGB image and its corresponding metric depth ground truth.

In our DC approach, we encode the sparse depth measurements as a fourth input channel. Below, we present the key modifications made to the model architecture to incorporate the fourth channel, followed by a detailed description of its representation.

DAV2 operates on 14×14 pixel patches of the input image, which are transformed into an embedding vector by a 2D convolutional layer. We base our approach on the smallest ViT-S [37] variant of [5] to minimize the latency for field deployments. The embedding dimension of this variant is 384, so the pretrained patch embedding 2D convolution weight has a shape of $384 \times 3 \times 14 \times 14$. Since our approach uses four input channels, we concatenate this weight along the channel dimension with randomly initialized weights, forming a new tensor of shape $384 \times 4 \times 14 \times 14$. The fourth channel provides sparse depth information at the pixel coordinates in the input image frame. This modification enables the network to process the fourth channel while still using the initial pretrained weights as a starting point for the optimization.

Following DAV2, we predict inverse (metric) depth and, for different cameras, use the canonical image space, as detailed in [4]. For each sparse depth measurement $\mathbf{d}$, we find the corresponding 14×14 pixel patch in the input image and assign a transformed depth value $\mathbf{d}_{s,c,i}$ to the full fourth channel of the patch (visualized in Figs. 2 and 4). The transformation for the sparse depth combines the canonical transformation ($\mathbf{d}_c$) from the camera's focal length f to a canonical focal length f_c, inverts the depth ($\mathbf{d}_{c,i}$), and scales the normalized output ($\mathbf{d}_{s,c,i}$) to support the minimum depth d_{min}, since the network outputs values in $[0, 1]$:

$$\begin{aligned} \mathbf{d}_c &= \tfrac{f_c}{f}\mathbf{d} \qquad & \mathbf{d}_{c,i} &= \tfrac{1}{\mathbf{d}_c} \\ \mathbf{d}_{s,c,i} &= d_{min}\mathbf{d}_{c,i} \end{aligned} \tag{1}$$

C. Training Loss

Since the prediction target is inverse metric depth, the loss is computed with respect to the inverse metric ground truth depth, transformed analogously with Eq. (1), from the synthetic training dataset as described in Section III-A. The main loss function is the scale-invariant loss function proposed in [18]:

$$\mathcal{L}_{si}(\hat{\mathbf{d}}, \mathbf{d}) = \frac{1}{N}\sum_{i=0}^{N-1} r_i^2 - \frac{\lambda_{si}}{N^2}\left(\sum_{i=0}^{N-1} r_i\right)^2 \tag{2}$$
$$\text{where } r_i = \log \hat{d}_i - \log d_i$$

for all N pixels in the depth prediction $\hat{\mathbf{d}}$ and the ground truth depth $\mathbf{d}$. We utilize $\lambda_{si} = 0.5$, a common choice to balance metric and ordinal correctness [18]. Additionally, we add a loss term, $\mathcal{L}_{grad}$ multiplied by a scaling factor λ_{grad}, for matching gradients between the prediction and the ground truth, as proposed in [25]. This is especially effective for thin structures and sharp edges in combination with synthetic ground truths, as described in [5]. The total loss function is then defined as $\mathcal{L} = \mathcal{L}_{si} + \lambda_{grad} \cdot \mathcal{L}_{grad}$.

D. Synthetic Depth Measurements at Training Time

At training time, synthetic depth measurements are dynamically derived from the ground truth depth map to simulate the type of measurements typically available in field robotics applications. The method detects simple and efficient corner features [38] in the rendered RGB image and randomly samples between $[N_{d,min}, N_{d,max}]$ features as input. The range is selected to roughly match the expected number of depth points in extremely sparse real-world scenarios. This approach significantly increases the variability of the dataset, as the number of points and their distribution are different for each training step. The generation process is illustrated in Fig. 4.

In addition, we add noise to the sparse synthetic depth points to increase the robustness to real-world sensor noise. The noise is activated with probability p_{noise} and takes the structure of a multiplicative noise on the depth value of the ground truth. The multiplication factor is uniformly sampled between $[n_{low}, n_{high}]$.

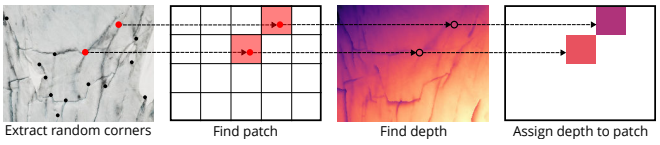

Fig. 4: At each training step, random corners are sampled from the image (e.g. two corners here), and the corresponding patches are assigned depth values based on the ground truth. The patches are enlarged in this visualization.

IV. Experiments

The experiments compare our approach against nine SOTA baselines on five real-world field robotics datasets. In particular, this section describes the characteristics, acquisition, and

generation process of these datasets, and presents the training process of our model, specific details of the baseline works, and the evaluation process.

A. Evaluation Datasets

The proposed approach is evaluated on five real-world field robotics datasets, all unseen during training: Industrial hall (IH), Farm field (FF), Glacier flight (GF), Ballast tank (BT) and Underwater fjord (UF). These datasets represent diverse and challenging field robotics environments, including indoor, outdoor, underwater and aerial scenarios. The first three datasets, obtained from dedicated field campaigns, are released with this paper; the remaining two are publicly available. Specifically, the datasets IH and FF were collected using a hand-held rig in an industrial hall and on a farm field in Switzerland. The GF dataset was collected during a UAV flight over the Rhône glacier in Switzerland. The publicly available BT and UF datasets were collected in an aerated ship ballast tank [39] and in the port of a fjord in Norway [40]. This paper also evaluates our approach on different sensor modalities: RGB or grayscale images and sparse depth measurements from a FMCW radar, as well as sparse visual landmarks from ReAQROVIO [40]. The number of measurements from this modality is comparable to that from the radar sensors (see Table 2).

Figure 1 shows representative example frames of the datasets with the sparse depth measurements overlaid in the RGB images. The extreme sparsity properties of the datasets are summarized in Table 2.

The IH dataset is characterized by a dark, low-light environment, with a repeating floor pattern. The FF dataset shows primarily vegetation and soil, with the absence of distinct human-built scale cues. The GF dataset is characterized by a dominant nadir view of the glacier, with snow, ice, crevasses and water streams. The flight spans from zero to approximately $30\,\mathrm{m}$ altitude, showing self-similar patterns in the ice and crevasses. The BT dataset is defined by a metal environment, captured in grayscale, with repeating patterns, low light and reflections, and some absolute scale features, such as manholes and ladders. The UF dataset represents the challenges of underwater perception, including limited visibility due to turbidity, overexposure from sunlight and suspended particles. The dominant features are a pier wall and the seabed.

The datasets GF and BT were collected with Ouster OS1 and OS0-64 LiDAR sensors, which are used as ground truth after reprojecting the point clouds to the image frame using known calibrations. For the remaining datasets without a LiDAR, SfM is employed to reconstruct the scene and obtain a sparse ground truth point cloud. For IH and FF, Pix4D and a DJI Mini 2 were used to collect $12\,\mathrm{MP}$ nadir images of the same area. In FF, the geolocation metadata of the nadir images provides metric scale to the reconstruction. In IH, the dimensions of known objects in the scene are measured to provide metric scale. In the UF dataset, four cameras arranged in a static rig configuration with known calibrations are used in Colmap [41] to obtain a sparse metric point cloud. Using the LiDAR or SfM point clouds and poses, the sparse ground truth depth for each frame in the evaluation datasets is created. Similarly, all sparse sensors (see Table 2) return 3D point clouds, which have been reprojected to the image frame using known calibrations. As in Section III-D, we assign the sparse depth value to the full patch corresponding to the pixel location of the measurement. The same depth value transformation as in Eq. (1) is employed.

Tab. 2: Overview of the real-world evaluation datasets: Industrial hall (IH), Farm field (FF), Glacier flight (GF), Ballast tank (BT) and Underwater fjord (UF). The column Frames refers to the number of frames in each dataset, while the column Measurements indicates the average number of depth measurements per frame and the column GT shows the average percentage of ground truth depth available.

Dataset	Sensors	Frames	Measurements	GT (%)
IH	FLIR FFY-U3-16S2C-S + TI AWR1843AOP	289	1.7	9.1
FF		321	2.5	11.2
GF		349	2.6	0.6
BT [39]	Gray + TI IWR6843AOP	299	4.3	0.7
UF [40]	Gray + visual landmarks	312	4.3	7.4

B. Training

Table 3 summarizes the hyperparameters used for our training approach. We initialize training with the ViT-S weights of DAV2 [5], pretrained for affine-invariant depth, and through our loss function, retrained for metric depth prediction. The concatenated random weights (see Section III-B) are trained with a high learning rate of η_{hi}. The DPT [44] decoder is also trained with a high learning rate, while the ViT encoder is trained with a low learning rate η_{lo}.

The model is trained purely on synthetic training datasets (as introduced in Section III-A). The training data is completely separate from the real-world evaluation datasets. In addition to the datasets shown in Table 1, we also use $10k$ frames from each of the synthetic datasets Hypersim [34]

Tab. 3: Hyperparameters used for training our MDE network.

Hyperparameter	Value
Steps, batch size	14k, 64
(η_{hi}, η_{lo})	$(5 \times 10^{-5}, 5 \times 10^{-6})$
$(\lambda_{si}, \lambda_{grad})$	$(0.5, 0.5)$
Optimizer, (β_1, β_2)	AdamW [42], $(0.9, 0.999)$
Weight decay, Gradient clipping	0.1, max. norm 1.0
LR scheduler	Linear warmup until 1k steps Constant until 2.5k steps Cosine [43] to 0.1, afterwards
Color jitter prob., intensity	80 %, [0.5, 1.5]
Gamma prob., intensity	100 %, [0.5, 1.5]
Horizontal flip prob., grayscale prob.	50 %, 20 %
Gaussian blur prob., σ	20 %, [0.1, 2.0]
f_c	900
(d_{min}, d_{max})	(0.5, 80)
$(N_{d,min}, N_{d,max})$	(1, 10)

Tab. 4: Overview of the performance of the baseline methods and our proposed approach. The table is sorted by the last column, the average rank. We highlight the best, second-best and third-best values with first, second, and third place colors. The DC column indicates whether the method is a depth completion method.

		Industrial hall		Farm field		Glacier flight		Ballast tank		Underwater fjord		
Models	DC	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	Avg. rank
DepthPrompting (NYU) [17]	✓	4.974	5.140	7.198	7.606	7.175	7.232	1.530	1.768	4.221	4.765	8.50
G2-MonoDepth [14]	✓	4.725	5.096	4.656	5.185	5.099	5.491	3.327	3.609	4.747	5.281	8.10
DAV2 [5] + LS	✓	4.239	4.883	11.877	13.445	1.163	1.466	23.669	33.143	2.000	2.666	6.90
Unik3D [6]	✗	2.611	2.738	3.552	3.724	8.018	8.031	0.821	1.018	2.700	6.085	6.40
Depth Pro [7]	✗	1.728	1.971	6.566	13.450	6.877	6.883	1.038	1.229	1.986	2.314	5.20
Metric3DV2 [4]	✗	2.156	2.256	4.185	4.465	6.393	6.404	0.883	1.087	2.346	3.088	5.20
DepthPrompting (KITTI) [17]	✓	1.273	1.591	2.768	3.250	4.907	5.280	2.117	2.427	3.944	4.340	5.10
Marigold-DC [15]	✓	2.098	2.530	3.461	4.122	0.302	0.369	1.497	1.845	2.485	3.188	**4.70**
OMNI-DC [16]	✓	1.977	2.335	2.247	2.802	0.840	1.026	1.272	1.619	2.304	2.987	**3.70**
Ours	✓	0.930	1.294	1.155	1.532	0.534	0.620	0.734	0.979	1.523	2.129	**1.20**

and Mid-Air [32] for increased diversity. We employ standard data augmentation techniques, namely color jittering, gamma changes, flipping, grayscale conversion and Gaussian blurring. The parameters for the augmentations are shown in Table 3 as well.

C. Baselines

The baseline methods are selected based on their reported performance in the literature, with a focus on generalization capabilities, as we evaluate on datasets unseen during training. We include MDE methods because they demonstrate competitive performance compared to DC methods in certain scenarios. While real-time capabilities for diffusion-based methods are a concern in field robotics, we include them due to their SOTA performance and future potential. For all methods relying on ViT architectures, the small ViT-S variant [37] is used, if supported. The Depth Pro [7] baseline uses a ViT-L encoder, as a ViT-S version is not released. Following the proposal in [15], we perform a LS fit on all sparse depth measurements to lift the affine-invariant prediction of DAV2 to the metric domain. This baseline is denoted as DAV2+LS. Additionally, the Metric3DV2 [4] method requires knowledge on the camera intrinsics, which we provide. DepthPrompting provides two variants, trained on NYU-v2 and KITTI [17]. We evaluate both variants on our field robotics datasets.

All baselines, and our approach, are evaluated in the same manner and with the same data, using the publicly available pretrained weights. All evaluation datasets are rescaled to an intermediate resolution of 640×480 pixels, and evaluated at this resolution.

V. Results

The obtained comparison results are presented in Table 4. The evaluation metrics are Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), following [15], and are averaged over all frames in each dataset. The table is sorted by the average rank across all metrics and datasets. Additionally, Fig. 1 provides qualitative results that illustrate the visual quality of the depth predictions.

At a high level, our approach achieves the highest average rank, demonstrating its effectiveness in adapting to unseen, real-world environments and noisy depth data. In the individual experiments, our approach performs consistently, achieving the lowest error metrics except in the Glacier flight experiment. The overall second- and third-ranked methods do not consistently achieve the second and third rank in each individual experiment. These observations suggest that our approach is more robust, and the dedicated design choices in model architecture and training protocol are effective in these scenarios.

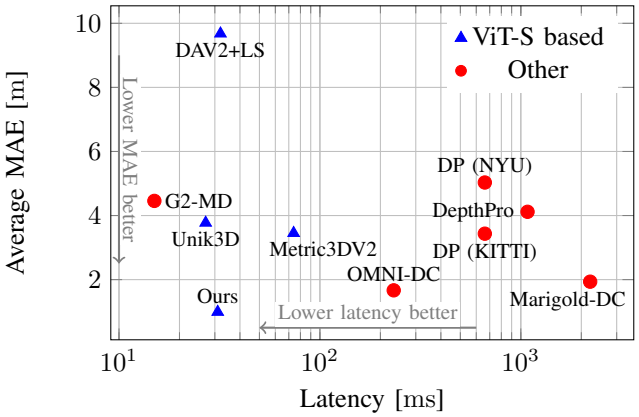

Fig. 5: Latency, defined as the total time required to process input and generate output on an Nvidia 3090 GPU, plotted against the average MAE across all experiments.

The Ballast tank experiment is particularly challenging, as the radar measurements are quite noisy due to the metal environment; this may explain the low performance of DAV2+LS, where fitting the sparse depth data to the dense affine-invariant depth yields very high errors. All DC methods (except our method) are outperformed by the MDE methods, further supporting this hypothesis. Similarly, in the Underwater fjord experiment, the second-best method is a MDE method. This suggests that noisy and extremely sparse depth data can have adverse effects on the performance of DC methods when treating the problem purely as a regression or in-painting task.

In the Glacier flight scenario, Marigold-DC [15] achieves a lower MAE and RMSE. This scenario is challenging

due to the nadir-view perspective, which does not provide strong clues on the orientation of the glacier. Marigold-DC appears to predict more planar depth maps that are aligned more accurately with the depth measurements, explaining the higher performance.

For practical deployment in the field, we highlight three key aspects: inference time, dataset size, and the use of synthetic data. The second- and third-ranked methods are likely not suited for real-time applications without further optimisation (see Fig. 5). We also conduct an additional inference time analysis of our method on an Nvidia Jetson AGX Orin platform for relative comparison between a desktop GPU and this embedded device. With an inference time of $53\,\mathrm{ms}$ on the embedded device, our approach is suitable for real-time onboard computation in many mobile robotics applications. By partially re-using pretrained weights of DAV2, our approach utilizes a relatively small dataset size ($60k$ ours versus $573k$ for OMNI-DC), and a short training time (approximately $7\,\mathrm{hours}$ on a single Nvidia 3090 GPU) until convergence. Lastly, training on a diverse synthetic dataset with simulated depth sensor data, all derived from real-world textured 3D meshes, suggests a promising direction, with our approach generalizing well to the real-world experiments.

VI. Conclusions

This paper presents a novel approach for depth completion (DC) with extremely sparse depth measurements, a common scenario in challenging field robotics applications. The model enables the combination of a monocular camera and sparse depth sensors as a viable dense metric depth sensor modality. Our approach achieves the highest average rank across all investigated unseen field robotics evaluation datasets, demonstrating the effectiveness of the proposed methodology across different scenarios while also keeping inference time low for real-time applicability. In addition to a DC method, this work also proposes a method for generating large amounts of synthetic training data for field robotics applications and releases relevant field robotic datasets. This contribution is expected to significantly reduce manual data collection and simultaneously address challenges related to incomplete or noisy depth ground truth. While the proposed method is specifically designed for extremely sparse depth measurements, it does not generalize well to very dense depth measurements due to the specialized depth input space and training process. Addressing this limitation to support any depth sensor density could be explored in future work.

References

[1] C. Cadena *et al.*, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] M. Nissov, N. Khedekar, and K. Alexis, "Degradation resilient lidar-radar-inertial odometry," in *IEEE International Conference on Robotics and Automation*, 2024, pp. 8587–8594.

[3] B. Ke, A. Obukhov, S. Huang, N. Metzger, R. C. Daudt, and K. Schindler, "Repurposing diffusion-based image generators for monocular depth estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9492–9502.

[4] M. 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*, vol. 46, no. 12, pp. 10 579–10 596, 2024.

[5] L. Yang *et al.*, "Depth anything v2," in *Advances in Neural Information Processing Systems*, vol. 37, 2024, pp. 21 875–21 911. DOI: 10.52202/079017-0688.

[6] L. Piccinelli *et al.*, "UniK3D: Universal camera monocular 3d estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[7] A. Bochkovskiy *et al.*, "Depth pro: Sharp monocular metric depth in less than a second," in *International Conference on Representation Learning*, 2025, pp. 75 602–75 637.

[8] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant cnns," in *International Conference on 3D Vision*, 2017, pp. 11–20.

[9] F. Ma and S. Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," in *IEEE International Conference on Robotics and Automation*, 2018, pp. 4796–4803.

[10] J. Qiu *et al.*, "Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[11] J.-T. Lin, D. Dai, and L. V. Gool, "Depth estimation from monocular images and sparse radar data," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 10 233–10 240.

[12] Y. Long, D. Morris, X. Liu, M. Castro, P. Chakravarty, and P. Narayanan, "Radar-camera pixel depth association for depth completion," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 502–12 511.

[13] Z. Feng, L. Jing, P. Yin, Y. Tian, and B. Li, "Advancing self-supervised monocular depth learning with sparse lidar," in *Conference on Robot Learning*, PMLR, 2022, pp. 685–694.

[14] H. Wang, M. Yang, and N. Zheng, "G2-monodepth: A general framework of generalized depth inference from monocular rgb+x data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 3753–3771, 2024.

[15] M. Viola *et al.*, "Marigold-dc: Zero-shot monocular depth completion with guided diffusion," in *IEEE/CVF International Conference on Computer Vision*, 2025.

[16] Y. Zuo, W. Yang, Z. Ma, and J. Deng, "Omni-dc: Highly robust depth completion with multiresolution depth integration," in *IEEE/CVF International Conference on Computer Vision*, 2025.

[17] J.-H. Park, C. Jeong, J. Lee, and H.-G. Jeon, "Depth prompting for sensor-agnostic depth estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9859–9869.

[18] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[19] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Fourth International Conference on 3D Vision*, 2016, pp. 239–248.

[20] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2024–2039, 2016.

[21] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep Ordinal Regression Network for Monocular Depth Estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[22] G. Yang, H. Tang, M. Ding, N. Sebe, and E. Ricci, "Transformer-based attention networks for continuous pixel-wise prediction," in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 16 249–16 259.

[23] S. F. Bhat, I. Alhashim, and P. Wonka, "Adabins: Depth estimation using adaptive bins," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4009–4018.

[24] S. Saxena *et al.*, "The surprising effectiveness of diffusion models for optical flow and monocular depth estimation," in *International Conference on Neural Information Processing Systems*, vol. 37, Curran Associates Inc., 2023.

[25] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, pp. 1623–1637, 2022.

[26] R. Wang *et al.*, "Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 5261–5271.

[27] S. F. Bhat, R. Birkl, D. Wofk, P. Wonka, and M. Müller, "Zoedepth: Zero-shot transfer by combining relative and metric depth," *CoRR*, vol. abs/2302.12288, 2023.

[28] V. Guizilini, I. Vasiljevic, D. Chen, R. Ambruș, and A. Gaidon, "Towards zero-shot scale-aware monocular depth estimation," in *IEEE/CVF International Conference on Computer Vision*, 2023, pp. 9199–9209.

[29] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision*, 2012, pp. 746–760.

[30] I. Vasiljevic *et al.*, "DIODE: A Dense Indoor and Outdoor DEpth Dataset," *CoRR*, vol. abs/1908.00463, 2019.

[31] Z. Li and N. Snavely, "Megadepth: Learning single-view depth prediction from internet photos," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2041–2050.

[32] M. Fonder and M. Van Droogenbroeck, "Mid-air: A multi-modal dataset for extremely low altitude drone flights," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 553–562.

[33] W. Wang *et al.*, "Tartanair: A dataset to push the limits of visual slam," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020.

[34] M. Roberts *et al.*, "Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding," in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 892–10 902.

[35] Y. Yao *et al.*, "Blendedmvs: A large-scale dataset for generalized multi-view stereo networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2020.

[36] D. Marelli, L. Morelli, E. M. Farella, S. Bianco, G. Ciocca, and F. Remondino, "Enrich: Multi-purpose dataset for benchmarking in computer vision and photogrammetry," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 198, pp. 84–98, 2023.

[37] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 139, PMLR, 2021, pp. 10 347–10 357.

[38] J. Shi and Tomasi, "Good features to track," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600.

[39] *Ballast Water Tank Dataset*, https://github.com/ntnu-arl/ballast_water_tank_dataset, Accessed: 2025-08-18, Mar. 2024.

[40] M. Singh and K. Alexis, "Online refractive camera model calibration in visual inertial odometry," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2024, pp. 12 609–12 616.

[41] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition*, 2016.

[42] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2017.

[43] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *International Conference on Learning Representations*, 2017.

[44] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 179–12 188.