2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

3DR-DIFF: Blind Diffusion Inpainting for 3D Point Cloud Reconstruction and Segmentation

K.T. Yasas Mahima[1], Asanka G. Perera[2], Sreenatha Anavatti[1], and Matt Garratt[1]

Abstract— **LiDAR-based 3D perception is a focal point in autonomous vehicle research due to its efficacy in real-world environments and falling costs. However, recent research reveals challenges with LiDAR sensing under corruptions that occur due to adverse weather conditions and sensor-level errors, known as common corruptions. In particular, the majority of these corruptions lead to sparsity or noise in LiDAR point clouds, degrading the performance of downstream perception tasks. To address this, we propose a blind inpainting method named 3DR-DIFF, utilizing diffusion networks to reconstruct and segment corrupted point clouds. 3DR-DIFF comprises two key components: a corrupted region prediction network, acting as a binary mask predictor, and a conditional diffusion network. The evaluation results demonstrate that the 3DR-DIFF is able to reconstruct the LiDAR samples with a depth error of less than 0.56 mean absolute error (MAE) and an intensity error of 0.02 MAE, along with an average segmentation performance of 0.43 mean intersection over union. Furthermore, benchmarking results highlight that 3DR-DIFF outperforms state-of-the-art methods in reconstructing LiDAR beam-missing scenarios, exhibiting an approximately 9.2% lower error for a degradation of 1 MAE.**

I. INTRODUCTION

Over the last decade, research into autonomous vehicle (AV) perception tasks has grown rapidly. Given the challenging nature of AV operating environments, the AV perception system has transitioned from camera-based 2D perception to 3D perception. This transformation is attributed to the utilization of advanced Light Detection and Ranging (LiDAR) sensors, which provide depth information with relatively low computational demands [1]. Notably, Deep Learning (DL) networks trained on the point clouds from LiDAR exhibit significantly higher accuracy percentages in perception tasks such as 3D object detection, 3D scene segmentation, etc [2].

Recent studies emphasize that LiDAR sensing is susceptible to weather conditions and sensor-related errors known as common corruptions similar to the 2D image-based perception [3], [4]. Due to these distortions, LiDAR point clouds become sparse or noisy and directly affect downstream perception tasks, such as semantic segmentation and object detection. To address this issue, previous research shows potential in employing generative networks targeting weather conditions or point cloud completion to mitigate sparsity [5], [6], [7], [8]. However, the majority of these works primarily aim to reconstruct the corrupted point cloud without placing emphasis on downstream perception tasks.

As a potential solution, this study proposes a mask-guided blind diffusion inpainting approach to reconstruct and segment the corrupted point clouds. Notably, here we select segmentation since it is more complex than approximate region-based 3D object detection [9], and LiDAR segmentation using diffusion networks is an ill-posed research topic. The proposed solution comprises two sub-components: 1) a corrupted region prediction network and 2) a conditional mask-guided diffusion-based reconstruction/segmentation network. The performance of the proposed solution is assessed against six LiDAR-related corruptions, covering both sensor and weather corruptions, including those that generate noisy and sparse point clouds. Specifically, the main contributions of this study are as follows:

1) Develop an Attention U-Net [10] based binary segmentation network as a mask to detect regions corrupted by weather corruptions such as fog, snow, wet ground, and sensor errors including beam missing, incomplete echo, and cross-sensor issues.
2) Introduce a mask-guided denoising diffusion network to reconstruct the corrupted LiDAR point clouds.
3) Demonstrate that denoising diffusion networks yield promising results in segmenting corrupted point clouds.

The rest of the article is organized as follows: Section II discusses related works. Section III provides a detailed description of the proposed method, and Section IV focuses on providing information about the experimental setup. The quantitative and qualitative evaluation results, along with a benchmark against state-of-the-art methods, are provided in Section V. Finally, Section VI discusses the future research directions and concludes the paper.

II. RELATED WORKS

A. Common Corruptions against LiDAR Perception

Common corruptions affecting LiDAR perception can be divided into two main groups: i) external distortions, such as adverse weather conditions, and ii) internal sensor errors, like cross-sensor issues. Recently, several studies have emerged, benchmarking LiDAR-based deep learning networks' perception against these common corruptions [3], [11], [4]. These works concentrate on introducing LiDAR simulation methods under adverse weather conditions and providing datasets to evaluate the robustness of LiDAR perception against common corruptions.

* This research has been supported by a UNSW Tuition Fee Scholarship (TFS).

[1]K.T. Yasas Mahima, Sreenatha Anavatti, and Matt Garratt are with the School of Engineering and Technology, University of New South Wales, Canberra, Australia yasas.mahima, a.sreenatha, m.garratt{@unsw.edu.au}

[2]Asanka G. Perera is with the School of Engineering, University of Southern Queensland, Brisbane, Australia asanka.perera@unisq.edu.au

B. LiDAR Point Cloud Generation and Reconstruction

Exploring the use of generative networks for real-world LiDAR point clouds is a novel research area, as the majority of previous studies have concentrated on point cloud objects derived from CAD models. Lucas et al. [12] initially assessed the performance of Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) in generating realistic LiDAR samples. In [13] Nakashima et al. presented a GAN-based approach to generate LiDAR point clouds simulating the ray-casting and ray-dropping processes. The authors of both [12], [13], examined the applicability of their methods to sparse point cloud restoration tasks. Moreover, [14] presented a method based on Vector Quantized Variational Autoencoders (VAEs) to generate and complete LiDAR samples, encompassing tasks like sparse-to-dense completion and addressing partially observed point clouds affected by distortions such as sensor dirt.

Studies [5], [6], [15] have introduced CycleGAN-based approaches aimed at transferring point clouds from sunny conditions to those under snow, fog, or rainy conditions. However, the study by Zhang et al. [5] exclusively focuses on denoising aspects within point clouds under snowy conditions. In contrast, studies [6], [15] demonstrate a comparatively higher intensity error rate.

Recently, diffusion-based generative models have exhibited a higher success rate in image generation and reconstruction tasks. In a recent study [7], LiDARGen, a Noise Conditional Score Network (NCSN), was proposed for real-world LiDAR point cloud generation and the reconstruction of sparse point clouds under beam missing corruptions. Following LiDARGen, Nakashima et al. [8] first evaluated denoising diffusion probabilistic models (DDPM) for LiDAR generation and reconstruction under beam missing and wet ground corruptions.

When considering the point cloud reconstruction task, these two studies [8], [7] bear a minor resemblance to ours, and we aim to address two main limitations of these studies.
1) The first limitation is that in both methods, the densification task does not use a corrupted region mask as a condition during training; only the sampling process is modified to incorporate a mask. Therefore, the sampling performance is entirely dependent on the accuracy of the mask. However, in real-world scenarios, accurately observing a corrupted region mask is challenging, often allowing only an estimated observation. Therefore, to enhance the real-world applicability of the reconstruction network, it is essential to integrate the estimated mask as a condition in the training process and reduce the sampling process's dependency on the predicted mask.
2) The second limitation is that these two approaches are solely evaluated on sparse, point-related corruptions, whereas it is essential to evaluate and introduce a solution to reconstruct both sparse and noisy point clouds.

C. Denoising Diffusion Models for Perception Tasks

Recently, there has been an increased interest in exploring denoising diffusion models for perception tasks, including classification, object detection, segmentation, and depth estimation. Among these, our main focus is on segmentation tasks, as making classifications at the pixel/point level is challenging. Most of the state-of-the-art diffusion-based segmentation techniques are limited to binary segmentation. However, recent studies [16], [17], [18], [19] have proposed various diffusion architectures for multi-class semantic and panoptic segmentation. Among these, the study by Ji et al. [18] is the only work that utilizes LiDAR point clouds along with camera images to generate Bird's Eye View (BEV) segmentation maps.

Based on the literature, our study is the first to explore a mask-conditional diffusion-based blind inpainting method for reconstructing and segmenting real-world LiDAR point clouds.

III. 3DR-DIFF METHODOLOGY

In this section, we present, the proposed mask-conditional diffusion inpainting pipeline named 3DR-DIFF (See Figure 1), including its sub-modules and data representation methods. Notably, our work is closely related to the ShadowDiffusion study [20], which proposed to remove shadows from images.

A. Problem Formulation

The primary objective of 3DR-DIFF is to reconstruct sparse or noisy point clouds, restoring them to normal conditions, via an inpainting approach. This can be mathematically represented as follows: Let the original LiDAR point cloud and corrupted point cloud be denoted as $P_{org} \in R^{N\times 4}$ and $P_{cor} \in R^{N\times 4}$. The corrupted input point cloud under inpainting settings can then be formulated as follows:

$$P_{cor} = P_{org} \odot (1 - m) + C \odot m. \tag{1}$$

where $m \in R^{N\times 1}$ is the binary mask representing the corrupted region, C is the corrupted signal (either adding noisy points or removing points), and $\odot$ is the Hadamard product operator. The 3DR-DIFF network $\mathcal{F}$ learns to reconstruct the P_{cor} using the m as $\mathcal{F}(P_{cor}, m) \rightarrow P_{gen}$. Here, the primary objective of inputting m to the network is to enhance its understanding of the corrupted regions. The reconstruction process of 3DR-DIFF is a blind inpainting approach as it only uses an approximation of the corrupted region m.

B. LiDAR Data Representation

Generally, LiDAR point clouds are distributed in Euclidean space where a single LiDAR point cloud be represented using their Cartesian coordinates as $P_i = (x_i, y_i, z_i, r_i)$ where r_i is the intensity. We convert the LiDAR point cloud to a 2-channel image where one channel represents the depth and the other represents the intensity. In particular, each LiDAR point P_i in the Cartesian coordinate system is converted to the spherical coordinate system as $Z \in (\theta_i, \phi_i, d_i)$ where $\theta_i = arccos\frac{z_i}{\sqrt{x_i^2+y_i^2+z_i^2}}$, $\phi_i = atan2(y_i, x_i)$ and $d_i = \sqrt{x_i^2 + y_i^2 + z_i^2}$. The two-channel range image $I \in R^{H\times W\times 2}$ is then created by

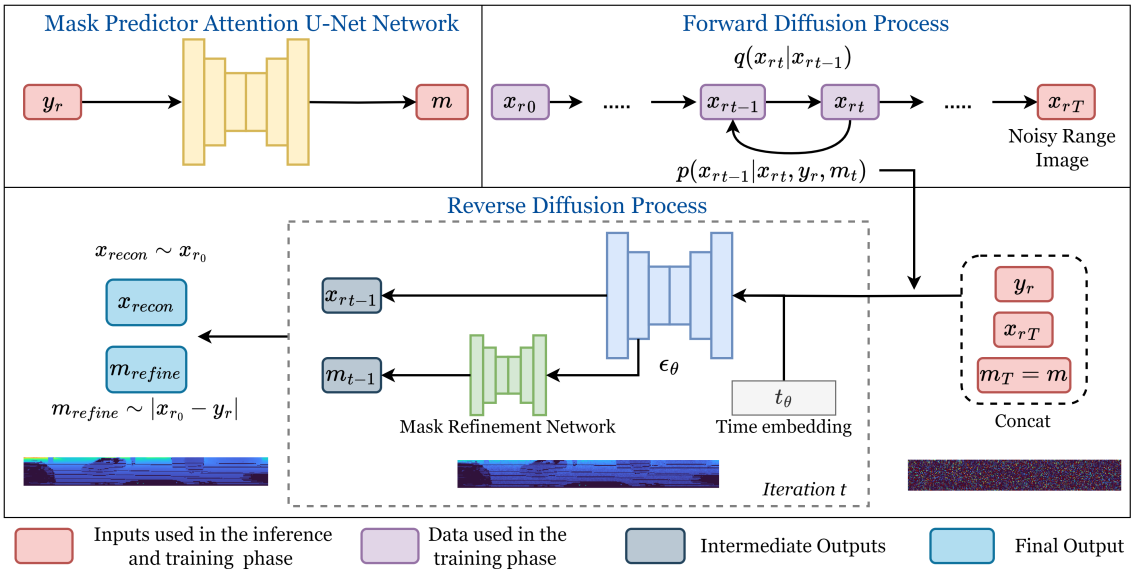

Fig. 1. LiDAR Point Cloud Reconstruction Process of the 3DR-DIFF.

projecting each point according to the $Z \in (\theta_i, \phi_i, d_i), I_{i,j} : I([\theta_i/s_\theta], [\phi_j/s_\phi]) = (\bar{d}_i, r_i)$ where $\bar{d}_i$ is the normalized d_i between [0,1] followed by LiDARGen [7], as $\bar{d}_i = \frac{1}{6} log_2(d_i + 1)$. In the following sections, we use x_r for non-corrupted and y_r for corrupted LiDAR range images.

C. Mask Predictor Network

In the context of real-world corruption, obtaining the m proves challenging due to the lack of knowledge about the uncorrupted point cloud at a given time. Consequently, we cannot precisely obtain the corrupted region using $|P_{org} - P_{cor}|$ to define the m. Hence, one possible option is to estimate the m. To approximate the value of m, we have used an Attention U-Net [10] based network. The main hypothesis of using Attention U-Nets is that incorporating attention gate mechanisms within the network enhances its ability to selectively focus on corrupted regions, surpassing feature activations in uncorrupted regions. This helps the network to discern corrupted regions distorted by noise or missing pixels, and gain better performance in approximating m.

In the attention mechanism, first, the input feature map of a corrupted point cloud range image y_{rf} and the gating signal G are first linearly mapped to a feature space with the dimensions of $R^{F \times H \times W}$. Next, the output is squeezed in the channel domain to obtain the attention weight map $W_{att} \in R^{1 \times H \times W}$. The final feature map after the attention function $F(y_{rf}, G)$ could be formulated as follows:

$$O_{feature} = y_{rf} \times \sigma(\phi_{y_{rf},G}(\delta(\phi_{y_{rf}}(y_{rf}) + \phi_G(G)))), \quad (2)$$

where $\phi_{y_{rf},G}$, $\phi_{y_{rf}}$ and ϕ_G are linear transformations with 1×1 convolutions while σ and δ are Sigmoid and ReLU activation functions respectively. The weighted binary cross-entropy loss is used to optimize the mask predictor network.

D. Mask Guided Diffusion Network

Similar to the image reconstruction tasks, having an approximate corrupted region mask is essential for reconstructing LiDAR images. This mask aids in distinguishing exact corrupted regions from ray-drop noise locations and naturally sparse point locations, such as the windows of vehicles.

In this study, we employ conditional denoising diffusion networks which consist of two main steps. The first step is the forward diffusion process which is a fixed Markov chain that gradually adds Gaussian noise to the input range image x_r according to a variance schedule $\beta_1 ... \beta_T$ as:

$$q(x_{rt}|x_{rt-1}) := \mathcal{N}(x_{rt}; \sqrt{1-\beta_t} x_{rt-1}, \beta_t I). \quad (3)$$

Using the simplification method proposed in the original DDPM paper [21], the intermediate x_{rt} could be directly obtained through Eq. 4 and in closed form, this can be expressed as shown in Eq. 5.

$$q(x_{rt}|x_{r0}) = \mathcal{N}(x_{rt}; \sqrt{\bar{\alpha}_t} x_{r0}, (1-\bar{\alpha}_t) I), \quad (4)$$

$$x_{rt} = \sqrt{\bar{\alpha}_t} x_{r0} + \sqrt{1-\bar{\alpha}_t} \epsilon. \quad (5)$$

where $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$ and $\epsilon \sim \mathcal{N}(0, I)$.

The second is the reverse diffusion process which learns to denoise the noisy sample back close to its original form at t time stamp based on a pre-defined condition y as $p_\theta(x_{rt-1}|x_{rt}, y)$. The reverse process of the diffusion network within T steps, conditioned by both the corrupted range image y_r and the predicted mask m by the proposed Attention U-Net network, is given in Eq. 6.

$$p_\theta(x_{r0:T}|y_r, m) = p(x_{rT}) \prod_{t=1}^{T} p_\theta(x_{rt-1}|x_{rt}, y_r, m). \quad (6)$$

Given that the reconstruction performance relies significantly on the predicted corrupted region mask, similar to [20], we integrate a predicted mask refinement block into the denoising network ϵ_θ and make it predict both the noise map e_t and the refined mask m_t, as depicted in Eq. 7.

$$e_t, m_t = \epsilon_\theta(\sqrt{\overline{\alpha_t}}x_{r0} + \sqrt{1-\overline{\alpha_t}}\epsilon, y_r, m, t). \quad (7)$$

During the training process, the denoising network is trained using the following joint objective.

$$\mathcal{L}_{diff} = \mathbb{E}_{x_{r0},t,\epsilon}|e_t - \epsilon| + \lambda\mathbb{E}_{t\sim[1,T]}|m_t - \bar{m}|. \quad (8)$$

Algorithm 1 Training Blind Diffusion Inpainting Network

Require: Corrupted range image $\boldsymbol{y_r}$, original range image $\boldsymbol{x_r}$, ground truth mask $\bar{\boldsymbol{m}}$
$m = MPN(y_r)$
while not converged **do**
  $t \sim$ Uniform($\{1...T\}$)
  $\epsilon \sim \mathcal{N}$(0,I)
  $e_t, m_t = \epsilon_\theta(\sqrt{\overline{\alpha_t}}x_{r0} + \sqrt{1-\overline{\alpha_t}}\epsilon, y_r, m, t)$
  Take a gradient descent step on $\nabla_\theta\mathcal{L}_{diff}(\theta)$
end while
return θ

Algorithm 2 Blind Diffusion Sampling

Require: Corrupted range image $\boldsymbol{y_r}$, number of sampling iterations T.
$m = MPN(y_r)$
$x_T \sim \mathcal{N}$(0,I)
for $t = T, ...1$ **do**
  $e_{t-1}, m_{t-1} = \epsilon_\theta(x_t, y_r, m, t)$
  if $t <= T \times 0.2$ **then** $m = m_{t-1}$
  $x_{t-1} = \sqrt{\overline{\alpha}_{t-1}}(\frac{x_t - \sqrt{1-\overline{\alpha}_t}.e_{t-1}}{\sqrt{\overline{\alpha}_t}}) + \sqrt{1-\overline{\alpha}_{t-1}}.e_{t-1}$
end for
return x_0

The architecture of the 3DR-DIFF is presented in Figure. 1 while the training process is detailed in Algorithm 1, and the inference reconstruction, using deterministic implicit sampling [22], is outlined in Algorithm 2.

E. Corrupted LiDAR Point Cloud Segmentation

We conduct three experiments on segmenting corrupted LiDAR point clouds as follows: First, we directly input 3DR-DIFF-reconstructed point clouds into a pre-trained range image-based LiDAR segmentation network. For this, we utilize the SqueezeSegV3 network [23], and we refer to this as 3DR-DIFF+S.

In the second study, we formulate 3DR-DIFF, conditioned by a corrupted range image y_r, a predicted corrupted region mask m, and a semantic mask of the point cloud s. Then, similar to the mask refinement block in 3DR-DIFF, we incorporate a semantic mask refinement block comprising five convolutional layers with ReLU activation where the final layer has 20 output channels (corresponding to the number of classes in the SemanticKITTI dataset). This block iteratively refines the predicted corrupted semantic mask to match the original non-corrupted semantic mask. We refer to this investigation as the Integrated Semantic 3DR-DIFF or in the short-term IS-3DR-DIFF. In order to obtain the initial semantic mask for the corrupted point cloud we use SqueezeSegV3 network [23]. The updated diffusion process could be formulated as Eq. 9 and the new training objective is as Eq. 10.

$$e_t, m_t, s_t = \epsilon_\theta(\sqrt{\overline{\alpha_t}}x_{r0} + \sqrt{1-\overline{\alpha_t}}\epsilon, y_r, m, s, t). \quad (9)$$

$$\bar{\mathcal{L}_{diff}} = \mathcal{L}_{diff} + \lambda_s\mathcal{L}_{CE}(s_t, \bar{s}), \quad (10)$$

where $\mathcal{L}_{CE}$ is the pixel-wise cross-entropy loss between the refined semantic mask s_t and the original semantic mask $\bar{s}$.

In the third study, we train another conditional diffusion network similar to 3DR-DIFF. This network utilizes a noisy semantic segmentation mask as the input and incorporates the predicted mask and corrupted LiDAR point cloud as conditions. To the best of our knowledge, there is no existing work that directly segments LiDAR point clouds using diffusion networks. This approach allowed us to conduct both segmentation and reconstruction in parallel without having any interference with each making the learning process simpler. Therefore, we named this method Parallel Semantic 3DR-DIFF, abbreviated as PS-3DR-DIFF.

Since diffusion networks require continuous data and operate as a regression task, they cannot be directly applied to tasks involving discrete data, such as semantic segmentation. As per the literature, one possible solution for this involves employing encoding methods such as Analog Bits encoding [24] and One-Hot encoding. In this study, we choose One-Hot encoding due to its simplicity. The updated diffusion process of PS-3DR-DIFF could be formulated as follows:

$$e_t, m_t = \epsilon_\theta(\sqrt{\overline{\alpha_t}}\varepsilon(s_0) + \sqrt{1-\overline{\alpha_t}}\epsilon, y_r, m, t), \quad (11)$$

where ε is the encoding function.

We experiment with two variants of PS-3DR-DIFF: i) utilizing a similar architecture to 3DR-DIFF, trained using L_{diff}, and ii) employing a deep network with additional downsampling and upsampling blocks, trained using $L_{diff} + L2(e_t, \epsilon)$, where $L2(e_t, \epsilon)$ represents the L2 distance between predicted and actual noise.

IV. EXPERIMENTAL SETUP

For the training, we utilize the SemanticKITTI-C [3] dataset introduced in the Robo3D benchmark. Specifically, the publicly available SemanticKITTI-C dataset is synthesized using the validation set of the SemanticKITTI dataset [25]. We use light, moderate, and high severity levels of LiDAR samples with snow, fog, wet ground, cross-sensor, beam missing, and incomplete echo as the common corruptions in this study. For a fair evaluation, in each

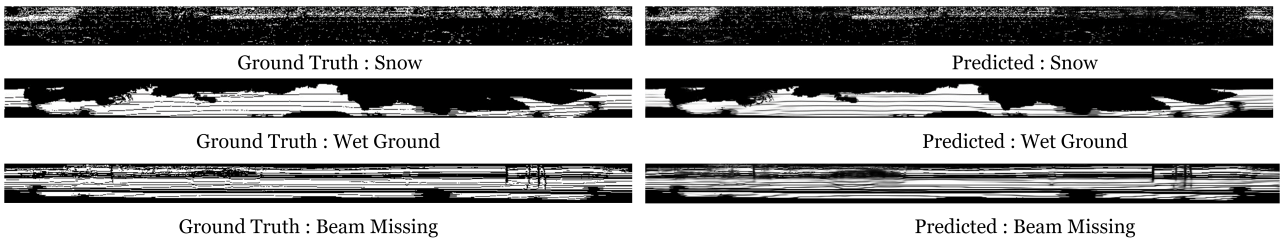

Fig. 2. Qualitative Results of the Mask Predictor Network.

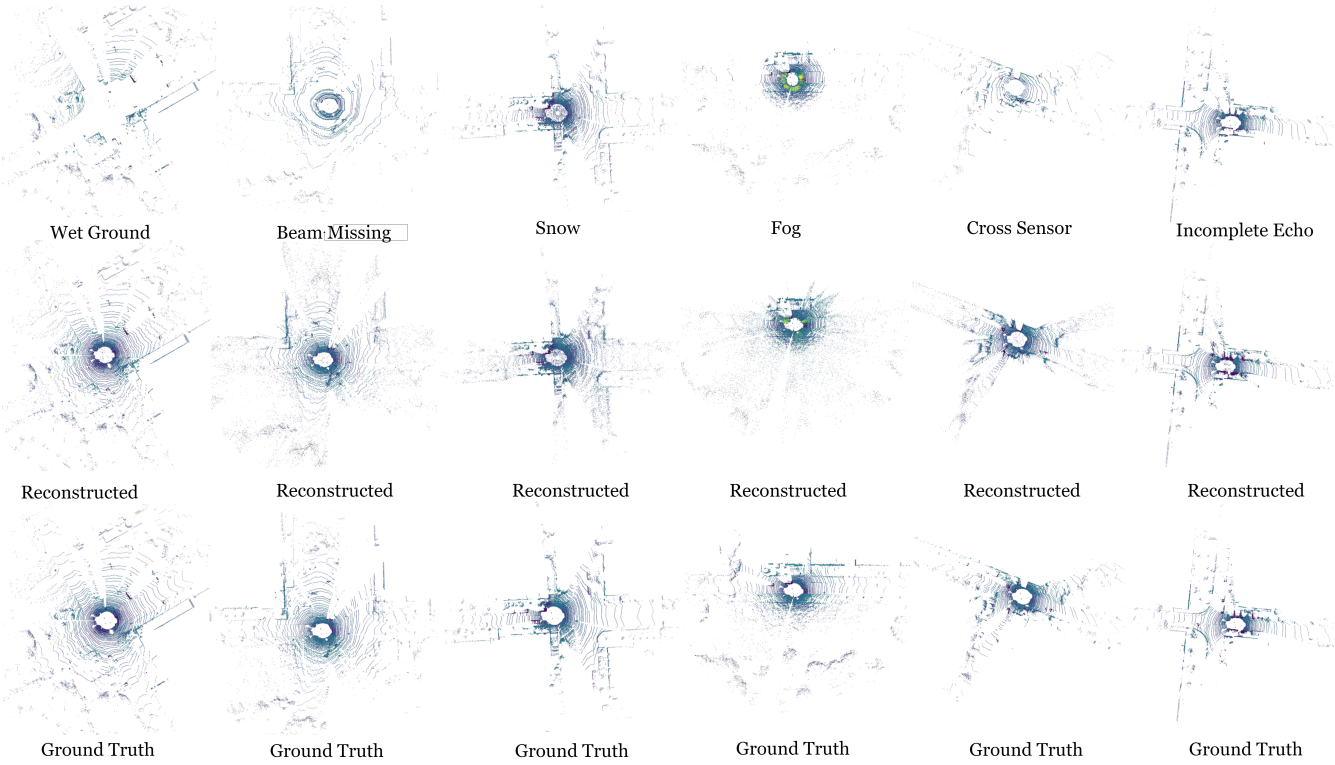

Fig. 3. Qualitative Results of the 3DR-DIFF Reconstructions.

severity of the selected corruptions, the first 3K samples are used for training. In the training process of the 3DR-DIFF reconstruction network, we augment the dataset by employing horizontal flipping. We further evaluate 3DR-DIFF trained on SemanticKITTI-C using LiDAR samples from KITTI-C [3], an extension of KITTI object detection [26]. Notably, while both datasets stem from KITTI, each has distinct LiDAR samples [27].

Our experiments are conducted using one RTX 3090 GPU. Both the corrupted region predictor and the 3DR-DIFF networks are trained with the Adam optimizer. Notably, we omit the intensity errors occurring under common corruptions and use the depth channel to calculate the ground truth mask $\bar{m}$ using the function presented in Eq. 12 for training the mask predictor network. The main reasons for this are to make the mask prediction task simpler, and most LiDAR-based perception networks focus on learning the geometry of point clouds rather than intensity values.

We followed the same U-Net architecture used in SR3 study [28], [20] with the Adam optimizer as the denoising network. We set 1000 diffusion steps for training.

$$f(x) = \begin{cases} 1, & \text{if } |x_r^{depth} - y_r^{depth}| > 0 \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

V. Evaluations

A. Evaluation Metrics

Jaccard similarity coefficient (mIOU) is used to assess both the masked corrupted region prediction network and semantic segmentation performance. During segmentation performance evaluation, we compute the mIOU across all 20 classes to emphasize and measure how 3DR-DIFF manages corrupted regions and addresses representation bias.

Similar to the previous works [8] we use Mean Absolute Error (MAE) and Route Mean Squared Error (RMSE) between generated and corrupted LiDAR range images to asses the 3DR-DIFF's reconstruction performance.

TABLE I

CORRUPTED REGION PREDICTION RESULTS.

Corruption	mIOU@P=0.5	mIOU@P=0.25
Beam Missing	0.859	0.879
Cross Sensor	0.742	0.772
Incomplete Echo	0.757	0.814
Wet Ground	0.919	0.932
Snow	0.764	0.791
Fog	0.431	0.433

TABLE II

3DR-DIFF RECONSTRUCTION RESULTS ON SEMANTICKITTI-C.

Corruption	Instance	MAE	RMSE	L1-D	L1-I
Beam Missing	Corrupted	1.92	5.55	3.74	0.10
	3DR-DIFF	0.22	1.65	0.43	0.02
Cross Sensor	Corrupted	3.00	7.27	5.84	0.16
	3DR-DIFF	0.36	2.26	0.68	0.04
Incomplete Echo	Corrupted	0.21	1.64	0.42	0.008
	3DR-DIFF	0.08	0.71	0.17	0.005
Wet Ground	Corrupted	1.06	3.41	2.06	0.06
	3DR-DIFF	0.07	0.45	0.13	0.01
Snow	Corrupted	0.61	3.70	1.19	0.032
	3DR-DIFF	0.35	2.27	0.67	0.032
Fog	Corrupted	1.29	5.24	2.51	0.07
	3DR-DIFF	0.66	3.13	1.30	0.03

B. Corrupted Region Approximation Results

Table I presents the mIOU for the masked corrupted region prediction network, employing the Sigmoid activation with probabilistic threshold values set at 0.25 and 0.5. These results demonstrate the effectiveness of the Attention U-Net network in accurately approximating corrupted regions, except for fog corruption. Specifically, it demonstrates an average mIOU rate of over 0.74 for all corruptions at both threshold levels. Moreover, Figure 2 illustrates qualitative results of the mask predictor network.

C. Reconstruction Results

To determine the optimal number of inference steps, we tested 3DR-DIFF on 100 samples from the test set for each type of corruption, using step sizes of 5, 10, and 25. We found that, except for fog, 3DR-DIFF effectively reconstructed other corruptions in approximately 5 steps. Therefore, considering inference efficiency, we selected 5 inference steps.

Table II presents the MAE and RMSE error of the intensity and depth channels before and after using 3DR-DIFF under each corruption. The results showcase the capability of 3DR-DIFF in accurately reconstructing sparse point-related corruptions while achieving reasonable performance in reconstructing noisy point-related corruptions, such as fog and snow. We present multiple reconstructed samples for each corruption using 3DR-DIFF, comparing them to the ground truth point cloud in Figure 3.

Table III shows the transferability assessment of 3DR-DIFF on the KITTI-C dataset. Notably, we exclude scenarios involving wet ground and incomplete echo, as the MAE between corrupted and original range images is quite low (Approximately 0.15 and 0.05, respectively) under those two corruptions. These results highlight the effective reconstruction capability of 3DR-DIFF for samples from an unseen dataset.

TABLE III

3DR-DIFF RECONSTRUCTION RESULTS ON KITTI-C.

Corruption	Instance	MAE	RMSE	L1-D	L1-I
Beam Missing	Corrupted	1.51	5.06	2.95	0.06
	3DR-DIFF	0.33	2.19	0.63	0.03
Cross Sensor	Corrupted	2.13	6.29	4.16	0.10
	3DR-DIFF	0.46	2.74	0.87	0.04
Snow	Corrupted	0.96	4.52	1.89	0.04
	3DR-DIFF	0.51	2.75	0.97	0.05
Fog	Corrupted	0.91	4.48	1.76	0.06
	3DR-DIFF	0.37	1.83	0.73	0.02

TABLE IV

RECONSTRUCTION RESULTS OF THE EXTENDED STUDIES.

Corruption	Instance	MAE	RMSE	L1-D	L1-I
Beam Missing	3DR-DIFF	0.22	1.65	0.43	0.02
	IS-3DR-DIFF	0.39	*1.51*	0.75	0.02
	F-3DR-DIFF	0.27	1.81	0.52	0.03
Cross Sensor	3DR-DIFF	0.36	2.26	0.68	0.04
	IS-3DR-DIFF	0.55	*2.07*	1.06	0.04
	F-3DR-DIFF	0.42	2.44	0.79	0.04
Incomplete Echo	3DR-DIFF	0.08	0.71	0.17	0.005
	IS-3DR-DIFF	0.32	0.84	0.62	0.009
	F-3DR-DIFF	0.16	0.87	0.33	0.007
Wet Ground	3DR-DIFF	0.07	0.45	0.13	0.01
	IS-3DR-DIFF	0.33	0.75	0.64	0.01
	F-3DR-DIFF	0.11	0.57	0.22	0.01
Snow	3DR-DIFF	0.35	2.27	0.67	0.032
	IS-3DR-DIFF	0.59	2.65	1.05	0.032
	F-3DR-DIFF	0.22	1.39	0.42	0.017
Fog	3DR-DIFF	0.66	3.13	1.30	0.03
	IS-3DR-DIFF	0.87	3.23	1.69	0.04
	F-3DR-DIFF	1.05	5.12	2.13	0.04

We further analyze the impact of mask feature concatenation. Specifically, drawing inspiration from VC-Net [29], we concatenate the bottleneck features of the mask predictor network with the downsampled features of the diffusion U-Net. Initially, features from the mask predictor network are upsampled using a transpose convolutional layer to match the U-net downsampled feature dimensions. This study is referred to as F-3DR-DIFF.

Table IV summarises the reconstruction performance of F-3DR-DIFF and IS-3DR-DIFF. The reconstruction performance of IS-3DR-DIFF is not as strong as that of 3DR-DIFF and under the incomplete echo scenario, it fares worse than the baseline. Moreover, under beam missing and cross-sensor corruptions, the IS-3DR-DIFF demonstrates lower RMSE values (Italic and Underlined), highlighting that IS-3DR-DIFF is sensitive to larger errors. We notice that this occurred because the network overfitted to the corrupted region more. F-3DR-DIFF demonstrates nearly equivalent reconstruction performance when compared to 3DR-DIFF, except under fog corruption. An interesting observation about F-3DR-DIFF is that it outperforms 3DR-DIFF under snow corruption, particularly at higher values as underlined.

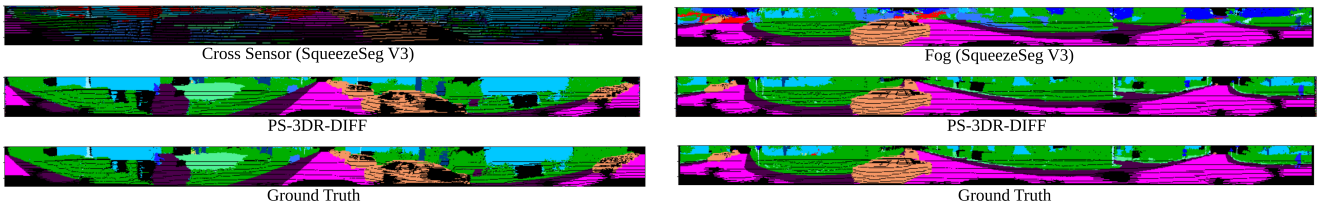

Fig. 4. Semantic Segmentation Results from the PS-3DR-DIFF ($L_{diff} + L2$).

TABLE V
SEMANTIC SEGMENTATION RESULTS.

Corruption	**Corr-upted**	**3DR-DIFF +S**	**IS-3DR-DIFF**	**PS-3DR-DIFF***	**PS-3DR-DIFF****
Beam Missing	0.18	0.34	0.33	0.36	0.43
Cross Sensor	0.07	0.30	0.25	0.34	0.41
Inc. Echo	0.31	0.35	0.55	0.38	0.44
Wet Ground	0.29	0.35	0.39	0.38	0.45
Snow	0.27	*0.24*	0.41	0.35	0.44
Fog	0.26	*0.22*	0.28	0.28	0.42

D. Segmentation Results

Table V presents the mIOU values for all 20 classes of 3DR-DIFF + SqueezeSegV3 (3DR-DIFF+S), IS-3DR-DIFF, and PS-3DR-DIFF, trained using L_{diff} loss (PS-3DR-DIFF*), and $L_{diff} + L2$ losses (PS-3DR-DIFF**), on the highest severity samples of each corruption. The SqueezeSegV3 network achieves an mIOU of 0.42 for our test split under non-corrupted instances. The results of 3DR-DIFF+S indicate that 3DR-DIFF enhances the performance of downstream segmentation networks, except under snow and fog conditions. In contrast, IS-3DR-DIFF and PS-3DR-DIFF, which employ diffusion networks, demonstrate overall better segmentation results when compared to the 3DR-DIFF+S as the slight distribution drift between the original and reconstructed samples affects the performance of the pre-trained SqueezeSegV3 network. Moreover, the deep PS-3DR-DIFF, trained on L_{diff} and $L2$ losses, exhibits a notable improvement in segmentation results, averaging 20% improvement (mIOU% difference) compared to segmentation results of corrupted point clouds using SqueezeSegV3 across all corruptions.

Based on the segmentation and reconstruction results of IS-3DR-DIFF (Tables IV and V), it is evident that learning to achieve both point cloud reconstruction and segmentation using a single network is a complex task. Hence, based on the results of PS-3DR-DIFF, we propose that employing two parallel diffusion networks for point cloud reconstruction and segmentation would be an optimal solution. Some qualitative samples of PS-3DR-DIFF are provided in Figure 4.

E. Benchmarking

To our knowledge, there is no existing work that predicts a mask for the corrupted region in LiDAR range images. Hence, we train residual blocks-based encoder-decoder mask predictor network employed in VC-Net [29] for LiDAR range images and use it as the baseline. Figure 5 depicts the benchmarking results for corrupted mask prediction, revealing that our Attention U-Net-based network outperforms the VC-Net's encoder-decoder network, except under fog.

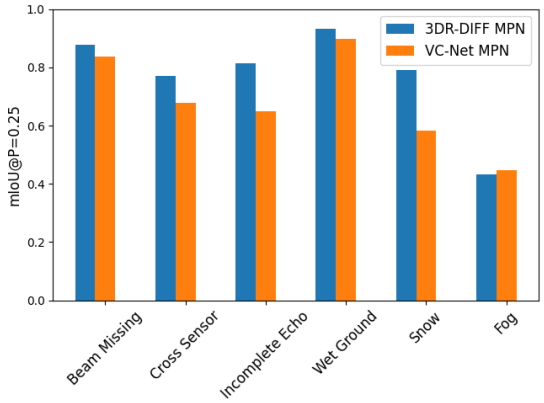

Fig. 5. Benchmarking Results for Corrupted Mask Prediction.

TABLE VI
BENCHMARKING RESULTS UNDER BEAM MISSING SCENARIO.

Study	R_{MAE}	R_{RMSE}	R_{L1-D}	R_{L1-I}
LiDARGen	56.4	32.7	57.3	34.5
R2DM	78.7	60.2	79.2	57.8
3DR-DIFF	**87.9**	**69.7**	**87.9**	**72.8**

We benchmark 3DR-DIFF for sparse point reconstruction, comparing its performance against LiDARGen [7] and R2DM [8]. We specifically benchmark the performance of 16-beam LiDAR densification, similar to reconstructing the highest severity of beam-missing scenarios in the Robo3D dataset. We use publicly available pre-trained weights and default configurations of these networks, originally trained on the KITTI-360 [30] dataset, as retraining them on the SemanticKITTI dataset is time-consuming. We utilize the benchmarking metric, Robustness Increment Per Error of 1%, given by $R_{Acc} = \frac{ACC_{cor} - ACC_{Rec}}{ACC_{cor}}\%$[1], where ACC_{cor} could be either MSE or RMSE. Table VI summarises the benchmarking results under the beam-missing scenario, highlighting that even when using an approximated mask, 3DR-DIFF outperforms both R2DM and LiDARGen.

When considering the inference sampling speed, LiDARGen took around 23 hours, R2DM took around 4 hours, while the entire 3DR-DIFF, including the mask prediction

[1]Getting higher is better.

network, only took around 3 minutes to sample 1071 LiDAR range images. This is because 3DR-DIFF uses implicit sampling, whereas R2DM employs 32 DDPM steps with 16 harmonization steps under each DDPM step.

VI. CONCLUSIONS

This paper proposes 3DR-DIFF, a masked guided blind denoising diffusion network designed to reconstruct LiDAR point clouds corrupted by six common types of corruptions, leading to sparsity or noise in the LiDAR point cloud. Our evaluation results demonstrate that 3DR-DIFF can accurately reconstruct corruptions involving beam missing, cross-sensor, incomplete echo, and wet ground scenarios, while the reconstruction results for snow and fog corruptions are moderate. Furthermore, we experiment and demonstrate that employing another conditional diffusion network, trained to segment the corrupted LiDAR point clouds in parallel with the 3DR-DIFF, can achieve relatively high segmentation performance. Moreover, our findings indicate that feature fusion improves the reconstruction performance against snow corruption. Potential future studies include incorporating latent diffusion mechanisms and novel feature fusion approaches to enhance reconstruction and segmentation performance. Additionally, it is crucial to assess the applicability of diffusion networks in enhancing robustness against point-based adversarial attacks and point-shifting-based corruptions.

REFERENCES

[1] Y. Mahima, A. Perera, S. Anavatti, and M. Garratt, "Towards robust 3d perception for autonomous vehicles: A review of adversarial attacks and countermeasures," 2023.

[2] Y. Li, L. Ma, Z. Zhong, F. Liu, M. A. Chapman, D. Cao, and J. Li, "Deep learning for LiDAR point clouds in autonomous driving: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 8, pp. 3412–3432, 2020.

[3] L. Kong, Y. Liu, X. Li, R. Chen, W. Zhang, J. Ren, L. Pan, K. Chen, and Z. Liu, "Robo3D: Towards robust and reliable 3D perception against corruptions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 994–20 006.

[4] X. Yan, C. Zheng, Z. Li, S. Cui, and D. Dai, "Benchmarking the robustness of LiDAR semantic segmentation models," *arXiv preprint arXiv:2301.00970*, 2023.

[5] Y. Zhang, M. Ding, H. Yang, Y. Niu, Y. Feng, K. Ohtani, and K. Takeda, "L-DIG: A GAN-based method for LiDAR point cloud processing under snow driving conditions," *Sensors*, vol. 23, no. 21, p. 8660, 2023.

[6] J. Lee, D. Shiotsuka, T. Nishimori, K. Nakao, and S. Kamijo, "GAN-based LiDAR translation between sunny and adverse weather for autonomous driving and driving simulation," *Sensors*, vol. 22, no. 14, p. 5287, 2022.

[7] V. Zyrianov, X. Zhu, and S. Wang, "Learning to generate realistic LiDAR point clouds," in *European Conference on Computer Vision*. Springer, 2022, pp. 17–35.

[8] K. Nakashima and R. Kurazume, "LiDAR data synthesis with denoising diffusion probabilistic models," *arXiv preprint arXiv:2309.09256*, 2023.

[9] V. Lakshmanan, M. Görner, and R. Gillard, *Practical machine learning for computer vision*. " O'Reilly Media, Inc.", 2021.

[10] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz *et al.*, "Attention U-Net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999*, 2018.

[11] M. Hahner, C. Sakaridis, M. Bijelic, F. Heide, F. Yu, D. Dai, and L. Van Gool, "LiDAR snowfall simulation for robust 3D object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 364–16 374.

[12] L. Caccia, H. Van Hoof, A. Courville, and J. Pineau, "Deep generative modeling of LiDAR data," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5034–5040.

[13] K. Nakashima, Y. Iwashita, and R. Kurazume, "Generative range imaging for learning scene priors of 3D LiDAR data," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 1256–1266.

[14] Y. Xiong, W.-C. Ma, J. Wang, and R. Urtasun, "Learning compact representations for LiDAR completion and generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1074–1083.

[15] J. Lee, D. Shiotsuka, T. Nishimori, K. Nakao, and S. Kamijo, "LiDAR translation based on empirical approach between sunny and foggy for driving simulation," in *25th International Symposium on Wireless Personal Multimedia Communications*. IEEE, 2022, pp. 430–435.

[16] B. Kolbeinsson and K. Mikolajczyk, "Multi-class segmentation from aerial views using recursive noise diffusion," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 8439–8449.

[17] T. Chen, L. Li, S. Saxena, G. Hinton, and D. J. Fleet, "A generalist framework for panoptic segmentation of images and videos," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 909–919.

[18] Y. Ji, Z. Chen, E. Xie, L. Hong, X. Liu, Z. Liu, T. Lu, Z. Li, and P. Luo, "DDP: Diffusion model for dense visual prediction," *arXiv preprint arXiv:2303.17559*, 2023.

[19] H. Wang, J. Cao, R. M. Anwer, J. Xie, F. S. Khan, and Y. Pang, "DFormer: Diffusion-guided transformer for universal image segmentation," *arXiv preprint arXiv:2306.03437*, 2023.

[20] L. Guo, C. Wang, W. Yang, S. Huang, Y. Wang, H. Pfister, and B. Wen, "Shadowdiffusion: When degradation prior meets diffusion model for shadow removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 049–14 058.

[21] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[22] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[23] C. Xu, B. Wu, Z. Wang, W. Zhan, P. Vajda, K. Keutzer, and M. Tomizuka, "SqueezeSegV3: Spatially-adaptive convolution for efficient point-cloud segmentation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*. Springer, 2020, pp. 1–19.

[24] T. Chen, R. Zhang, and G. Hinton, "Analog bits: Generating discrete data using diffusion models with self-conditioning," *arXiv preprint arXiv:2208.04202*, 2022.

[25] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9297–9307.

[26] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[27] J. Behley, A. Milioto, and C. Stachniss, "A benchmark for LiDAR-based panoptic segmentation based on KITTI," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 596–13 603.

[28] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4713–4726, 2022.

[29] Y. Wang, Y.-C. Chen, X. Tao, and J. Jia, "VCNet: A robust approach to blind image inpainting," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*. Springer, 2020, pp. 752–768.

[30] Y. Liao, J. Xie, and A. Geiger, "KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2D and 3D," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3292–3310, 2022.