Gaussian Belief Propagation Network for Depth Completion

Jie Tang[1⋆], Pingping Xie[1], Jian Li[1], and Ping Tan[2]

[1] National University of Defense Technology, Changsha, China
[2] Hong Kong University of Science and Technology, Hong Kong, China
tangjiephd@gmail.com, {pingpingxie, lijian}@nudt.edu.cn, pingtan@ust.hk

Abstract. Depth completion aims to predict a dense depth map from a color image with sparse depth measurements. Although deep learning methods have achieved state-of-the-art (SOTA), effectively handling the sparse and irregular nature of input depth data in deep networks remains a significant challenge, often limiting performance, especially under high sparsity. To overcome this limitation, we introduce the Gaussian Belief Propagation Network (GBPN), a novel hybrid framework synergistically integrating deep learning with probabilistic graphical models for end-to-end depth completion. Specifically, a scene-specific Markov Random Field (MRF) is dynamically constructed by the Graphical Model Construction Network (GMCN), and then inferred via Gaussian Belief Propagation (GBP) to yield the dense depth distribution. Crucially, the GMCN learns to construct not only the data-dependent potentials of MRF but also its structure by predicting adaptive non-local edges, enabling the capture of complex, long-range spatial dependencies. Furthermore, we enhance GBP with a serial & parallel message passing scheme, designed for effective information propagation, particularly from sparse measurements. Extensive experiments demonstrate that GBPN achieves SOTA performance on the NYUv2 and KITTI benchmarks. Evaluations across varying sparsity levels, sparsity patterns, and datasets highlight GBPN's superior performance, notable robustness, and generalizable capability.

1 Introduction

Dense depth estimation is critical for various computer vision and robotics tasks. While dedicated sensors or methods provide sparse and irregular depth measurements [34], acquiring dense depth directly is often challenging or costly. Depth completion (DC) bridges this gap through inferring a dense depth map from sparse depth guided by a synchronized color image. Sparse depth provides essential scale and absolute constraints, while the high-resolution color image offers rich structural and semantic information necessary for propagating depth and filling missing regions. Consequently, DC is a vital technique attracting increasing interest for enhancing downstream applications like 3D object detection [44], novel view synthesis [31], and robotic manipulation [21].

⋆ indicates the corresponding author.

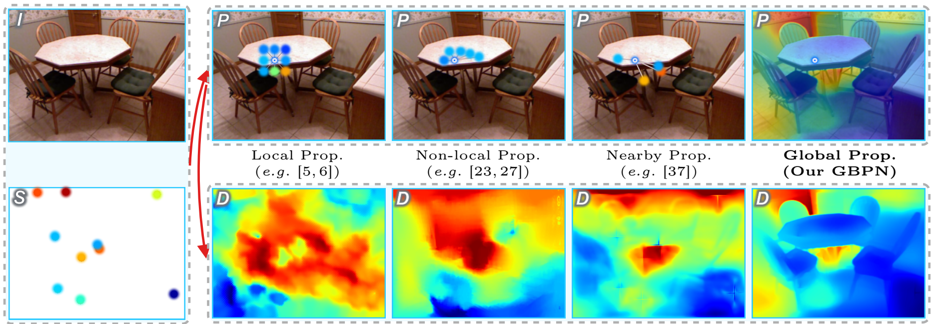

Fig. 1: Learning-based DC methods typically struggle to process sparse data (S) with limited propagation range (P), and degrade dramatically under high sparsity. Our GBPN addresses these limitations with full-image propagation range, achieving visually clearer depth estimation (D) even under extreme sparsity (down to a single point shown in Fig. 4 and Appendix Fig. 8), while all methods are trained with 500 random points.

Traditional DC methods mostly relied on hand-crafted pipelines [20] or fixed graphical models like Markov Random Field (MRF) [10]. Although offering some robustness, their rigid, pre-defined priors struggled to capture complex geometry and fine details. Recently, learning-based methods, primarily deep neural networks, have achieved impressive accuracy gains by directly regressing dense depth from extracted features [25]. However, effectively processing sparse and irregular input within standard deep architectures remains a significant challenge [17, 37, 39], often degrading performance and robustness, particularly under high sparsity.

In this paper, we introduce the Gaussian Belief Propagation Network (GBPN), a novel hybrid framework that synergistically combines the representational ability of deep learning with the structured inference of probabilistic graphical models. Unlike methods that directly regress depth, GBPN trains a deep network (the Graphical Model Construction Network, GMCN) to dynamically construct a scene-specific MRF over dense depth variables. Dense depth is then efficiently inferred via Gaussian Belief Propagation (GBP) on the learned MRF.

This formulation offers several key advantages. Firstly, constructing a scene-specific MRF overcomes the limitations of fixed, hand-crafted models, enabling adaptation to diverse geometries. Secondly, sparse depth measurements are naturally integrated as principled data terms within the globally consistent MRF framework, inherently addressing input sparsity and irregularity by propagating depth across the entire image, as illustrated in Fig. 1. Finally, GBPN yields a depth distribution, providing valuable confidence estimation for risk-aware downstream tasks, such as planning [2].

To realize this, our GMCN infers not only the potentials of the MRF but also its structure by predicting non-local edges, allowing the model to adaptively capture complex, long-range spatial dependencies guided by image content. We also propose a novel serial & parallel message passing scheme for GBP to enhance information flow, particularly from sparse measurements to distant unmeasured pixels. The entire GBPN is trained end-to-end using a probability-based loss function leveraging the estimated mean and precision from GBP, promoting the learning of reliable depth predictions along with an estimate of their confidence.

We validate the efficacy of GBPN through extensive experiments on two leading DC benchmarks: NYUv2 for indoor scenes and KITTI for outdoor scenes. Our method achieves state-of-the-art performance on these datasets at the time of submission. Comprehensive ablation studies demonstrate the effectiveness of each component within GBPN, including the dynamic MRF construction, propagation scheme and probability-based loss function, *etc*. Furthermore, evaluations across varying sparsity levels, sparsity patterns, and datasets highlight GBPN's superior performance, notable robustness, and generalizable capability. Code and trained models will be available at `https://github.com/kakaxi314/GBPN`.

2 Related Work

Research on depth completion (DC) has undergone a significant evolution, transitioning from traditional hand-crafted techniques to data-driven learning methods. Notably, concepts and principles from traditional approaches have significantly influenced the design of recent learning-based methods. This section briefly reviews these two lines of work and the combination of their respective strengths.

Traditional Hand-crafted Techniques. Early DC approaches largely relied on explicitly defined priors and assumptions about scene geometry and texture, often borrowing techniques from traditional image processing and inpainting. These methods constructed hand-crafted pipelines to infer dense depth from sparse measurements. For example, Kopf *et al.* [19] proposed joint bilateral filtering guided by color information to upsample low-resolution depth maps. Ku *et al.* [20] employed a sequence of classical image processing operators, such as dilation and hole filling, for depth map densification. Zhao *et al.* [48] leveraged local surface smoothness assumptions to estimate depth by computing surface normals in a spherical coordinate system. Hawe *et al.* [15] reconstructed dense disparity from sparse measurements by minimizing an energy function formulated based on Compressive Sensing. Diebel and Thrun [10] modeled depth completion as a multi-resolution MRF with simple smoothness potentials, and then solved using conjugate gradient optimization. Chen and Koltun [3] developed a global optimization approach to reconstruct dense depth modeled by MRF. While demonstrating efficiency in certain scenarios, these hand-crafted methods often struggle to capture complex geometric structures and fine-grained details.

Data-driven Learning Methods. In contrast, learning-based methods typically formulate DC as an end-to-end regression task, taking color images and sparse depth maps as inputs and directly predicting the target dense depth map. A primary focus in this line of work has been on effectively handling the sparse input data and fusing information from different modalities (color and depth). For explicitly processing sparse data, pioneering work by Uhrig *et al.* [39] introduced sparsity-invariant convolutions, designed to handle missing data by maintaining validity masks. Huang *et al.* [17] integrated the sparsity-invariant convolutions into an encoder-decoder architecture. Eldesokey *et al.* [11] extended this concept by propagating a continuous confidence measure. However, general-purpose network architectures employing standard convolutional layers, combined with sophisticated multi-modal fusion strategies, have frequently demonstrated stronger performance. For instance, Tang *et al.* [38] proposed to learn content-dependent and spatially-variant kernels to guide the fusion of color and depth features. Zhang *et al.* [46] leveraged Transformer architectures to capture long-range dependencies for feature extraction in depth completion. Chen *et al.* [4] used 2D convolutions for image features and continuous convolutions for 3D point cloud features, followed by fusion. While learning complex mappings from data with impressive results, these methods often exhibit limited generalization performance, particularly when faced with data of different sparsity levels.

Combination of Models and Learning. More recently, a significant trend has emerged towards approaches that combine the strengths of both traditional modeling and data-driven learning. These methods aim to leverage the benefits of learned features and powerful network architectures while incorporating explicit priors or structured inference mechanisms. One pioneering work is Liu *et al.* [24], combining a deep network with a closed-form solver for monocular depth estimation. For depth completion, a prominent line of work, exemplified by CSPN-based methods [5, 6, 23, 27], employs deep networks to predict parameters for a learned anisotropic diffusion process, refining the regressed depth from a deep network. Wang *et al.* [40] integrated traditional image processing techniques to generate an initial depth map before applying a learned refinement module. Tang *et al.* [37] learned a bilateral filter-like propagation process to effectively spread information from sparse measurements. Qu *et al.* [30] combined deep learning with a least-squares solver to estimate depth with constraints derived from sparse measurements. Zuo and Deng [50] formulated depth completion as a learned optimization problem, where a network predicts local depth differences used in an energy function iteratively minimized via conjugate gradient. However, these methods still struggle with sparse data processing and limited propagation range. Our method falls within this hybrid category, learning an MRF and inferring it via GBP, inherently addressing the issues of input sparsity and irregularity.

3 The proposed method

3.1 Overview

Given a color image $I \in \mathbb{R}^{H \times W \times 3}$, regardless of how sparse depth is measured—whether via active sensor, like LiDAR [13], or passive method, like SFM [34], or even interactive user guidance [32], we can project these depth measurements onto the image plane to yield a sparse depth map $S \in \mathbb{R}^{H \times W}$, with the same resolution (H, W). The valid pixels in S are typically irregularly distributed, and may vary significantly in number and location. Unlike most learning-based approaches that employ dedicated neural network layers to process S [11, 17]

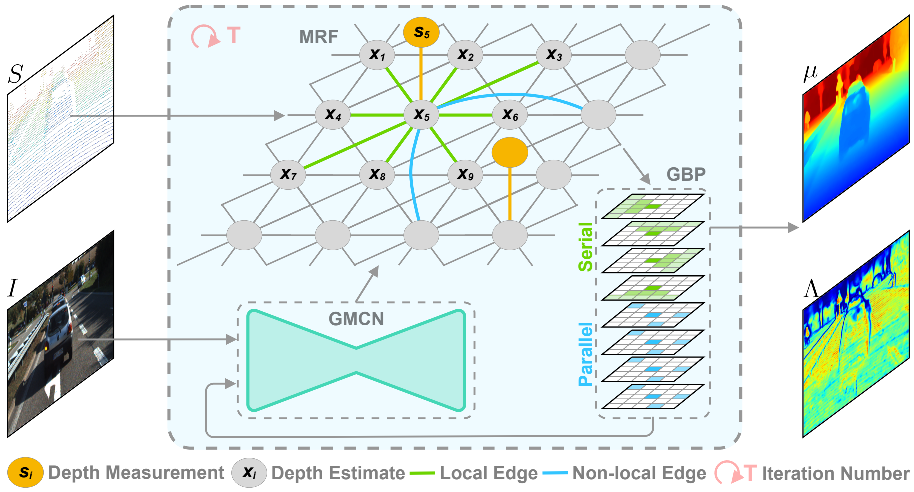

Fig. 2: Overview of the proposed approach. Markov Random Field (MRF) is constructed depending on parameters dynamically generated from the Graphical Model Construction Network (GMCN), and then optimized via Gaussian Belief Propagation (GBP) for the distribution of dense depth map.

and directly regress the dense depth map $X \in \mathbb{R}^{H \times W}$, as illustrated in Fig. 2, we formulate the dense depth estimation task as inference in a Markov Random Field (MRF) (in Sec. 3.2), and infer X via Gaussian Belief Propagation (GBP) (in Sec. 3.3). In this formulation, S serves as the data term within the global optimization framework of MRF, eliminating the need for designing specific neural network architectures tailored to processing sparse input data [37]. Specifically, the MRF structure, particularly its edges, is dynamically generated by a graphical model construction network (in Sec. 3.4), depending on the input color image I and optionally on intermediate estimates of the dense depth distribution. The entire framework is end-to-end trained with a probability-based loss function (in Sec. 3.5).

3.2 Problem formulation

We formulate the dense depth estimation problem using a Markov Random Field (MRF), *i.e.* a type of undirected graphical model $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes representing the image pixels and $\mathcal{E}$ is the set of pairwise edges connecting neighboring pixels. As shown in Fig. 2, each node $i \in \mathcal{V}$ is associated with a random variable x_i, representing the depth at pixel i. The joint probability distribution over the depth variables is defined according to the MRF structure:

$$p(X|I,S) \propto \prod_{i \in \mathcal{V}_v} \phi_i \prod_{(i,j) \in \mathcal{E}} \psi_{ij}, \tag{1}$$

where ϕ_i and ψ_{ij} are the abbreviations for $\phi_i(x_i)$ and $\psi_{ij}(x_i, x_j)$, indicating unary and pairwise potentials respectively, and $\mathcal{V}_v$ is the set of nodes corresponding to pixels with a valid depth measurement in the sparse map S.

The *unary potential* ϕ_i defined for pixel $i \in \mathcal{V}_v$, constrains the quadratic distance between estimated depth x_i and the depth measurement s_i, written as

$$\phi_i = \exp(-\frac{w_i(x_i - s_i)^2}{2}). \tag{2}$$

Here, $w_i > 0$ is a weight indicating the confidence for the depth measurement, and ϕ_i is only available if depth measurement s_i is valid.

The *pairwise potential* ψ_{ij} defined for edge $(i, j) \in \mathcal{E}$, encourages spatial coherence between the estimated depths of neighboring pixels i and j, expressed as

$$\psi_{ij} = \exp(-\frac{w_{ij}(x_i - x_j - r_{ij})^2}{2}). \tag{3}$$

Here, $w_{ij} > 0$ is a weight controlling the strength of the spatial constraint, and r_{ij} is the expected depth difference between x_i and x_j .

In traditional MRF-based approaches, the parameters w and r are typically hand-crafted based on simple assumptions or features. For instance, r_{ij} is often set to 0 to enforce simple smoothness [10], and w_{ij} is commonly derived from local image features like color differences. These approaches, while providing some robustness, often limit the model's ability to capture complex scene geometry and fine details, leading to suboptimal results. In contrast, we employ an end-to-end trained deep network to dynamically construct the MRF based on the color image and optional optimized intermediate dense depth map distribution. This allows the MRF to adaptively model the scene, via placing stronger data constraints where measurements seem reliable, and applying sophisticated smoothness constraints that vary based on image content. Moreover, we further enhance the MRF's expressive power with *dynamic parameters* and *dynamic edges*.

MRF with Dynamic Parameters. The constructed MRF is approximately inferred in an iterative manner. Unlike traditional methods that define the MRF parameters a priori and keep them fixed during optimization, our approach updates the MRF parameters dynamically as the iterative inference progresses. This adaptive parameterization allows the MRF to evolve alongside the solution, making it more responsive to the current state of the variables and potentially improving both the solution quality and the convergence behavior of the optimization.

MRF with Dynamic Edges. Traditionally, the graph structure of an MRF, specifically the edges connecting a variable to its neighbors, is fixed and predefined, *e.g.* 8-connected local pixels for each variable. As illustrated in Fig. 2, alongside these fixed local edges (depicted in green), our framework also dynamically estimates and establishes non-local connections for each pixel (shown in blue). This dynamic graph structure allows the MRF to capture dependencies beyond immediate spatial neighbors, increasing its flexibility and capacity to model complex scene structures. Consequently, during inference, each variable's state is conditioned on both its predefined local neighbors and its dynamically generated non-local connections.

3.3 Approximate Inference Technology

Gaussian Belief Propagation. For a high-resolution image, computing the exact posterior for the above MRF is computationally prohibitive, due to the large number of variables and complex dependencies. Belief Propagation (BP) [28] is a widely used algorithm for performing approximate probabilistic inference in graphical models. By iteratively operating on belief updating and message passing, BP enables parallel computation and often exhibits reasonably fast empirical convergence compared to other iterative inference methods [42]. Though lacking formal convergence guarantees for general graphs, it has been successfully applied to various graphs, even with cycles [12,36], known as loopy belief propagation. As introduced below, we adopt damping tricks and graph decomposition to improve the stability and convergence.

In BP, the belief on variable x_i is proportional to the product of its local evidence ϕ_i and incoming messages from all neighboring variables, written as

$$b_i \propto \phi_i \prod_{(i,j)\in\mathcal{E}} m_{j\to i}. \tag{4}$$

Messages are computed by marginalizing over the variable of the sending node, considering its local evidence and incoming messages from its other neighbors, expressed as

$$m_{j\to i} \propto \int_{x_j} \psi_{ij}\phi_j \prod_{(k,j)\in\mathcal{E}\setminus(i,j)} m_{k\to j}\, dx_j. \tag{5}$$

The formulated MRF in Sec. 3.2 is a Gaussian graph model, for which we can use Gaussian Belief Propagation (GBP) for inference. GBP [1] assumes all beliefs b and messages m are under Gaussian distribution. This assumption allows the complex integration and product operations in Eq. (5) to be simplified into algebraic updates on the parameters of the Gaussian distributions. We can take moment form $\mathcal{N}(\mu, \Lambda^{-1})$ or canonical form $\mathcal{N}^{-1}(\eta, \Lambda)$ as Gaussian parametrization for convenience, where $\eta = \mu\Lambda$. Then, in GBP, the belief updating corresponding to Eq. (4) is given by:

$$\begin{aligned} \eta_i &= w_i s_i + \sum_{(i,j)\in\mathcal{E}} \hat{\eta}_{j\to i}, \\ \Lambda_i &= w_i + \sum_{(i,j)\in\mathcal{E}} \hat{\Lambda}_{j\to i}. \end{aligned} \tag{6}$$

And message passing corresponding to Eq. (5) is given by:

$$\begin{aligned} \mu_{j\to i} &= \mu_{j\setminus i} + r_{ij}, \\ \Lambda^{-1}_{j\to i} &= \Lambda^{-1}_{j\setminus i} + w^{-1}_{ij}. \end{aligned} \tag{7}$$

Here, $j \setminus i$ means the set of all messages sent to j except the message from i. We leave the detailed derivation of Gaussian belief propagation in Appendix A.1. To improve convergence and stability, we adopt the *damping* trick for message passing [26] by applying weighted average of the message from the previous iteration and the message computed in the current iteration, *i.e.* $\hat{m}_{j\to i,t} = \hat{m}^{\beta_i}_{j\to i,t-1} m^{1-\beta_i}_{j\to i,t}$. Once again corresponding to Gaussian parameters update, given by:

$$\begin{aligned} \hat{\eta}_{j\to i,t} &= \beta_i \hat{\eta}_{j\to i,t-1} + (1-\beta_i)\eta_{j\to i,t}, \\ \hat{\Lambda}_{j\to i,t} &= \beta_i \hat{\Lambda}_{j\to i,t-1} + (1-\beta_i)\Lambda_{j\to i,t}. \end{aligned} \tag{8}$$

Propagation Scheme. A crucial aspect of Belief Propagation is the definition of its message propagation scheme [9]. Both serial and parallel propagation schemes offer distinct advantages. Serial propagation is effective at propagating information across the entire graph structure, allowing local evidence to influence distant nodes. This is particularly important for depth completion, ensuring that information from depth measurements propagates broadly, resulting in a dense depth estimation where every variable receives sufficient incoming messages to form a valid belief. In contrast, parallel propagation, propagating messages over short ranges, is significantly more efficient by effectively leveraging modern hardware parallelism.

To combine both advantages, we design a hybrid serial & parallel propagation scheme. As illustrated in Fig. 2, our scheme splits the message passing into updates performed serially on directional local connections and updates performed in parallel on non-local connections, decomposing the loopy graph into loop-free subgraphs. Specifically, we categorize edges $\mathcal{E}$ in MRF into four sets corresponding to local directional sweeps: left-to-right (LR), top-to-bottom (TB), right-to-left (RL), and bottom-to-top (BT), denoted $\mathcal{E}_{LR}$, $\mathcal{E}_{TB}$, $\mathcal{E}_{RL}$, and $\mathcal{E}_{BT}$ respectively, and a set for non-local connections, $\mathcal{E}_{NL}$. With this decomposition, each set of edges is with directionality and loop-free, which is more stable for convergence.

For serial propagation, we sequentially perform message passing sweeps utilizing these four local directional edge sets. Under each sweep direction, message and belief updates are processed serially according to a defined order (e.g., column by column for horizontal sweeps). For instance, during the sweep using edges $\mathcal{E}_{LR}$ (left-to-right), message updates for variables in column n are computed only after updates related to column $n-1$ have been completed. And the update for a variable at pixel (m,n) incorporates messages received from specific neighbors in column $n-1$, such as those connected via edges from $\mathcal{E}_{LR}$ linking $(m-1,n-1)$, $(m,n-1)$, and $(m+1,n-1)$ to (m,n). For non-local propagation, message updates for all pixels based on connections in $\mathcal{E}_{NL}$ are computed simultaneously. These non-local connections are dynamically generated to link pixels that are relevant but not necessarily spatially adjacent.

The whole propagation scheme is detailed in Algorithm 1. We initialize the η and Λ parameters for all messages on all relevant edges to 0. We empirically set a fixed total number of iterations T. In each iteration, we first sequentially perform the four directional local serial propagation sweeps (LR, TB, RL, BT). Following the serial sweeps, we execute T_n steps of parallel non-local propagation. Finally, after T iterations, we yield the marginal beliefs for each pixel i. The estimated depth is the mean $\mu_i = \eta_i / \Lambda_i$ with the precision Λ_i serving as a measure of confidence. The visualization of propagation and learned edges is introduced in Appendix A.4.

Algorithm 1 Serial & Parallel Propagation Scheme

Require: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Edge subsets $\mathcal{E}_{LR}, \mathcal{E}_{TB}, \mathcal{E}_{RL}, \mathcal{E}_{BT}, \mathcal{E}_{NL}$.
Ensure: Estimated marginal belief represented by η, Λ.

Initialization:
for all edges $(i, j) \in \mathcal{E}$ **do** ▷ Initialize messages
  $\eta_{j \to i} = 0$
  $\Lambda_{j \to i} = 0$
end for
Iterations:
for $t = 1$ to T **do**
  for all $\mathcal{E}_{LR}, \mathcal{E}_{TB}, \mathcal{E}_{RL}, \mathcal{E}_{BT}$ **do** ▷ Serial Propagation
    Serial message passing(Eqs. (7) and (8)) and belief updating(Eq. (6))
  end for
  for $t_n = 1$ to T_n **do** ▷ Parallel Propagation
    message passing (Eqs. (7) and (8)) for all $(i, j) \in \mathcal{E}_{NL}$ in parallel
    belief updating (Eq. (6))
  end for
end for
return η, Λ

3.4 Graphical Model Construction Network

As previously mentioned, the Markov Random Field (MRF) is dynamically constructed and subsequently optimized using Gaussian Belief Propagation (GBP). Our Graphical Model Construction Network is designed to learn the parameters and structures of this MRF from input data.

We employ a U-Net architecture [33], comprising encoder and decoder layers, to extract multi-scale features essential for MRF construction. Within the U-Net, we introduce a novel global-local processing unit. Each unit combines a dilated neighborhood attention layer [14] and a ResNet block [16]. The dilated neighborhood attention layer is utilized to capture long-range dependencies, effectively expanding the receptive field without increasing computational cost. As a complement, the ResNet block focuses on extracting and refining local features. More details about the network architecture are introduced in Appendix A.2.

We apply convolutional layers on the aggregated features to estimate the MRF parameters, such as r and w. Additionally, the network estimates the damping rate β used in the GBP and the offsets for constructing non-local neighboring pixels. Inspired by Deformable Convolutional Networks [8], we employ bilinear interpolation to sample Gaussian parameters at these non-local neighbor locations defined by the estimated float offsets. This approach ensures that gradients can be effectively backpropagated during the training phase. Considering that GBP typically converges to an accurate mean μ but not the exact precision Λ [42], we also utilize convolution layers to estimate a residual term. This residual is added to the estimated Λ from GBP, and a sigmoid activation function is then applied to yield the updated precision.

We provide two approaches to construct the MRF: a color image-only approach (termed GBPN-1) and a multi-modal fusion approach (termed GBPN-2) incorporating depth information. In GBPN-1, the U-Net architecture receives a color image as its sole input to construct the MRF. The GBPN-2 utilizes a second U-Net that takes both the color image and the optimized depth distribution from the GBPN-1 (*i.e.* μ and Λ) as input. Both take 3D position encoding derived from pixel coordinates with camera intrinsic as an additional input to enhance spatial awareness. For GBPN-2, cross-attention is leveraged within the dilated attention layers, where the query is generated from multi-modal features, while the keys and values are derived from the color image features in the first U-Net.

3.5 Probability-based Loss Function

We adopt the combination of L_1 and L_2 loss as the loss on depth. For a pixel i, the loss on depth is

$$L_i^X = \frac{\|\mu_i - x_i^g\|_2^2 + \alpha \|\mu_i - x_i^g\|_1}{max(\|\mu - x^g\|_1)}, \tag{9}$$

where x_i^g is the ground-truth depth, and α is a hyperparameter to balance the L_1 and L_2 loss. The combined L_1 and L_2 loss is normalized with the maximum L_1 loss across the image, which makes the convergence more stable. As the output of our framework is the distribution of dense depth map, which is in Gaussian and can be represented with μ_i and Λ_i for each pixel, we adopt the probability-based loss [18] as the final loss:

$$L = \frac{1}{|\mathcal{V}_g|} \sum_{i \in \mathcal{V}_g} \Lambda_i L_i^X - \log(\Lambda_i). \tag{10}$$

Here, $\mathcal{V}_g$ denotes the set of pixels with valid ground-truth depth. In this way, the precision Λ can also be directly supervised, without the need for its ground-truth.

4 Experiments

We evaluate our proposed method, GBPN, on two leading depth completion benchmarks: NYUv2 [35] for indoor scenes and KITTI [13] for outdoor scenes.

To demonstrate its effectiveness, we provide comprehensive comparisons against state-of-the-art (SOTA) methods in Sec. 4.1. We then perform thorough ablation studies in Sec. 4.2 to analyze the contribution of core components. We also assess the robustness of GBPN by varying input sparsity levels in Sec. 4.3, a critical factor for real-world deployment. Finally, cross-dataset evaluations in Sec. 4.4 verify the model's generalization capability. The appendix contains further analysis on the experimental setup (Appendix A.3), noise sensitivity (Appendix A.7), and runtime efficiency (Appendix A.8).

Table 1: Performance on KITTI and NYUv2 datasets. For the KITTI dataset, results are evaluated by the KITTI testing server. For the NYUv2 dataset, authors report their results in their papers. The best result under each criterion is in **bold**. The second best is with <u>underline</u>.

	KITTI				NYUv2			
	RMSE↓ (mm)	MAE↓ (mm)	iRMSE↓ ($1/km$)	iMAE↓ ($1/km$)	RMSE↓ (m)	REL↓	$\delta_{1.02}\uparrow$ (%)	$\delta_{1.05}\uparrow$ (%)
S2D [25]	814.73	249.95	2.80	1.21	0.230	0.044	–	–
DeepLiDAR [29]	758.38	226.50	2.56	1.15	0.115	0.022	–	–
GuideNet [38]	736.24	218.83	2.25	0.99	0.101	0.015	82.0	93.9
NLSPN [27]	741.68	199.59	1.99	0.84	0.092	0.012	88.0	95.4
ACMNet [47]	744.91	206.09	2.08	0.90	0.105	0.015	–	–
DySPN [23]	709.12	192.71	1.88	<u>0.82</u>	0.090	0.012	–	–
BEV@DC [49]	697.44	189.44	1.83	<u>0.82</u>	0.089	0.012	–	–
CFormer [46]	708.87	203.45	2.01	0.88	0.090	0.012	87.5	95.3
LRRU [40]	696.51	189.96	1.87	**0.81**	0.091	<u>0.011</u>	–	–
TPVD [45]	693.97	<u>188.60</u>	<u>1.82</u>	**0.81**	<u>0.086</u>	**0.010**	–	–
ImprovingDC [41]	686.46	**187.95**	1.83	**0.81**	0.091	<u>0.011</u>	–	–
OGNI-DC [50]	708.38	193.20	1.86	0.83	0.087	<u>0.011</u>	<u>88.3</u>	<u>95.6</u>
BP-Net [37]	684.90	194.69	<u>1.82</u>	0.84	0.089	0.012	87.2	95.3
DMD3C [22]	**678.12**	194.46	<u>1.82</u>	0.85	**0.085**	<u>0.011</u>	–	–
GBPN	<u>681.61</u>	192.16	**1.79**	<u>0.82</u>	**0.085**	<u>0.011</u>	**89.3**	**95.9**

4.1 Comparison with State-of-the-Art Methods

We evaluate GBPN (the GBPN-2) on the official test sets of the NYUv2 [35] dataset and the KITTI Depth Completion (DC) [13] dataset. Quantitative comparisons between GBPN and other top-performing published methods are presented in Tab. 1. On the KITTI DC benchmark[1], our method achieves the best iRMSE among all submissions at the time of writing, and demonstrates highly competitive performance across other evaluation metrics. Specifically, our method ranks second under RMSE among all published papers, where DMD3C [22]

[1] `www.cvlibs.net/datasets/kitti/eval_depth.php?benchmark=depth_completion`

obtains the lowest RMSE. It's worth noting that DMD3C utilizes exact BP-Net [37] but incorporates additional supervision derived from a foundation model during training. In contrast, our method is trained solely on the standard KITTI training set from scratch, similar to BP-Net, while surpassing BP-Net in all evaluation metrics. Training with more data or distilling from a widely trained foundation model to improve our method is an interesting direction left for future work. On the NYUv2 dataset, our method achieves the best RMSE. As the commonly used $\delta_{1.25}$ metric is nearing saturation (typically $\geq 99.6\%$), we provide results using the stricter $\delta_{1.02}$ and $\delta_{1.05}$ metrics in Tab. 1 to better highlight the superiority of GBPN compared to other methods with publicly available models on the NYUv2 dataset. More comparisons can be found in Appendix A.5.

4.2 Ablation Studies

We conduct ablation studies on the NYUv2 dataset to evaluate the contribution of core components in our GBPN. Starting from a simple optimization-based baseline V_1, we incrementally add components to arrive at V_2 to V_9. All models are trained with half the epochs of the full training schedule, due to resource constraints. Quantitative results, including RMSE and $\delta_{1.02}$, are presented in Tab. 2.

Table 2: Ablation studies on NYUv2 dataset.

	Basic Block		Local Edges		Dynamic MRF		GBP Iters.		Loss	Criteria	
	Conv.	Attn.	4	8	Param.	Edge	3	5	Probability	RMSE↓(mm)	$\delta_{1.02}$ ↑(%)
V_1	✓									342.70	21.58
V_2	✓								✓	340.84	25.23
V_3	✓		✓				✓		✓	108.29	83.71
V_4		✓	✓				✓		✓	109.54	83.27
V_5	✓	✓	✓				✓		✓	107.01	84.01
V_6	✓	✓	✓		✓		✓		✓	103.92	84.69
V_7	✓	✓	✓		✓	✓	✓		✓	101.42	85.19
V_8	✓	✓		✓	✓	✓	✓		✓	100.95	85.20
V_9	✓	✓		✓	✓	✓		✓	✓	100.69	85.27

The baseline model V_1 employs a convolutional U-Net to estimate an initial depth and confidence map from a RGB image, and then optimizes the parameters of an affine transformation using constraints from sparse depth measurements. Similar to [7], this process involves solving a weighted least squares problem based solely on depth observation constraints, without any pair-wise terms. The model is trained using only an L2 loss on depth, following [37, 38].

In our ablation study, we first replace the L2 loss in V_1 with our proposed optimization-based loss function, resulting in V_2 with slight performance improvement. Building on V_2, we develop V_3, V_4, and V_5, which generate depth by solving a fixed MRF with fixed local edges, but using features from different backbones. All three variants outperform V_2 by a large margin, with V_5 achieving

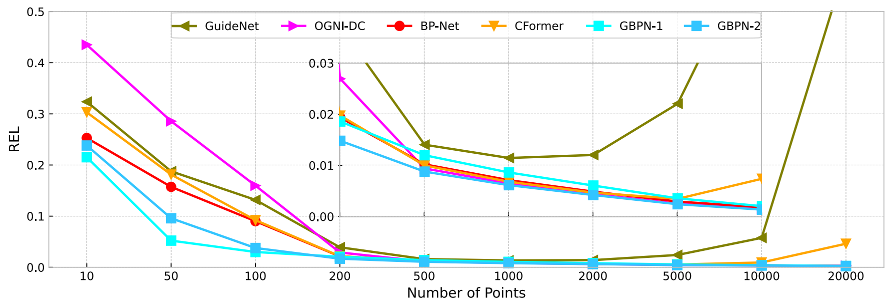

Fig. 3: REL on NYUv2 testset under various sparsity. The plot from 200 to 10000 points is zoomed in the center for better visualization.

the best RMSE. This confirms the effectiveness of our problem formulation and the proposed global-local processing unit, which combines local convolutions with long-range attention. We then extend V_5 by incorporating an MRF with dynamic parameters (V_6) and dynamic non-local edges (V_7). The performance gains of V_6 and V_7 demonstrate the advantage of a more expressive graph model. Subsequently, we increase the number of local edges in V_7 to construct V_8, and then the number of iterations to form V_9. The performance improvements seen in V_8 and V_9 highlight the benefits of more dense constraints and a greater number of optimization iterations. We attempt further increases iteration numbers but observe little performance improvement, and finally choose V_9 as GBPN-1.

4.3 Sparsity Robustness Analysis

In real-world applications, the sparsity level of the input depth map may vary significantly depending on the sensor and environment. To evaluate the robustness of our method on input sparsity, we conduct experiments on the NYUv2 dataset, comparing our approach against other SOTA methods with publicly available code and models. For this analysis, all methods for comparison are from released models trained with 500 depth points on NYUv2. These models are then directly evaluated on sparse inputs generated at various sparsity levels, ranging from 10 to 20,000 points.

As demonstrated in Fig. 3, both GBPN-1 and GBPN-2 exhibit consistently better robustness across the entire range of tested sparsity levels than others. For GBPN, the REL consistently decreases as the number of points increases. In contrast, the REL of methods like GuideNet [38] and CFormer [46] increases notably when the number of depth points becomes significantly denser (beyond approximately 5000 points) than the training sparsity (500 points). More about sparsity robustness analysis is in Appendix A.6.

We also visualize qualitative results from these compared methods under various sparsity levels in Fig. 4. In very sparse scenarios, such as the 1 and 10 points cases visualized in the first and second rows, depth maps from our

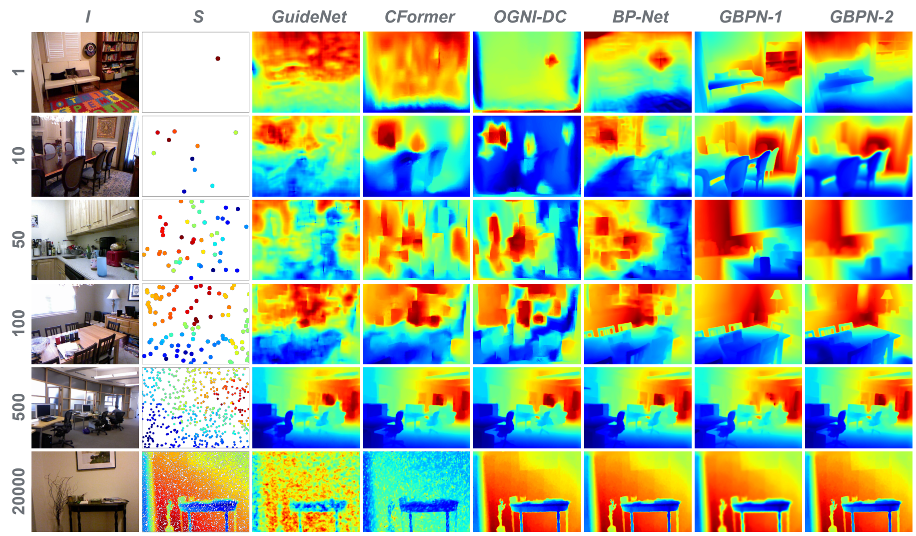

Fig. 4: Qualitative comparisons under different sparsity. All comparing methods are trained with 500 depth points and directly tested with depth input from various sparsity levels.

method maintain relatively sharp boundaries, whereas the results from other methods appear ambiguous or messy. For the 50 and 100 points cases shown in the third and fourth rows, our method can produce relatively clear depth maps with structural details, while results from others tend towards being unclear and over-smooth. In very dense scenarios, *i.e.* 20,000 points case in the last row, our method consistently produces very clear depth maps, while results from some comparison methods tend to be noisy. We attribute this superior robustness to our problem formulation, which incorporates sparse input data as a data term within a globally optimized MRF framework, making it inherently robust to the irregularity and varying density of the input sparse measurements.

4.4 Generalization Capability

To demonstrate the generalization of GBPN, we evaluate state-of-the-art depth completion methods on the VOID [43] benchmark, whose sparse depth is from visual odometry. All methods are trained on NYUv2 with 500 random points and are evaluated zero-shot on the VOID validation set under 150, 500, and 1500 point settings. Thus, this setup tests generalization across different sparsity levels, sparsity patterns, and scene domains.

As reported in Tab. 3, GBPN consistently surpasses other methods across all sparsity levels in both RMSE and MAE. We believe this robustness stems from the formulation of our model, which integrates sparse depth as principled observation terms within an MRF. By avoiding specialized neural layers for sparse input, our method achieves stronger generalization.

Table 3: Depth completion results on the VOID dataset at different sparsity levels (1500, 500, 150 points). RMSE and MAE are reported in meters.

Method	VOID 1500		VOID 500		VOID 150	
	RMSE (m)	MAE (m)	RMSE (m)	MAE (m)	RMSE (m)	MAE (m)
OGNI [50]	0.919	0.385	1.097	0.606	1.249	0.747
NLSPN [27]	0.687	0.221	0.758	0.297	0.932	0.426
CFormer [46]	0.726	0.261	0.821	0.385	0.956	0.487
BP-Net [37]	0.672	0.251	0.721	0.341	0.847	0.432
GuideNet [38]	0.744	0.317	0.895	0.485	1.190	0.750
GBPN	**0.649**	**0.218**	**0.692**	**0.282**	**0.830**	**0.377**

5 Conclusion

This paper presents the Gaussian Belief Propagation Network (GBPN), a novel framework that seamlessly integrates deep learning with probabilistic graphical models for depth completion. GBPN employs a Graphical Model Construction Network (GMCN) to dynamically build a scene-specific Markov Random Field (MRF). This learned MRF formulation naturally incorporates sparse data and models complex spatial dependencies through adaptive non-local edges and dynamic parameters. Dense depth inference is efficiently performed using Gaussian Belief Propagation, enhanced by a serial & parallel message passing scheme. Extensive experiments on the NYUv2, KITTI, and VOID benchmarks demonstrate that GBPN achieves state-of-the-art accuracy and exhibits superior robustness and generalizable capability.

Limitations and Future work. As a pioneering work integrating deep learning with probabilistic graphical models for depth completion, several hyperparameters in our model, such as the number of neighbors and propagation steps, are set empirically. Fully exploring the hyperparameter space to find an optimal trade-off between accuracy and efficiency or developing methods to adaptively determine these parameters, represents interesting directions for future exploration. Furthermore, our model is only trained on limited, curated data. While demonstrating impressive robustness against depth sparsity, training on larger, more diverse datasets or leveraging supervision from pre-trained foundation models is promising avenues for enhancing generalization and performance in broader scenarios.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China (NSFC) under Grants T2521006, 62273353, and Innovation Research Foundation of National University of Defense Technology (Innovation: Research Foundation of NUDT).

References

1. Bickson, D.: Gaussian belief propagation: Theory and aplication. arXiv preprint arXiv:0811.2518 (2008)
2. Burns, B., Brock, O.: Sampling-based motion planning with sensing uncertainty. In: ICRA. pp. 3313–3318. IEEE (2007)
3. Chen, Q., Koltun, V.: Fast mrf optimization with application to depth reconstruction. In: CVPR. pp. 3914–3921 (2014)
4. Chen, Y., Yang, B., Liang, M., Urtasun, R.: Learning joint 2d-3d representations for depth completion. In: ICCV. pp. 10023–10032 (2019)
5. Cheng, X., Wang, P., Guan, C., Yang, R.: Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion. In: AAAI. vol. 34, pp. 10615–10622 (2020)
6. Cheng, X., Wang, P., Yang, R.: Learning depth with convolutional spatial propagation network. IEEE TPAMI **42**(10), 2361–2379 (2019)
7. Conti, A., Poggi, M., Mattoccia, S.: Sparsity agnostic depth completion. In: WACV. pp. 5871–5880 (2023)
8. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: ICCV. pp. 764–773 (2017)
9. Davison, A.J., Ortiz, J.: Futuremapping 2: Gaussian belief propagation for spatial ai. arXiv preprint arXiv:1910.14139 (2019)
10. Diebel, J., Thrun, S.: An application of markov random fields to range sensing. NeurIPS **18** (2005)
11. Eldesokey, A., Felsberg, M., Khan, F.S.: Confidence propagation through cnns for guided sparse depth regression. IEEE TPAMI **42**(10), 2423–2436 (2019)
12. Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision. IJCV **40**, 25–47 (2000)
13. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
14. Hassani, A., Shi, H.: Dilated neighborhood attention transformer. arXiv preprint arXiv:2209.15001 (2022)
15. Hawe, S., Kleinsteuber, M., Diepold, K.: Dense disparity maps from sparse disparity measurements. In: ICCV. pp. 2126–2133. IEEE (2011)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
17. Huang, Z., Fan, J., Cheng, S., Yi, S., Wang, X., Li, H.: Hms-net: Hierarchical multi-scale sparsity-invariant network for sparse depth completion. IEEE TIP **29**, 3429–3441 (2019)
18. Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? NeurIPS **30** (2017)
19. Kopf, J., Cohen, M.F., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. ACM TOG **26**(3), 96–es (2007)
20. Ku, J., Harakeh, A., Waslander, S.L.: In defense of classical image processing: Fast depth completion on the cpu. In: 2018 15th Conference on Computer and Robot Vision (CRV). pp. 16–22. IEEE (2018)
21. Li, T., Chen, Z., Liu, H., Wang, C.: Fdct: Fast depth completion for transparent objects. IEEE Robotics and Automation Letters **8**(9), 5823–5830 (2023)
22. Liang, Y., Hu, Y., Shao, W., Fu, Y.: Distilling monocular foundation model for fine-grained depth completion. CVPR (2025)

23. Lin, Y., Cheng, T., Zhong, Q., Zhou, W., Yang, H.: Dynamic spatial propagation network for depth completion. In: AAAI. vol. 36, pp. 1638–1646 (2022)
24. Liu, F., Shen, C., Lin, G.: Deep convolutional neural fields for depth estimation from a single image. In: CVPR. pp. 5162–5170 (2015)
25. Ma, F., Karaman, S.: Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In: ICRA. pp. 4796–4803. IEEE (2018)
26. Murphy, K., Weiss, Y., Jordan, M.I.: Loopy belief propagation for approximate inference: An empirical study. arXiv preprint arXiv:1301.6725 (2013)
27. Park, J., Joo, K., Hu, Z., Liu, C.K., So Kweon, I.: Non-local spatial propagation network for depth completion. In: ECCV. pp. 120–136. Springer (2020)
28. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann (1988)
29. Qiu, J., Cui, Z., Zhang, Y., Zhang, X., Liu, S., Zeng, B., Pollefeys, M.: Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In: CVPR. pp. 3313–3322 (2019)
30. Qu, C., Nguyen, T., Taylor, C.: Depth completion via deep basis fitting. In: WACV. pp. 71–80 (2020)
31. Roessle, B., Barron, J.T., Mildenhall, B., Srinivasan, P.P., Nießner, M.: Dense depth priors for neural radiance fields from sparse input views. In: CVPR. pp. 12892–12901 (2022)
32. Ron, D., Duan, K., Ma, C., Xu, N., Wang, S., Hanumante, S., Sagar, D.: Monocular depth estimation via deep structured models with ordinal constraints. In: 2018 International Conference on 3D Vision (3DV). pp. 570–577. IEEE (2018)
33. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
34. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: CVPR. pp. 4104–4113 (2016)
35. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV. pp. 746–760 (2012)
36. Sun, J., Zheng, N.N., Shum, H.Y.: Stereo matching using belief propagation. IEEE TPAMI **25**(7), 787–800 (2003)
37. Tang, J., Tian, F.P., An, B., Li, J., Tan, P.: Bilateral propagation network for depth completion. In: CVPR. pp. 9763–9772 (2024)
38. Tang, J., Tian, F.P., Feng, W., Li, J., Tan, P.: Learning guided convolutional network for depth completion. IEEE TIP **30**, 1116–1129 (2020)
39. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity invariant cnns. In: IEEE International Conference on 3D Vision (3DV). pp. 11–20 (2017)
40. Wang, Y., Li, B., Zhang, G., Liu, Q., Gao, T., Dai, Y.: Lrru: Long-short range recurrent updating networks for depth completion. In: ICCV. pp. 9422–9432 (2023)
41. Wang, Y., Zhang, G., Wang, S., Li, B., Liu, Q., Hui, L., Dai, Y.: Improving depth completion via depth feature upsampling. In: CVPR. pp. 21104–21113 (2024)
42. Weiss, Y., Freeman, W.: Correctness of belief propagation in gaussian graphical models of arbitrary topology. NeurIPS **12** (1999)
43. Wong, A., Fei, X., Tsuei, S., Soatto, S.: Unsupervised depth completion from visual inertial odometry. IEEE Robotics and Automation Letters **5**(2), 1899–1906 (2020)
44. Wu, X., Peng, L., Yang, H., Xie, L., Huang, C., Deng, C., Liu, H., Cai, D.: Sparse fuse dense: Towards high quality 3d detection with depth completion. In: CVPR. pp. 5418–5427 (2022)

45. Yan, Z., Lin, Y., Wang, K., Zheng, Y., Wang, Y., Zhang, Z., Li, J., Yang, J.: Tri-perspective view decomposition for geometry-aware depth completion. In: CVPR. pp. 4874–4884 (2024)
46. Zhang, Y., Guo, X., Poggi, M., Zhu, Z., Huang, G., Mattoccia, S.: Completionformer: Depth completion with convolutions and vision transformers. In: CVPR. pp. 18527–18536 (2023)
47. Zhao, S., Gong, M., Fu, H., Tao, D.: Adaptive context-aware multi-modal network for depth completion. IEEE TIP **30**, 5264–5276 (2021)
48. Zhao, Y., Bai, L., Zhang, Z., Huang, X.: A surface geometry model for lidar depth completion. IEEE Robotics and Automation Letters **6**(3), 4457–4464 (2021)
49. Zhou, W., Yan, X., Liao, Y., Lin, Y., Huang, J., Zhao, G., Cui, S., Li, Z.: Bev@ dc: Bird's-eye view assisted training for depth completion. In: CVPR. pp. 9233–9242 (2023)
50. Zuo, Y., Deng, J.: Ogni-dc: Robust depth completion with optimization-guided neural iterations. In: ECCV. pp. 78–95. Springer (2024)