This CVPR workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Few-Shot Depth Completion Using Denoising Diffusion Probabilistic Model

Weihang Ran*, Wei Yuan*†, Ryosuke Shibasaki
The University of Tokyo
5-1-5 Kashiwanoha, 277-8568 Chiba, Japan
{ranweihang, miloyw, shiba}@csis.u-tokyo.ac.jp

Abstract

Generating dense depth maps from sparse LiDAR data is a challenging task, benefiting a lot of computer vision and photogrammetry tasks including autonomous driving, 3D point cloud generation, and aerial spatial awareness. Using RGB images as guidance to generate pixel-wise depth map is good, but these multi-modal data fusion networks always need numerous high-quality datasets like KITTI dataset to train on. Since this may be difficult in some cases, how to achieve few-shot learning with less train samples is worth discussing. So in this paper, we firstly proposed a few-shot learning paradigm for depth completion based on pre-trained denoising diffusion probabilistic model. To evaluate our model and other baselines, we constructed a smaller train set with only 12.5% samples from KITTI depth completion dataset to test their few-shot learning ability. Our model achieved the best on all metrics with a 5% improvement in RMSE compared to the second-place model.

1. Introduction

A depth map refers to an image in which the value of each pixel corresponds to a depth value. Traditional vision-based methods, such as dense stereo matching, usually suffer from occlusion and illumination to generate fully completed pixel-wised depth maps [38, 39]. Meanwhile, depth maps acquired by LiDAR and RGB-D cameras are usually in a low resolution, and the sensing platform is usually quite expensive. Thus, how to generate dense pixel-wise depth maps from sparse and low-resolution depth maps has become a hot research topic in the last decades. It has received tremendous attention in computer vision and photogrammetry fields with the potential to achieve high-resolution dense 3D points cloud generation, autonomous driving, and aerial spatial awareness [18, 35]. Although there are already some SLAM (simultaneous Localization and Mapping) approaches, such as infiniTAM can generate dense depth maps, they are usually not scalable to large scenes, restricting their applicability in real-world scenarios such as high-resolution DSM/DOM generation [31]. Depth completion can compensate for the shortcomings of previous methods, providing a good foundation for the fully automatic generation of high-precision dense depth maps [40]. With the rapid development of deep learning technology in recent years, it has shown dominant power in depth completion field and is worth further exploration.

The problem of depth completion has been discussed for years. At first, researchers employed convolutional neural networks to predict dense depth maps from a single sparse depth map [6], which is called the unguided method. In order to deal with this particular sparse image data, a number of variant CNNs were introduced into this field, such as Sparsity-invariant CNNs [32] and normalized-CNNs [8]. However, due to the lack of semantic information, the performance of unguided methods is limited.

To solve this problem, multi-modal methods were applied to depth completion. Multi-modal methods aimed to leverage the complementary strengths of different modality data to overcome the limitations of individual sensors or measurements. For example, by combining depth measurements with color or texture information from RGB images, the multi-modal method, called the image-guided method, can provide a more complete and accurate depth completion result. Several studies have investigated the effectiveness of multi-modal methods in depth completion research. [22, 25] concatenated the sparse depth map and the corresponding RGB image before inputting them to the model, while [14, 30] attempted to use dual-encoder networks or double-branch networks to perform late-fusion. These researches showed that the multi-modal method improves the accuracy of depth completion models compared to using individual sparse depth maps alone. Despite their benefits, multi-modal methods also have some limitations. One challenge is the integration of multiple modalities, which can be computationally expensive and require careful data fusion and alignment. Moreover, the effectiveness of multi-modal

*Equal contribution.
†Corresponding author.

methods depends on the compatibility and quality of the different types of data. In some cases, certain modalities may not provide additional information or may even introduce noise or artifacts into the depth completion.

In order to make full use of the information present in the original RGB image without destroying the structure of its contents, spatial propagation network (SPN) [20] was introduced into depth completion. SPN-based methods utilize an affinity matrix to represent the correlation between one point and its neighbors, thus it is always used to refine and gain a fine-grained prediction in computer vision tasks. [4,5] extended the vanilla SPN for the task of depth completion and integrated convolutional operations into the spatial propagation module. Relying on such explicit construction modules in deep learning models, SPN-based methods improved the performance of traditional multi-modal methods as a refinement method.

But if we omit this post-processing module, we can find that the current state-of-the-art models still need a pre-trained backbone to provide reliable coarse predictions to the SPN module for further refinement. This can be difficult under some circumstances because high-quality raw depth maps with corresponding RGB images and dense annotations are quite expensive, not to mention semantic annotations. Therefore, how to use fewer training samples to obtain a backbone network with comparable performance is worth exploring. [12] developed a generative model named the denoising diffusion probabilistic model (DDPM) that achieved image generation by adding noise and reducing iteratively. This kind of model has been proven powerful in many computer vision tasks, including image segmentation [2], change detection [1], and image super-resolution [26]. Since the training process of the diffusion model only needs RGB image data, which can be seen as self-supervised training [33, 36], it also has the potential of achieving few-shot learning (FSL) by generalizing features learned from abundant RGB data to fewer samples with ground truth.

So in this paper, we first introduced DDPM into the depth completion tasks and proposed a few-shot learning paradigm for depth completion by pre-training a diffusion model on a mass of RGB data without corresponding depth annotations. Then only a limited number of samples with corresponding depth maps is needed to fine-tune a fusion module and achieve good results. We trained our Dif-Depth model and other baselines on a small subset of the KITTI depth completion dataset and tested them on the selected validation set. The main contribution of this paper can be summarized as follows:

1. We firstly proposed a DDPM-based framework that can achieve few-shot learning on depth completion tasks. The robust feature-extracting capability of our backbone is obtained through a fully unsupervised process. So it can be generalized easily to a specific task after retraining by fewer samples.

2. We constructed a 50-shot train set from the original KITTI depth completion dataset by sampling data from each sequence respectively. This training set contains only 12.5% of the original dataset and can be used for few-shot learning.

3. We conducted a two-stage pre-training-and-fine-tuning strategy to train our model and evaluated it with other baselines on KITTI selected validation set. Our model achieved the best on all metrics, indicating its robust feature extracting and data fusion ability on small training sets.

2. Related Work

Image-guided method. When it became agreement that image-guided methods worked better than unguided methods, a lot of research aiming at how to fuse multi-modal data more effectively sprung up. The simplest way is to concatenate sparse depth maps and RGB images together before inputting to a deep learning model [22, 25], or to use different initial convolutional layers at the beginning to extract features [13, 21, 37]. Another approach is applying a two-stage network to perform a coarse-to-refinement prediction [3, 7]. However, this kind of method is too straightforward, which overlooks the complicated spatial relation between RGB images and depth maps. To perform data fusion more explicitly, more complex network architectures were proposed to fuse intermediate features on different levels. [14, 28] designed dual-branch encoder networks to process RGB image and raw depth data separately and concatenate corresponding features before delivering to the decoder. In contrast, [16] further introduced multi-scale skip connections into a network architecture to merge different source data. Dual-encoder networks only achieved data fusion during the encoding process, leaving the decoding process unconsidered. [27, 30] introduced double encoder-decoder networks into depth completion. Meanwhile, SPN-based networks that upsampled the original sparse depth map with affinity matrix learned from RGB data also developed gradually, including CSPN [5], CSPN++ [4], and DySPN [19].

Guided Filtering. [11] proposed a new filter that calculates output by considering the contents of the guide image, which can be the input image itself or a different image. It's called joint/guided image filtering and can be useful in image denoising, image defogging, and detail augmentation. [17] introduced joint filtering into depth completion, and [34] proposed a joint filtering layer to perform joint upsampling. But the original guided filter is unsuitable for sparse data like LiDAR data. [15] originally published dynamic filtering network (DFN) where the network generates filter kernels dynamically based on the input image to

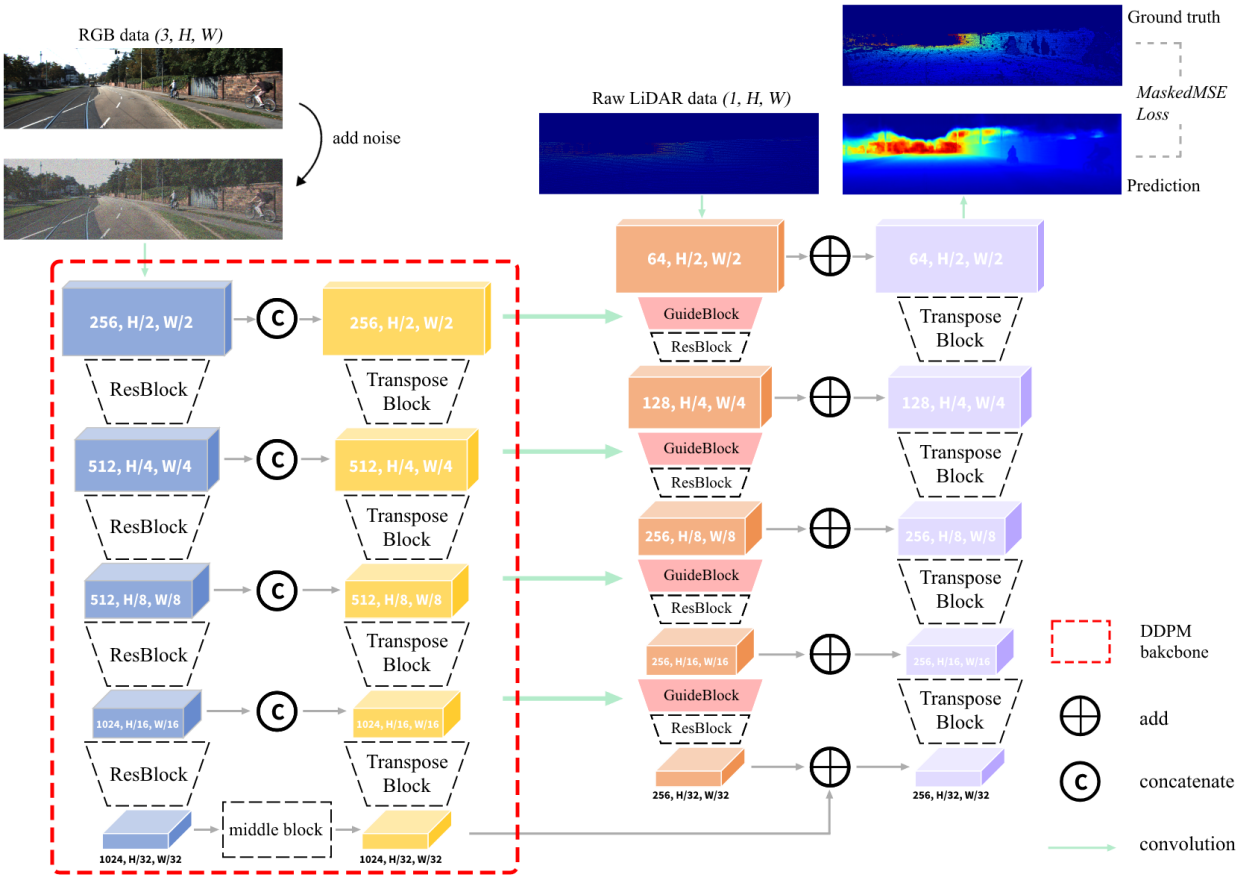

Figure 1. The entire architecture of our model. The left part is the DDPM backbone, which will be pre-trained with all RGB data. The right part is the multi-modal fusion network based on guided convolution. It will be trained on a small dataset with a backbone frozen. Our model takes an RGB image and a sparse depth map as input and predicts a dense depth map.

enable operations like local spatial transformation on the input features. But the computational consumption of kernels generated by DFN is too large, which limits its application on multi-scale levels. To address this problem, [30] suggested learning more general and powerful kernels from the guidance image and apply the kernels to fuse multi-scale features for depth completion. Inspired by these works, we also applied similar operations in our network to make full use of multi-scale features extracted by the diffusion model.

Diffusion model. Diffusion model represents one kind of generative model that can learn from two reversed processes: adding noise and denoising. The first denoising diffusion probabilistic model (DDPM) was proposed by [12], and has been found to have potential with semantic segmentation [2], image super-resolution [26], and image denoising [36]. Then a lot of variants were proposed. [24] came up with an improved DDPM which changed the latent sampling schedule from linear to cosine and made variance learnable. [29] further improved the inference speed of DDPM by making the diffusion process a non-Markov Chain process. A diffusion model takes a single RGB image as input and adds Gaussian noise into it step by step until it becomes an entire isotropic Gaussian noise image. Then it removes noise gradually and recovers the original image. They are called the forward process and the reverse process, respectively. This kind of model has many advantages, including the self-supervised training process and the strong capability of extracting multi-scale features. Thus it provides favorable conditions for few-shot multi-modal data fusion. By pre-training a diffusion model on one kind of data and fine-tuning the fusion module after the backbone, the cost of collecting high-quality data can be greatly reduced.

3. Methodology

In this paper, we propose a two-stage pre-training-and-fine-tuning strategy to achieve the few-shot learning of depth completion. First, we designed a backbone network based on the diffusion model to achieve pre-training on RGB data. This step does not require the involvement of

any depth-relevant data and can therefore be viewed as a self-supervised learning process. Then we proposed a fusion module based on guided convolution operation, which can predict a dense depth map with multi-scale features extracted from the backbone network and a sparse depth map as input. In order to improve the performance of guided convolution on sparse data, the original sparse depth map was first refined by the nearest neighbor interpolation. The training target is set to the masked MSE loss The entire architecture of our model is shown in Fig. 1.

3.1. Self-supervised pre-training

Given an RGB image $I_0 \in \mathbb{R}^{3\times H\times W}$, where H and W represent the height and width of the image, respectively. In order to learn the texture, outline, and structure features in this image, the denoising diffusion model is applied to infer the distribution of RGB feature representation. The forward process of the diffusion model is adding Gaussian noise to the original image during the time step T until it becomes an isotropic Gaussian distribution. This can be seen as a Markov process because the status under time step I_t only depends on the previous status I_{t-1}. At time step t the noisy data I_t can be defined as follows:

$$P(I_t|I_{t-1}) = \mathcal{N}(I_t; \sqrt{\alpha_t}I_{t-1}, (1-\alpha_t)\boldsymbol{Z}), \quad (1)$$

where $\mathcal{N}(\mathbf{0}, \boldsymbol{Z})$ denotes the Gaussian distribution, and $(I_0, I_1, ..., I_T)$ denotes the T-step Markov chain. $\alpha_{1:T} = (\alpha_1, ..., \alpha_T)$ represents the noise schedule that controls the variance of noise added at each step. In our research, we adopted the cosine schedule which was proposed in [24].

Since the forward process of original DDPM is performed step by step, which is very slow. [29] speeds up this process by calculating the marginal distribution of I_t given I_0 using the following formula:

$$P(I_t|I_0) = \mathcal{N}(I_t; \sqrt{\bar{\alpha}_t}I_0, (1-\bar{\alpha}_t)\boldsymbol{Z}), \quad (2)$$

where $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$. So the status under time steps t can be derived easily when given the original status I_0 and time step t.

In the reverse process, a neural network ϵ_θ is applied to perform denoising operation step by step, *e.g.* from I_x to I_{x-1}, until getting back to the original image I_0. In this process the model is trained to learn the parameters of the reverse distribution, which can be represented as follows:

$$Q(I_{t-1}|I_t) = \mathcal{N}(I_{t-1}; \mu_\theta(I_t, t), \sigma_t^2 \boldsymbol{Z}), \quad (3)$$

where σ^2 is the variance of the conditional distribution $Q(I_{t-1}|I_t)$, which can be derived from:

$$\sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t, \quad (4)$$

where $\beta_t = 1 - \alpha_t$. And the mean $\mu_\theta(I_t, t)$ of the distribution $Q(I_{t-1}|I_t)$ can be formulated as:

$$\mu_\theta(I_t, t) = \frac{1}{\sqrt{\alpha_t}}(I_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(I_t, t)) \quad (5)$$

The optimization target of the diffusion model is the difference between network prediction and the noise of the sample γ:

$$\mathcal{L}_{diff} = ||\gamma - \epsilon_\theta(I_t, t)||_2 \quad (6)$$

3.2. Few-shot data fusion

When the diffusion backbone is well pre-trained, it will be capable of extracting robust and refined features from the original image. These features can be generalized to specific tasks like depth completion with only a few training samples. We can sample a noisy image from a random time step, *e.g.* 50, and extract features as guidance for depth completion. When performing data fusion, an ordinary method concatenates features extracted from different modalities together. This kind of method overlooked the complicated structure of each modality data, thus resulting in low effective data fusion. To take full advantage of the fine-extracted features, we proposed a decoder based on guided filtering. The concept of guided filtering is to learn a changeable kernel from a guided image and use it to guide the processing of another image, such as smoothing, denoising or refinement. We adopt the guided convolution operation proposed by [30].

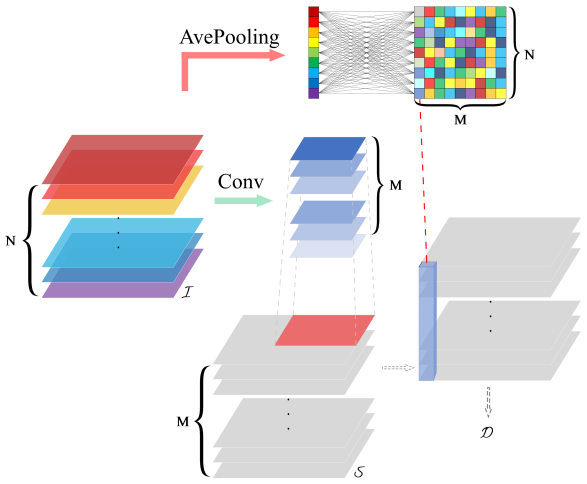

Figure 2. The entire process of Guided convolution. Depth feature $\mathcal{S}$ is filtered by kernels generated from image feature $\mathcal{I}$ and get $\mathcal{D}$. Here "AvePooling" and "Conv" refer to average pooling and convolution operations, respectively.

Given a sparse depth map *S* and a guidance image *I*, we first use a convolutional neural network to extract features $\mathcal{S}$ and $\mathcal{I}$ with *M* and *N* channels, respectively. Then a two-stage convolution is performed. In the first stage, a standard

convolutional layer $\boldsymbol{f}^{'}$ is applied to generate a spatially-variant kernel $W^{'}$ with the same resolution as depth feature $\mathcal{S}$ from the guidance image feature $\mathcal{I}$. Then the m-th channel of the kernel weights $W^{'}_{m}$ will be used to calculate with corresponding depth feature $\mathcal{S}_m$ through convolution to get $\mathcal{D}^{'}_{m}$:

$$\mathcal{D}^{'}_{m} = W^{'}_{m}(\mathcal{I}; \Theta^{'}) \otimes \mathcal{S}_m, \tag{7}$$

where $\Theta^{'}$ is the parameters of $\boldsymbol{f}^{'}$, and $\otimes$ refers to the convolution operation. Then the second stage will perform cross-channel convolution. First of all an average pooling layer is applied to the guidance image feature $\mathcal{I}$ at each channel individually to obtain a latent feature $\mathcal{I}^{'}$ with size $M \times 1 \times 1$. Then $\mathcal{I}^{'}$ is fed to a fully-connected layer $\boldsymbol{f}^{''}$ and projected to $M \times N \times 1 \times 1$ to get $W^{''}$. We use $W^{''}$ to perform 1×1 convolution and output the process depth map $\mathcal{D}$:

$$\mathcal{D} = W^{''}(\mathcal{I}^{'}; \Theta^{''}) \otimes \mathcal{D}^{'}, \tag{8}$$

where $\Theta^{''}$ is the parameters of the fully-connected layer. By doing this two-stage guided filtering, the latent information about objects and scenes in the original image is used to guide the generation of depth features. Since the filtering kernel is related to the guidance image, it would be better than simply training a standard convolutional kernel through a back-propagation process. Besides, the gradient of a normal convolution kernel is calculated as the summation over entire feature maps, which means global optimal. But it may not be the optimal status for every position because the gradients at each spatial position may not share the same descent direction. Learning a spatial-variant filtering kernel from the guidance image can address this problem and output better results. The whole process can be seen in Fig. 2.

4. Experiments

4.1. Dataset

To test the performance of our model, we conduct comprehensive experiments on the KITTI depth completion dataset [32]. It's a large outdoor dataset for autonomous vehicles, containing 86,898 RGB images with corresponding LiDAR data and depth annotation for training, 1,000 for validating, and 1,000 for testing. The original depth data is collected by using LiDAR HDL-64, which provides valid depth values on only 5.9% of all pixels. To achieve Velodyne-to-Camera calibration, the laser scanner and the cameras are firstly registered by using the fully automatic method of [10]. Then the number of disparity outliers with respect to the top performing methods in [9] jointly with the reprojection errors of a few manually selected correspondences between the laser point cloud and the images. As correspondences, edges which can be easily located by humans in both images and point clouds are selected. And optimization is carried out by drawing samples using Metropolis-Hastings and selecting the solution with the lowest energy. The ground truth is generated by accumulating LiDAR and stereo estimation of the scenes, increasing valid depth values to 16% of all pixels. To evaluate the capability of few-shot learning, we sampled around 50 data from each sequence and formed a 50-shot train set with around 11,000 data. All of the models are trained on this small training set and validated on the selected validation set.

4.2. Evaluation metrics

Following the KITTI benchmark and previous depth completion research, we employ four commonly used metrics to evaluate the performance of our model: root mean squared error (RMSE), mean absolute error (MAE), root mean squared error of the inverse depth (iRMSE) and mean absolute error of the inverse depth (iMAE). Since RMSE and MAE measure the depth accuracy directly and RMSE is more sensitive, it's selected as the primary evaluation criterion on the KITTI leaderboard. iRMSE and iMAR compute the mean error of inverse depth by giving less weight to far-away points. All the metrics for evaluation are shown as follows:

$$\begin{aligned}
&RMSE(mm): \sqrt{\frac{1}{v}\sum_{x}(\hat{h_x} - h_x)^2} \\
&MAE(mm): \frac{1}{v}\sum_{x}|\hat{h_x} - h_x| \\
&iRMSE(1/km): \sqrt{\frac{1}{v}\sum_{x}(\frac{1}{\hat{h_x}} - \frac{1}{h_x})^2} \\
&iMAE(1/km): \frac{1}{v}\sum_{x}|\frac{1}{\hat{h_x}} - \frac{1}{h_x}|
\end{aligned} \tag{9}$$

4.3. Implementation details

All of the experiments were conducted on 3 NVIDIA GeForce RTX 3090 24-GB GPUs. The experimental environment is Python 3.6 and PyTorch 1.10.1 with CUDA 11.1. For the first-stage training, we used a diffusion model pre-trained on the ImageNet dataset with a resolution of 256×256. We continue to train it on the KITTI depth completion dataset for around 40,000 iterations to enhance its performance on outdoor scene images. The training target is set to $\mathcal{L}_1$, and we use an Adam optimizer for training. The initial learning rate is 1e-3, and the batch size is 3 in this process.

When the pre-training is finished, the diffusion model will be very powerful in extracting latent features from input images. Then we freeze the backbone and continue to train the guide filter part. The training target of this stage is

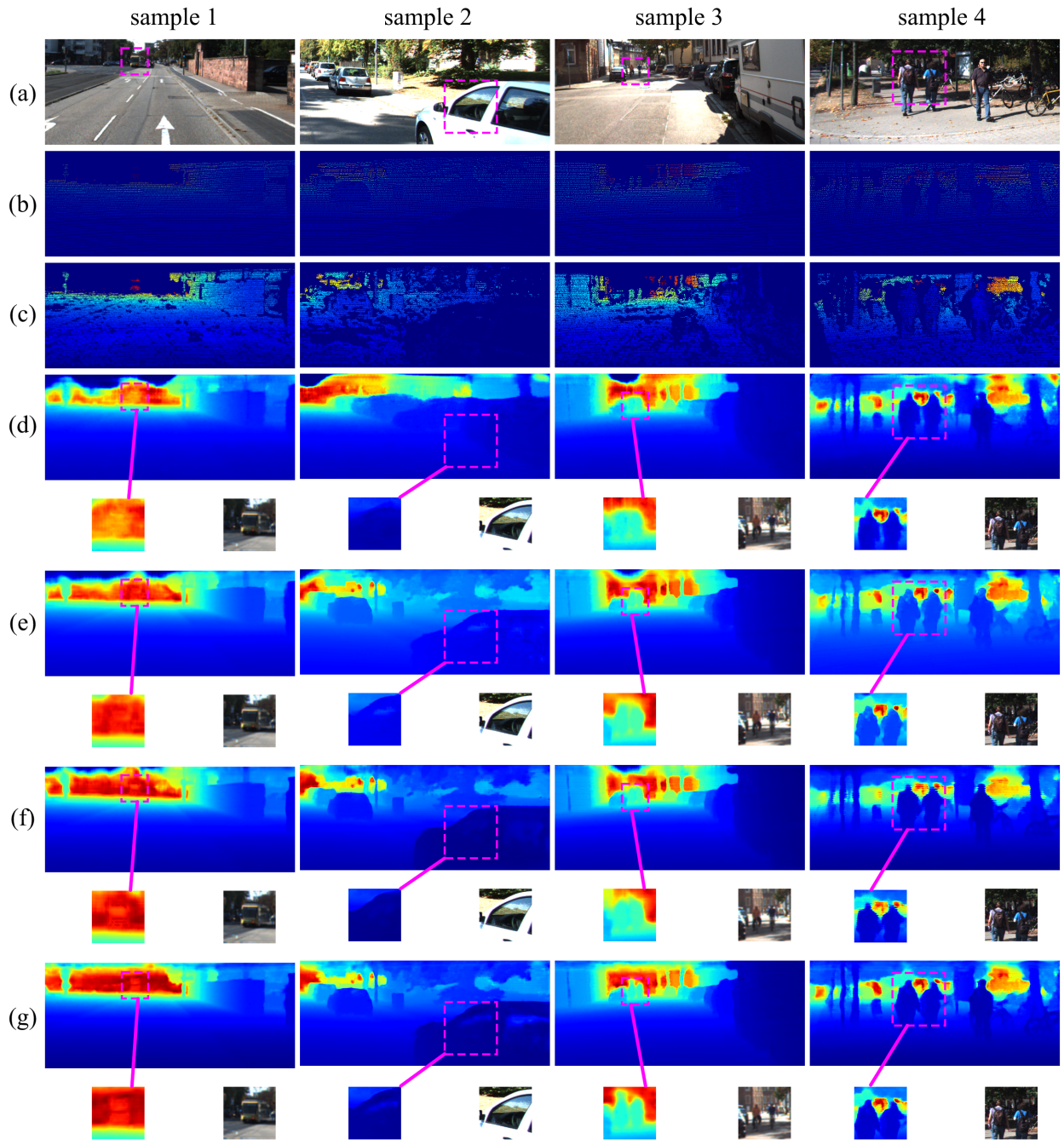

Figure 3. Qualitative results on KITTI selected validation set. From top to bottom: (a) original RGB images; (b) sparse LiDAR data; (c) ground truth; (d) Sparse-to-Dense taking RGB and sparse LiDAR data as input; (e) GuideNet; (f) SemAttNet taking RGB, LiDAR and semantic annotations as input; (g) Our model.

masked MSE loss, which can be defined as follows:

$$L_{masked} = argmin||D_p^{gt} - D_p||^2 \quad (10)$$

where D_p^{gt} and D_p are the ground truth depth and the predicted depth value at pixel p. Since the ground truth contains a lot of pixels without depth value, we only take pixels with valid depth value into consideration. An Adam optimizer with a weight decay of 1e-6 is employed for optimizing. The initial learning rate is set to 1e-4, and the batch size is 3.

Method	Pre-Trained	RMSE(mm)	MAE(mm)	iRMSE(1/km)	iMAE(1/km)
SemAttNet (bb)* [23]	None	1344.371	796.255	7.075	4.962
Sparse-to-Dense (gd) [21]	ImageNet	1298.242	524.353	6.836	3.193
SemAttNet* [23]	None	1254.914	651.044	5.381	3.485
Sparse-to-Dense (rgbd) [21]	ImageNet	1247.484	501.211	6.141	2.912
GuideNet [30]	None	1016.314	297.465	7.937	1.539
Dif-Depth (Ours)	ImageNet + KITTI (RGB)	**965.861**	**290.761**	**3.625**	**1.475**

Table 1. Quantitative results on KITTI selected validation set. All evaluation metrics are lower the better. Different methods are ranked according to RMSE. * denotes extra semantic annotation data. "gb" means taking gray images and LiDAR data as input, and "rgbd" means taking RGB images and LiDAR data as input. "bb" means only three-branches backbone was used.

4.4. Experiment results on KITTI validation set

Tab. 1 shows the experimental results on KITTI selected validation set. Our model achieved the best performance on all metrics, proving its powerful feature-extracting and generalizing ability to learn from very few train samples. We can see that compared with concatenating data fusion methods, guided convolution-based methods such as GuideNet and our model show stronger learning capability on smaller datasets. This indicates guided convolution is easier to train than the standard convolution layer when conducting multi-modal features fusion. And compared with GuideNet, we can see that the pre-trained backbone improved the performance of network by a large margin. This confirmed the effectiveness of our strategy to improve the few-shot learning ability by collecting general features from a large amount of samples in a self-supervised way. When comparing with models taking extra data like semantic annotations for inputting like SemAttNet(bb) or models employing post-processing modules like SemAttNet, our model still performed better than them. This showed that the features learned by our model also contain semantic information and rich in detail.

Visualization results of our model and other baselines are in Fig. 3. It can be seen that our model performed better in both overall accuracy and detail refinement, especially the identification of objects from a very long distance. For example, for the bus in sample 1, our model painted a more clear outline than the other models. And for the two riders in sample 3, our model also successfully separated them into two different objects in the prediction results. It's also capable of dealing with transparent objects, such as car windows in sample 2. Another obvious advantage of the predictions generated by our model is the top part of the image, where the sky is. Intuitively the sky should be infinitely far. However, due to the absence of depth values in the ground truth, most previous models always predict badly in this part and generate irregular boundaries for the sky. Our model predicts better results in this part and draws a clear outline of the sky. This demonstrated the geometric correctness and content validity of features extracted by the pre-trained diffusion model.

4.5. Comparison of different time steps

According to previous research [1, 2], since the diffusion model is trained to rebuild original images from noisy images, the difficulty varies with different degrees of added noise. And the features extracted by diffusion model vary with the input time step as well. So in our experiments, we further explored the performance of our model under different time steps. Results are shown in Tab. 2.

From the results under different time steps we can see that the performance of our model increased first and then decreased with the increase of time step. When time step is very small, *e.g.* $t = 5$, features output by diffusion model capture less semantic information than when $t = 50$. However, when time step is very large, *e.g.* $t = 400$, they can be uninformative as well. And the characteristic of feature representation under different time step can also influence the result of multi-modal data fusion. We believe that this is related to the training and learning methods of the diffusion model. Because the time step is randomly selected during the training process, the probability of selecting very small or very large time steps is low, which results in the model having limited opportunities to learn in these two scenarios. On the other hand, within the range of moderately sized time steps, more noise can lead to excessive damage to the structure of the original image, thereby increasing the difficulty of learning. Therefore, the model performs best at a slightly smaller but not-close-to-zero time step.

time step	RMSE	MAE	iRMSE	iMAE
t=5	968.330	292.060	5.039	1.452
t=50	**965.861**	**290.761**	**3.625**	**1.475**
t=150	997.471	305.883	3.711	1.499
t=400	1003.739	304.426	3.446	1.433

Table 2. Quantitative results of different time steps on KITTI selected validation set.

5. Conclusion

In this paper we proposed a new paradigm for few-shot depth completion. We firstly introduced diffusion model into depth completion field and regarded it as a solution when there are not a lot of high-quality data with ground truth. Then we designed a two-stage pre-training-and-fine-tuning strategy to extract general features from plenty of RGB data and fuse them with LiDAR data effectively. To evaluate the performance of our model and compare it with other baselines, we constructed a 50-shot train set from KITTI depth completion dataset containing only 12.5% of the original data. Expensive experiments were conducted and we proved that our model can achieve state-of-the-art performance in few-shot depth completion with a 5% improvement in RMSE compared to the second-place model. This indicates the effectiveness of our strategy to learn general features in self-supervised way and fine-tuned to specific tasks. In the future we would like to test our model on more different datasets and think of the possibility to combine different datasets for pre-training.

6. Acknowledgement

This work is supported in part by the National Natural Science Foundation of China under Grant 41771479, in part by the National High-Resolution Earth Observation System (the Civil Part) under Grant 50-H31D01-0508-13/15 and in part by the Japan Society for the Promotion of Science under Grant 23K13419.

References

[1] Wele Gedara Chaminda Bandara, Nithin Gopalakrishnan Nair, and Vishal M Patel. Ddpm-cd: Remote sensing change detection using denoising diffusion probabilistic models. *arXiv preprint arXiv:2206.11892*, 2022. 2, 7

[2] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021. 2, 3, 7

[3] Zhao Chen, Vijay Badrinarayanan, Gilad Drozdov, and Andrew Rabinovich. Estimating depth from rgb and sparse sensing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 167–182, 2018. 2

[4] Xinjing Cheng, Peng Wang, Chenye Guan, and Ruigang Yang. Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10615–10622, 2020. 2

[5] Xinjing Cheng, Peng Wang, and Ruigang Yang. Learning depth with convolutional spatial propagation network. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2361–2379, 2019. 2

[6] Nathaniel Chodosh, Chaoyang Wang, and Simon Lucey. Deep convolutional compressed sensing for lidar depth completion. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part I 14*, pages 499–513. Springer, 2019. 1

[7] Martin Dimitrievski, Peter Veelaert, and Wilfried Philips. Learning morphological operators for depth completion. In *Advanced Concepts for Intelligent Vision Systems: 19th International Conference, ACIVS 2018, Poitiers, France, September 24–27, 2018, Proceedings 19*, pages 450–461. Springer, 2018. 2

[8] Abdelrahman Eldesokey, Michael Felsberg, and Fahad Shahbaz Khan. Propagating confidences through cnns for sparse data regression. *arXiv preprint arXiv:1805.11913*, 2018. 1

[9] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 5

[10] Andreas Geiger, Frank Moosmann, Ömer Car, and Bernhard Schuster. Automatic camera and range sensor calibration using a single shot. In *2012 IEEE international conference on robotics and automation*, pages 3936–3943. IEEE, 2012. 5

[11] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. *IEEE transactions on pattern analysis and machine intelligence*, 35(6):1397–1409, 2012. 2

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2, 3

[13] Saif Imran, Yunfei Long, Xiaoming Liu, and Daniel Morris. Depth coefficients for depth completion. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12438–12447. IEEE, 2019. 2

[14] Maximilian Jaritz, Raoul De Charette, Emilie Wirbel, Xavier Perrotton, and Fawzi Nashashibi. Sparse and dense data with cnns: Depth completion and semantic segmentation. In *2018 International Conference on 3D Vision (3DV)*, pages 52–60. IEEE, 2018. 1, 2

[15] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 2

[16] Ang Li, Zejian Yuan, Yonggen Ling, Wanchao Chi, Chong Zhang, et al. A multi-scale guided cascade hourglass network for depth completion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 32–40, 2020. 2

[17] Yijun Li, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep joint image filtering. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 154–169. Springer, 2016. 2

[18] Ming Liang, Bin Yang, Yun Chen, Rui Hu, and Raquel Urtasun. Multi-task multi-sensor fusion for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7345–7353, 2019. 1

[19] Yuankai Lin, Tao Cheng, Qi Zhong, Wending Zhou, and Hua Yang. Dynamic spatial propagation network for depth com-

pletion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1638–1646, 2022. 2

[20] Sifei Liu, Shalini De Mello, Jinwei Gu, Guangyu Zhong, Ming-Hsuan Yang, and Jan Kautz. Learning affinity via spatial propagation networks. *Advances in Neural Information Processing Systems*, 30, 2017. 2

[21] Fangchang Ma, Guilherme Venturelli Cavalheiro, and Sertac Karaman. Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3288–3295. IEEE, 2019. 2, 7

[22] Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 4796–4803. IEEE, 2018. 1, 2

[23] Danish Nazir, Alain Pagani, Marcus Liwicki, Didier Stricker, and Muhammad Zeshan Afzal. Semattnet: Toward attention-based semantic aware guided depth completion. *IEEE Access*, 10:120781–120791, 2022. 7

[24] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 3, 4

[25] Jiaxiong Qiu, Zhaopeng Cui, Yinda Zhang, Xingdi Zhang, Shuaicheng Liu, Bing Zeng, and Marc Pollefeys. Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3313–3322, 2019. 1, 2

[26] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2, 3

[27] René Schuster, Oliver Wasenmuller, Christian Unger, and Didier Stricker. Ssgp: Sparse spatial guided propagation for robust and generic interpolation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 197–206, 2021. 2

[28] Shreyas S Shivakumar, Ty Nguyen, Ian D Miller, Steven W Chen, Vijay Kumar, and Camillo J Taylor. Dfusenet: Deep fusion of rgb and sparse depth information for image guided dense depth completion. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 13–20. IEEE, 2019. 2

[29] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3, 4

[30] Jie Tang, Fei-Peng Tian, Wei Feng, Jian Li, and Ping Tan. Learning guided convolutional network for depth completion. *IEEE Transactions on Image Processing*, 30:1116–1129, 2020. 1, 2, 3, 4, 7

[31] Lucas Teixeira, Martin R Oswald, Marc Pollefeys, and Margarita Chli. Aerial single-view depth completion with image-guided uncertainty estimation. *IEEE Robotics and Automation Letters*, 5(2):1055–1062, 2020. 1

[32] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *2017 international conference on 3D Vision (3DV)*, pages 11–20. IEEE, 2017. 1, 5

[33] Chulin Wang, Kyongmin Yeo, Xiao Jin, Andres Codas, Levente J Klein, and Bruce Elmegreen. S3rp: Self-supervised super-resolution and prediction for advection-diffusion process. *arXiv preprint arXiv:2111.04639*, 2021. 2

[34] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Fast end-to-end trainable guided filter. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1838–1847, 2018. 2

[35] Xiaopei Wu, Liang Peng, Honghui Yang, Liang Xie, Chenxi Huang, Chengqi Deng, Haifeng Liu, and Deng Cai. Sparse fuse dense: Towards high quality 3d detection with depth completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5418–5427, 2022. 1

[36] Tiange Xiang, Mahmut Yurt, Ali B Syed, Kawin Setsompop, and Akshay Chaudhari. Ddm ̂2: Self-supervised diffusion mri denoising with generative diffusion models. *arXiv preprint arXiv:2302.03018*, 2023. 2, 3

[37] Yan Xu, Xinge Zhu, Jianping Shi, Guofeng Zhang, Hujun Bao, and Hongsheng Li. Depth completion from sparse lidar data with depth-normal constraints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2811–2820, 2019. 2

[38] Wei Yuan, Xiuxiao Yuan, Yang Cai, and Ryosuke Shibasaki. Fully automatic dom generation method based on optical flow field dense image matching. *Geo-spatial Information Science*, pages 1–15, 2023. 1

[39] Wei Yuan, Xiuxiao Yuan, Shu Xu, Jianya Gong, and Ryosuke Shibasaki. Dense image-matching via optical flow field estimation and fast-guided filter refinement. *Remote Sensing*, 11(20):2410, 2019. 1

[40] Xiuxiao Yuan, Yang Cai, and Wei Yuan. Voronoi centerline-based seamline network generation method. *Remote Sensing*, 15(4):917, 2023. 1