Robo3D: Towards Robust and Reliable 3D Perception against Corruptions

Lingdong Kong[1,2,3,*] Youquan Liu[1,4,*] Xin Li[1,5,*] Runnan Chen[1,6] Wenwei Zhang[1,7]
Jiawei Ren[7] Liang Pan[7] Kai Chen[1] Ziwei Liu[7,✉]

[1]Shanghai AI Laboratory [2]National University of Singapore [3]CNRS@CREATE [4]Hochschule Bremerhaven
[5]East China Normal University [6]The University of Hong Kong [7]S-Lab, Nanyang Technological University

{konglingdong,liuyouquan,lixin,zhangwenwei,chenkai}@pjlab.org.cn {jiawei011,liang.pan,ziwei.liu}@ntu.edu.sg

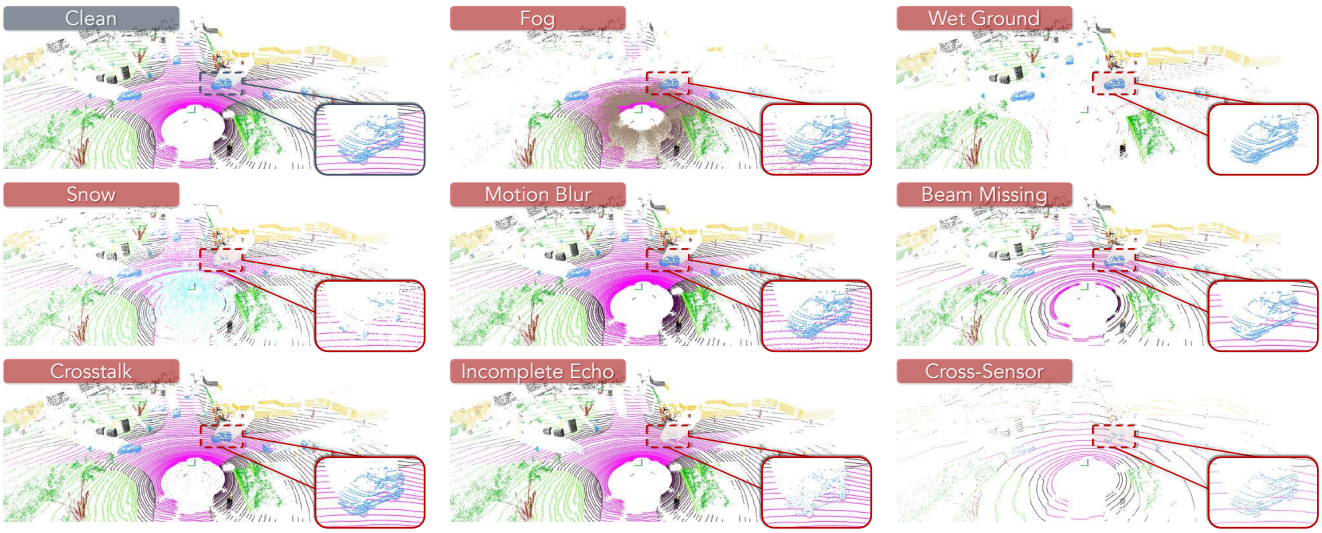


Figure 1: Taxonomy of the **Robo3D** benchmark. We simulate ***eight*** corruption types from ***three*** categories: **1)** *Severe weather conditions*, such as fog, rain, and snow; **2)** *External disturbances* that are caused by motion blur or result in the missing of LiDAR beams; and **3)** *Internal sensor failure*, including LiDAR crosstalk, possible incomplete echo, and cross-sensor scenarios. Each corruption is further split into ***three*** levels (light, moderate, and heavy) based on its severity.

Abstract

*The robustness of 3D perception systems under natural corruptions from environments and sensors is pivotal for safety-critical applications. Existing large-scale 3D perception datasets often contain data that are meticulously cleaned. Such configurations, however, cannot reflect the reliability of perception models during the deployment stage. In this work, we present **Robo3D**, the first comprehensive benchmark heading toward probing the robustness of 3D detectors and segmentors under out-of-distribution scenarios against natural corruptions that occur in real-world environments. Specifically, we consider eight corruption types stemming from severe weather conditions, external disturbances, and internal sensor failure. We uncover that, although promising results have been progressively achieved on standard benchmarks, state-of-the-art 3D perception models are at risk of being vulnerable to corruptions. We draw key observations on the use of data representations, augmentation schemes, and training strategies, that could severely affect the model's performance. To pursue better robustness, we propose a density-insensitive training framework along with a simple flexible voxelization strategy to enhance the model resiliency. We hope our benchmark and approach could inspire future research in designing more robust and reliable 3D perception models. Our robustness benchmark suite is publicly available*[1].

1. Introduction

3D perception aims to detect and segment accurate position, orientation, semantics, and temporary relation of

(*) The first three authors contributed equally to this work.

[1]https://github.com/ldkong1205/Robo3D.

the objects and backgrounds around the ego-vehicle in the three-dimensional world [3, 19, 25]. With the emergence of large-scale autonomous driving perception datasets, various approaches in the fields of LiDAR semantic segmentation and 3D object detection advent each year, with record-breaking performances on the mainstream benchmarks [18, 4, 8, 17, 61].

Despite the great success achieved on the "clean" evaluation sets, the model's robustness against out-of-distribution (OoD) scenarios remain obscure. Recent attempts mainly focus on probing the OoD robustness from two aspects. The first line focuses on the transfer of 3D perception models to unseen domains, *e.g.*, sim2real [72], day2night [26], and city2city [30] adaptations, to probe the model's generalizability. The second line aims to design adversarial examples which can cause the model to make incorrect predictions while keeping the attacked input close to its original format, *i.e.*, to test the model's worst-case scenarios [50, 9, 65].

In this work, different from the above two directions, we aim at understanding the cause of performance deterioration under real-world corruption and sensor failure. Current 3D perception models learn point features from LiDAR sensors or RGB-D cameras, where data corruptions are inevitable due to issues of data collection, processing, weather conditions, and scene complexity [49]. While recent works target creating corrupted point clouds from indoor scenes [28] or object-centric CAD models [60, 89, 2], we simulate corruptions on large-scale LiDAR point clouds from the complex outdoor driving scenes [18, 4, 8, 61].

As shown in Fig. 1, we consider three distinct corruption sources that are with a high likelihood to occur in real-world deployment: ***1) Severe weather conditions*** (*fog*, *rain*, and *snow*) which cause back-scattering, attenuation, and reflection of the laser pulses [21, 20, 59]; ***2) External disturbances***, *e.g.*, bumpy surfaces, dust, insects, that often lead to nonnegligible *motion blur* and LiDAR *beam missing* issues [46]; and ***3) Internal sensor failure***, such as the *incomplete echo* or miss detection of instances with a dark color (*e.g.*, black car) and *crosstalk* among multiple sensors, which likely deteriorates the 3D perception accuracy [83, 7]. Besides the environmental factors, it is also important to understand the *cross-sensor* discrepancy to avoid sudden failure caused by the sensor configuration change.

To properly fulfill such pursues, we simulate physically-principled corruptions on the *val* sets of KITTI [18], SemanticKITTI [4], nuScenes [8], and Waymo Oepn [61], as our corruption suite dubbed ***Robo3D***. Analogous to the popular 2D corruption benchmarks [23, 81, 41], we create three severity levels for each corruption and design suitable metrics as the main indicator for robustness comparisons. Finally, we conduct exhaustive experiments to understand the pros and cons of different designs from existing models. We observe that modern 3D perception models are at risk of being vulnerable even though their performance on standard benchmarks is improving. Through fine-grained analyses on a wide range of 3D perception datasets, we diagnose that:

- *Sensor setups have direct impacts on feature learning*. 3D perception models trained on data collected with different sensor configurations and protocols often yield inconsistent resilience.
- *3D data representations often coupled with the model's robustness*. The voxel and point-voxel fusion approaches exhibit clear superiority over the projection-based methods, *e.g.*, range view.
- *3D detectors and segmentors show distinct sensitivities to different corruption types*. A sophisticated combination of both tasks is a viable way to achieve robust and reliable 3D perception.
- *Out-of-context augmentation (OCA) and flexible rasterization strategies can improve model's robustness*. We thus propose a solution to enhance the robustness of existing 3D perception models, which consists of a density-insensitive training framework and a simple flexible voxelization strategy.

The key contributions of this work are summarized as:

- We introduce Robo3D, the first systematically-designed robustness evaluation suite for LiDAR-based 3D perception under corruptions and sensor failure.
- We benchmark 34 perception models for LiDAR-based semantic segmentation and 3D object detection tasks, on their robustness against corruptions.
- Based on our observations, we draw in-depth discussions on the design receipt and propose novel techniques for building more robust 3D perception models.

2. Related Work

LiDAR-based Semantic Segmentation. The design choice of 3D segmentors often correlates with the LiDAR representations, which can be categorized into raw point [64], range view [69, 42, 29], bird's eye view [85], voxel [12], and multi-view fusion [38, 74] methods. The projection-based approach rasterizes irregular point clouds into 2D grids, which avoids the need for 3D operators and is thus more hardware-friendly for deployment [13, 86, 11]. The voxel-based methods which retain the 3D structure are achieving better performance than other single modalities [90, 80]. Efficient operators like the sparse convolution are widely adopted to ease the memory footprint [63, 62]. Most recently, some works start to explore possible complementary between two views [38, 75, 48] or even more views [74]. Although promising results have been achieved, the

robustness of 3D segmentors against corruptions remains obscure. As we will discuss in the next sections, these methods have the tendency of being less robust, mainly due to the lack of a comprehensive robustness evaluation benchmark.

LiDAR-based 3D Object Detection. Sharing similar basics with LiDAR segmentation, modern 3D object detectors also adopt various data representations. Point-based methods [56, 58, 79, 78] implicitly capture local structures and fine-grained patterns without any quantization to retain the original point cloud geometry. Voxel-based methods [77, 87, 82, 57, 35, 37, 34, 39] transform irregular point clouds to compact grids while only those non-empty voxels are stored and utilized for feature extraction through the sparse convolution [77]. Recently, some works [40, 88] start to explore long-range contextual dependencies among voxels with self-attention [66]. The pillar-based methods [32, 52] better balance the accuracy and speed by controlling the resolution in the vertical axis. The point-voxel fusion methods [54, 53] can integrate the merits of both representations to learn more discriminative features. The above methods, however, mainly focused on obtaining better performance on clean point clouds, while paying much less attention to the model's robustness. As we will show in the following sections, these models are prone to degradation under data corruptions and sensor failure.

Common Corruptions. The corruption robustness often refers to the capability of a conventionally trained model for maintaining satisfactory performance under natural distribution shifts. ImageNet-C [23] is the pioneering work in this line of research which benchmarks image classification models to common corruptions and perturbations. Follow-up studies extend on a similar aspect to other perception tasks, *e.g.*, object detection [41], image segmentation [27], navigation [10], video classification [81], and pose estimation [67]. The importance of evaluating the model's robustness against corruptions has been constantly proven. Since we are targeting a different sensor, *i.e.*, LiDAR, most of the well-studied corruption types – such as those designed for camera malfunctions – become unrealistic or unsuitable for such a data format. This motivates us to explore new taxonomy for defining more proper corruption types for the 3D perception tasks in autonomous driving scenarios.

3D Perception Robustness. Several recent attempts proposed to investigate the vulnerability of point cloud classifiers and detectors in indoor scenes [28, 49, 60, 89, 2]. Recently, there are works started to explore the robustness of 3D object detectors under adversarial attacks [45, 65, 73]. In the context of corruption robustness, we notice several concurrent works [83, 33, 1, 15, 76]. These works, however, all consider a single task alone and might be constrained by either a limited number of corruption types or candidate datasets. Our benchmark properly defines a more diverse range of corruption types for the general 3D perception task and includes significantly more models from both LiDAR-based semantic segmentation and 3D object detection tasks.

3. The Robo3D Benchmark

Tailored for LiDAR-based 3D perception tasks, we summarize eight corruption types commonly occurring in real-world deployment in our benchmark, as shown in Fig. 1. This section elaborates on the detailed definition of each corruption type (Sec. 3.1), configurations of different robustness simulation sets (Sec. 3.2), and evaluation metrics for robustness measurements (Sec. 3.3).

3.1. Corruption Types

Given a point $\mathbf{p} \in \mathbb{R}^4$ in a LiDAR point cloud with coordinates (p^x, p^y, p^z) and intensity p^i, our goal is to simulate a corrupted point $\hat{\mathbf{p}}$ via a mapping $\hat{\mathbf{p}} = \mathcal{C}(\mathbf{p})$, with rules constrained by *physical principles* or *engineering experiences*. Due to space limits, We present more detailed definitions and implementation procedures of our corruption simulation algorithms in the Appendix.

1) Fog. The LiDAR sensor emits laser pulses for accurate range measurement. Back-scattering and attenuation of LiDAR points tend to happen in foggy weather since the water particles in the air will cause inevitable pulse reflection [5, 6]. In our benchmark, we adopt the physically valid fog simulation method [21] to create fog-corrupted data. For each $\mathbf{p}$, we calculate its attenuated response $p^{i_{\text{hard}}}$ and the maximum fog response $p^{i_{\text{soft}}}$ as follows:

$$p^{i_{\text{hard}}} = p^i e^{-2\alpha\sqrt{(p^x)^2+(p^y)^2+(p^z)^2}} , \tag{1}$$

$$p^{i_{\text{soft}}} = p^i \frac{(p^x)^2 + (p^y)^2 + (p^z)^2}{\beta_0} \beta_{\text{bs}} \times p^i_{\text{tmp}} , \tag{2}$$

$$\hat{\mathbf{p}} = \mathcal{C}_{\text{fog}}(\mathbf{p}) = \begin{cases} (\hat{p}^x, \hat{p}^y, \hat{p}^z, p^{i_{\text{soft}}}), & \text{if } p^{i_{\text{soft}}} > p^{i_{\text{hard}}} , \\ (p^x, p^y, p^z, p^{i_{\text{hard}}}), & \text{else} . \end{cases} \tag{3}$$

where α is the attenuation coefficient, β_{bs} denotes the back-scattering coefficient, β_0 describes the differential reflectivity of the target objects, and the p^i_{tmp} symbol is the received response for the soft target term.

2) Wet Ground. The emitted laser pulses will likely lose certain amounts of energy when hitting wet surfaces, which causes significantly attenuated laser echoes depending on the water height d_w and mirror refraction rate [59]. We follow [20] to model the attenuation caused by ground wetness. A pre-processing step is taken to estimate the ground plane with existing semantic labels or RANSAC [16]. Next, a ground plane point of its measured intensity $\hat{p}^i$ is obtained based on the modified reflectivity, and the point is only kept

if its intensity is greater than the noise floor i_n via the following mapping:

$$\mathcal{C}_{\text{wet}}(\mathbf{p}) = \begin{cases} (p^x, p^y, p^z, \hat{p}^i), & \text{if} \quad \hat{p}^i > i_n \ \& \ \mathbf{p} \in \text{ground} \ , \\ \text{None}, & \text{elif} \ \hat{p}^i < i_n \ \& \ \mathbf{p} \in \text{ground} \ , \\ (p^x, p^y, p^z, p^i), & \text{elif} \ \mathbf{p} \notin \text{ground} \ . \end{cases} \tag{4}$$

3) Snow. For each laser beam in snowy weather, the set of particles in the air will intersect with it and derive the angle of the beam cross-section that is reflected by each particle, taking potential occlusions into account [51]. We follow [20] to simulate snow-corrupted data $\mathcal{C}_{\text{snow}}(\mathbf{p})$ which is similar to the fog simulation. This physically-based method samples snow particles in the 2D space and modify the measurement for each LiDAR beam in accordance with the induced geometry, where the number of sampling snow particles is set according to a given snowfall rate r_s.

4) Motion Blur. Since the LiDAR sensor is often mounted on the roof-top or side of the vehicle, it inevitably suffers from the blur caused by vehicle movement, especially on bumpy surfaces or during U-turning. To simulate blur-corrupted data $\mathcal{C}_{\text{motion}}(\mathbf{p})$, we add a jittering noise to each coordinate (p^x, p^y, p^z) with a translation value sampled from the Gaussian distribution with standard deviation σ_t. This simulation process is shown as follows:

$$\mathcal{C}_{\text{motion}}(\mathbf{p}) = (p^x + o_1, p^y + o_2, p^z + o_3, p^i) \ , \tag{5}$$

where o_1, o_2, o_3 are the random offsets sampled from Gaussian distribution $N \in \{0, {\sigma_t}^2\}$ and $\{o_1, o_2, o_3\} \in \mathbb{R}^{1\times1}$.

5) Beam Missing. The dust and insect tend to form agglomerates in front of the LiDAR surface and will not likely disappear without human intervention, such as drying and cleaning [46]. This type of occlusion causes zero readings on masked areas and results in the loss of certain light impulses. To mimic such a behavior, we randomly sample a total number of m beams and drop points on these beams from the original point cloud to generate $\mathcal{C}_{\text{beam}}(\mathbf{p})$:

$$\mathcal{C}_{\text{beam}}(\mathbf{p}) = \begin{cases} (p^x, p^y, p^z, p^i), & \text{if} \quad \mathbf{p} \notin m \ , \\ \text{None}, & \text{else} \ . \end{cases} \tag{6}$$

6) Crosstalk. Considering that the road is often shared by multiple vehicles, the time-of-flight of light impulses from one sensor might interfere with impulses from other sensors within a similar frequency range [7]. Such a crosstalk phenomenon often creates noisy points within the mid-range areas in between two (or multiple) sensors. To simulate this corruption $\mathcal{C}_{\text{cross}}(\mathbf{p})$, we randomly sample a subset of k_t percent points from the original point cloud and add large jittering noise with a translation value sampled from the Gaussian distribution with standard deviation σ_c. This simulation process is shown as follows:

$$\mathcal{C}_{\text{cross}}(\mathbf{p}) = \begin{cases} (p^x, p^y, p^z, p^i), & \text{if} \ \ \mathbf{p} \notin \text{set of} \ \{k_t\} \ , \\ (p^x, p^y, p^z, p^i) + \xi_c, & \text{else} \ , \end{cases} \tag{7}$$

where ξ_c is the random offset sampled from Gaussian distribution $N \in \{0, {\sigma_c}^2\}$ and $\xi_c \in \mathbb{R}^{1\times4}$.

7) Incomplete Echo. The near-infrared spectrum of the laser pulse emitted from the LiDAR sensor is vulnerable to vehicles or other instances with dark colors [83]. The LiDAR readings are thus incomplete in such scan echoes, resulting in significant point miss detection. We simulate this corruption which denotes $\mathcal{C}_{\text{echo}}(\mathbf{p})$ by randomly querying k_e percent points for *vehicle*, *bicycle*, and *motorcycle* classes, via either semantic masks or 3D bounding boxes. Next, we drop the queried points from the original point cloud, along with their point-level semantic labels. Note that we do not alter the ground-truth bounding boxes since they should remain at their original positions in the real world. The overall operation can be summarized as follows:

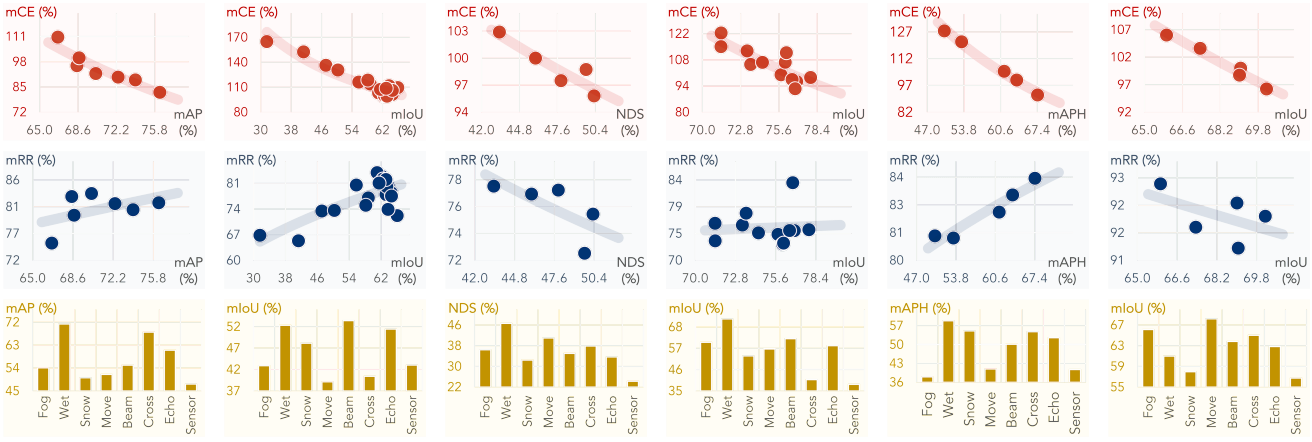
$$\mathcal{C}_{\text{echo}}(\mathbf{p}) = \begin{cases} (p^x, p^y, p^z, p^i), & \text{if} \ \ \mathbf{p} \notin \text{set of} \ \{k_e\} \ , \\ \text{None}, & \text{else} \ . \end{cases} \tag{8}$$

8) Cross-Sensor. Due to the large variety of LiDAR sensor configurations (*e.g.*, beam number, FOV, and sampling frequency), it is important to design robust 3D perception models that are capable of maintaining satisfactory performance under cross-device cases [79]. While previous works directly form such settings with two different datasets, the domain idiosyncrasy in between (*e.g.* different label mappings and data collection protocols) further hinders the direct robustness comparison. In our benchmark, we follow [68] and generate cross-sensor data $\mathcal{C}_{\text{sensor}}(\mathbf{p})$ by first dropping points of certain beams from the point cloud and then sub-sample k_c percent points from each beam. This simulation process is shown as follows:

$$\mathcal{C}_{\text{sensor}}(\mathbf{p}) = \begin{cases} \text{None}, & \text{if} \ \ \mathbf{p} \in \text{set of} \ \{k_c\} \ , \\ (p^x, p^y, p^z, p^i), & \text{else} \ . \end{cases} \tag{9}$$

3.2. Corruption Sets

Following the above taxonomy, we create new robustness evaluation sets upon the *val* sets of existing large-scale 3D perception datasets [18, 4, 8, 17, 61] to fulfill *SemanticKITTI-C*, *KITTI-C*, *nuScenes-C*, and *WOD-C*. They are constructed with eight corruption types under three severity levels, resulting in a total number of 97704, 90456, 144456, and 143424 annotated LiDAR point clouds, respectively. Kindly refer to the Appendix for more details in terms of these robustness evaluation collections.



(a) KITTI-C (b) SemanticKITTI-C (c) nuScenes-C (Det3D) (d) nuScenes-C (Seg3D) (e) WOD-C (Det3D) (f) WOD-C (Seg3D)

Figure 2: Benchmarking results of ***34*** LiDAR-based detection and segmentation models on the ***six*** robustness sets in Robo3D. Figures from top to bottom: the task-specific accuracy (mAP, mIoU, NDS, mAPH) *vs.* **[first row]** mean corruption error (mCE), **[second row]** mean resilience rate (mRR), and **[third row]** sensitivity analysis among different corruption types.

3.3. Evaluation Metrics

Corruption Error (CE). We follow [23] and use the mean CE (mCE) as the primary metric in comparing models' robustness. To normalize the severity effects, we choose CenterPoint [82] and MinkUNet [63] as the baseline models for the 3D detectors and segmentors, respectively. The CE and mCE scores are calculated as follows:

$$\text{CE}_i = \frac{\sum_{l=1}^{3}(1 - \text{Acc}_{i,l})}{\sum_{l=1}^{3}(1 - \text{Acc}_{i,l}^{\text{baseline}})}, \quad \text{mCE} = \frac{1}{N}\sum_{i=1}^{N}\text{CE}_i, \tag{10}$$

where $\text{Acc}_{i,l}$ denotes the task-specific accuracy scores, *i.e.*, mIoU for LiDAR semantic segmentation, and AP, NDS, or APH(L2) for 3D object detection, on corruption type i at severity level l. $N = 8$ is the total number of corruption types.

Resilience Rate (RR). We define mean RR (mRR) as the relative robustness indicator for measuring how much accuracy can a model retain when evaluated on the corruption sets. The RR and mRR scores are calculated as follows.

$$\text{RR}_i = \frac{\sum_{l=1}^{3}\text{Acc}_{i,l}}{3 \times \text{Acc}_{\text{clean}}}, \quad \text{mRR} = \frac{1}{N}\sum_{i=1}^{N}\text{RR}_i, \tag{11}$$

where $\text{Acc}_{\text{clean}}$ denotes the task-specific accuracy score on the "clean" evaluation set.

4. Experimental Analysis

4.1. Benchmark Configuration

3D Perception Models. We benchmark 34 LiDAR-based detection and segmentation models and variants. ***Detectors:*** SECOND [77], PointPillars [32] PointRCNN [56], Part-A^2 [57], PV-RCNN [53], CenterPoint [82], and PV-RCNN++ [55]. ***Segmentors:*** SqueezeSeg [69], SqueezeSegV2 [70], RangeNet++ [42], SalsaNext [13], FIDNet [86], CENet [11], PolarNet [85], KPConv [64], PIDS [84], WaffleIron [47], MinkUNet [12], Cylinder3D [90], SPVCNN [63], RPVNet [74], CPGNet [36], 2DPASS [75], and GFNet [48]. We also include three recent 3D augmentation methods, *i.e.*, Mix3D [44], LaserMix [31], and PolarMix [71].

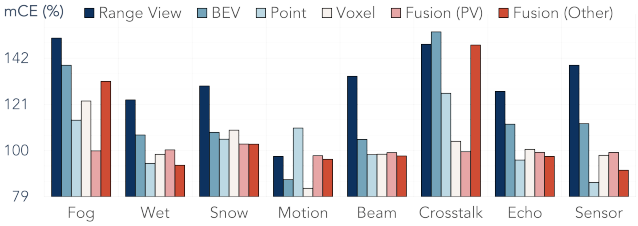


Figure 3: The robustness comparisons among different LiDAR representations (modalities) on *SemanticKITTI-C*.

Evaluation Protocol. Most models benchmarked follow similar data augmentation, pre-training, and validation configurations. We thus directly use public checkpoints for evaluation whenever applicable, or re-train the model following default settings. We notice that some models use extra tricks on original validation sets, *e.g.*, test-time augmentation, model ensemble, *etc*. For such cases, we re-train their models with conventional settings and report the reproduced results. This is to ensure that the robustness comparisons across different models on our corruption sets are fair and convincing. Kindly refer to the Appendix for more details on training and evaluation protocols and to access the

pre-trained model weights for reproduction purposes.

4.2. Benchmark Analysis

In this section, we draw the following key observations based on the benchmarking results and analyze the potential causes behind them.

O-1: 3D Perception Robustness - *existing 3D detectors and segmentors are vulnerable to real-world corruptions*. As shown in Fig. 2, although the models' corruption errors often correlate with the task-specific accuracy (first row), their resilience scores are rather flattened or even descending towards vulnerabilities (second row). The per-corruption errors shown in Tab. 1 to Tab. 6 further verify such crux. Taking 3D segmentors as an example: although the very recent state-of-the-art methods [75, 48, 47] have achieved promising results on the standard benchmark, they are actually less robust than the baseline, *i.e.*, their mCE scores are higher than MinkUNet [12]. A similar trend appears for the 3D detectors, *e.g.* Fig. 2(c), where models with higher NDS are becoming less resilient. Due to the lack of a suitable robustness evaluation benchmark, the existing 3D perception models tend to overfit the "clean" data distributions rather than realistic scenarios.

O-2: Sensor Configurations - *models trained with LiDAR data from different sources exhibit inconsistent sensitivities to each corruption type*. As shown in the third row of Fig. 2, the same corruption applied on different datasets shows diverse behaviors on model's robustness. Different data collection protocols and sensor setups cause a direct impact on model representation learning. For example, 3D detectors trained on 64-beam datasets (KITTI, WOD) are less robust to *motion blur* and *snow*, compared to their counterparts trained on the sparser dataset (nuScenes). We conjecture that the low-density inputs have incorporated certain resilience for models against noises that occur locally but might become fragile for scenarios that lose points in a global manner, *i.e.*, the *cross-sensor* corruption.

O-3: Data Representations - *representing the LiDAR data as raw points, sparse voxels, or the fusion of them tend to yield better robustness*. It can be easily seen from Fig. 3 that the corruption errors of projection-based methods, *i.e.* range view and BEV, are much higher than other modalities, for almost every corruption type in the benchmark. Such disadvantages also hold for fusion-based models that use a 2D branch, *e.g.*, RPVNet [74] and GFNet [48]. In general, the point-based methods [64, 47, 84] are more robust to situations where a significant amount of points are missing while suffering from translation, jittering, and outliers. We conjecture that the sub-sampling and local aggregation widely used in existing point-based architectures are natural rescues for point drops and occlusions. Among all representations, voxel/pillar and point-voxel fusion exhibit a clear superiority under various corruption types, as verified in Tab. 1, Tab. 2, and Tab. 3, respectively. The voxelization processes that quantize the irregular points are conducive to mitigating the local variations and often yield a more steady representation for feature learning.

O-4: Task Particularity - *The 3D detectors and segmentors show different sensitivities to corruption scenarios*. The detection task only targets classification and localization at the object level; corruptions that occur at points in-

Table 1: The **Corruption Error (CE)** of *22 segmentors* on *SemanticKITTI-C*. **Bold**: Best in col. Underline: Second best in col. Dark : Best in row. Red : Worst in row.

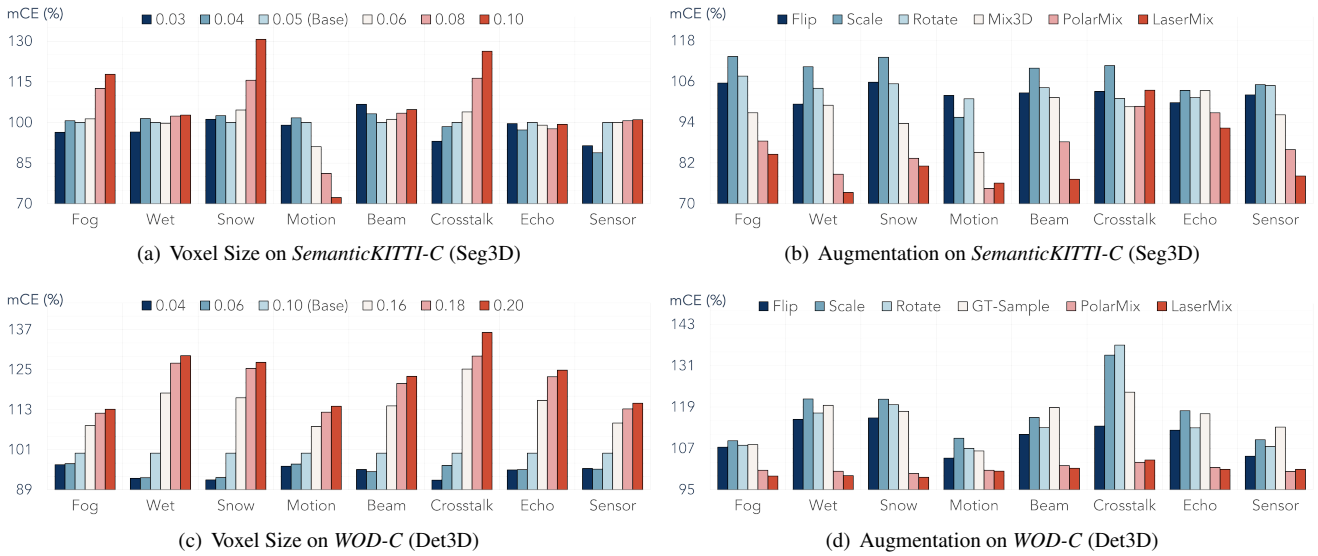
Method	mCE ↓	Fog	Wet	Snow	Move	Beam	Cross	Echo	Sensor
MinkU$_{18}$ [12]	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
SqSeg [69]	164.9	183.9	158.0	165.5	122.4	171.7	188.1	158.7	170.8
SqSegV2 [70]	152.5	168.5	141.2	154.6	115.2	155.2	176.0	145.3	163.5
RGNet$_{21}$ [42]	136.3	156.3	128.5	133.9	102.6	141.6	148.9	128.3	150.6
RGNet$_{53}$ [42]	130.7	144.3	123.7	128.4	104.2	135.5	129.4	125.8	153.9
SalsaNext [13]	116.1	147.5	112.1	116.6	77.6	115.3	143.5	114.0	102.5
FIDNet [86]	113.8	127.7	105.1	107.7	88.9	116.0	121.3	113.7	130.0
CENet [11]	103.4	129.8	92.7	99.2	70.5	101.2	131.1	102.3	100.4
PolarNet [85]	118.6	138.8	107.1	108.3	86.8	105.1	178.1	112.0	112.3
KPConv [64]	99.5	103.2	91.9	98.1	110.7	97.6	111.9	97.3	85.4
PIDS$_{1.2\times}$ [84]	104.1	118.1	98.9	109.5	114.8	103.2	103.9	97.0	87.6
PIDS$_{2.0\times}$ [84]	101.2	110.6	95.7	104.6	115.6	98.6	102.2	97.5	84.8
Waffle [47]	109.5	123.5	90.1	108.5	99.9	93.2	186.1	**91.0**	84.1
MinkU$_{34}$ [12]	100.6	105.3	99.4	106.7	98.7	97.6	99.9	99.0	98.3
Cy3D$_{\text{SPC}}$ [90]	103.3	142.5	92.5	113.6	70.9	97.0	105.7	104.2	99.7
Cy3D$_{\text{TSC}}$ [90]	103.1	142.5	101.3	116.9	61.7	98.9	111.4	99.0	93.4
SPV$_{18}$ [63]	100.3	101.3	100.0	104.0	97.6	99.2	100.6	99.6	100.2
SPV$_{34}$ [63]	**99.2**	**98.5**	100.7	102.0	97.8	99.0	**98.4**	98.8	98.1
RPVNet [74]	111.7	118.7	101.0	104.6	78.6	106.4	185.7	99.2	99.8
CPGNet [36]	107.3	141.0	92.6	104.3	**61.1**	**90.9**	195.6	95.0	**78.2**
2DPASS [75]	106.1	134.9	**85.5**	110.2	62.9	94.4	171.7	96.9	92.7
GFNet [48]	108.7	131.3	94.4	**92.7**	61.7	98.6	198.9	98.2	93.6

Table 2: The **Corruption Error (CE)** of *12 segmentors* on *nuScenes-C (Seg3D)*. **Bold**: Best in col. Underline: Second best in col. Dark : Best in row. Red : Worst in row.

Method	mCE ↓	Fog	Wet	Snow	Move	Beam	Cross	Echo	Sensor
MinkU$_{18}$ [12]	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
FIDNet [86]	122.4	75.9	122.6	68.8	192.0	164.8	58.0	141.7	155.6
CENet [11]	112.8	71.2	115.5	64.3	156.7	159.0	53.3	129.1	153.4
PolarNet [85]	115.1	90.1	115.3	59.0	208.2	121.1	80.7	128.2	118.2
Waffle [47]	106.7	94.7	99.9	84.5	152.4	110.7	91.1	106.4	114.2
MinkU$_{34}$ [12]	96.4	93.0	96.1	104.8	**93.1**	**95.0**	96.3	**96.9**	**95.9**
Cy3D$_{\text{SPC}}$ [90]	111.8	86.6	104.7	70.3	217.5	113.0	75.7	109.2	117.8
Cy3D$_{\text{TSC}}$ [90]	105.6	83.2	111.1	69.7	165.3	114.0	74.4	110.7	116.2
SPV$_{18}$ [63]	106.7	88.4	105.6	98.8	156.5	110.1	86.0	104.3	103.6
SPV$_{34}$ [63]	97.5	95.2	99.5	97.3	95.3	98.7	97.9	**96.9**	98.7
2DPASS [75]	98.6	76.6	**89.1**	76.4	142.7	102.2	89.4	101.8	110.4
GFNet [48]	**92.6**	**65.6**	93.8	**47.2**	152.5	112.9	**45.3**	105.5	117.6

Table 3: The **Corruption Error (CE)** of *5 segmentors* on *WOD-C (Seg3D)*. **Bold**: Best in col. Underline: Second best in col. Dark : Best in row. Red : Worst in row.

Method	mCE ↓	Fog	Wet	Snow	Move	Beam	Cross	Echo	Sensor
MinkU$_{18}$ [12]	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
MinkU$_{34}$ [12]	**96.2**	**96.0**	94.9	99.5	**96.2**	**95.4**	**96.8**	**96.8**	**94.1**
Cy3D$_{\text{TSC}}$ [90]	106.0	111.8	104.1	**98.4**	110.3	105.8	106.9	108.2	102.6
SPV$_{18}$ [63]	103.6	105.6	104.8	99.2	105.4	104.8	99.7	104.3	104.9
SPV$_{34}$ [63]	98.7	99.7	**96.4**	100.4	100.0	98.5	101.9	97.9	95.0



(a) Voxel Size on *SemanticKITTI-C* (Seg3D)

(b) Augmentation on *SemanticKITTI-C* (Seg3D)

(c) Voxel Size on *WOD-C* (Det3D)

(d) Augmentation on *WOD-C* (Det3D)

Figure 4: Corruption sensitivity analysis on *voxel size* (a & c) and *augmentation* (b & d) for the baseline LiDAR semantic segmentation and 3D object detection models [12, 82]. Different corruptions exhibit variances under certain configurations.

Table 4: The **Corruption Error (CE)** of *7 detectors* on *KITTI-C*. **Bold**: Best in col. Underline: Second best in col. Dark : Best in row. Red : Worst in row.

Method	mCE ↓	Fog	Wet	Snow	Move	Beam	Cross	Echo	Sensor
CenterPP [82]	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
SECOND [77]	95.9	99.7	100.6	87.6	97.6	91.5	96.5	99.2	94.8
P-Pillars [32]	110.7	115.8	106.4	124.9	101.6	95.3	117.6	109.9	113.9
P-RCNN [56]	91.9	93.2	90.1	96.8	93.1	86.1	100.9	92.4	**82.5**
PartA^2-F [57]	**82.2**	**89.4**	**75.8**	**81.3**	**86.2**	**80.9**	**71.8**	**83.6**	88.9
PartA^2-A [57]	88.6	92.6	83.2	94.6	86.4	87.0	83.2	89.3	92.7
PVRCNN [53]	90.0	95.2	86.6	93.1	87.5	86.0	87.1	90.0	94.7

Table 5: The **Corruption Error (CE)** of *5 detectors* on *nuScenes-C (Det3D)*. **Bold**: Best in col. Underline: Second best in col. Dark : Best in row. Red : Worst in row.

Method	mCE ↓	Fog	Wet	Snow	Move	Beam	Cross	Echo	Sensor
CenterPP [82]	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
SECOND [77]	97.5	95.4	96.0	96.1	100.8	99.3	92.2	97.6	102.6
P-Pillars [32]	102.9	102.9	104.6	102.5	106.4	102.4	100.9	102.4	**101.1**
CenterLR [82]	98.7	97.9	96.5	97.7	102.2	101.1	95.5	**95.6**	103.5
CenterHR [82]	**95.8**	**93.0**	**92.0**	**94.9**	**97.6**	**98.4**	**91.1**	96.2	103.2

Table 6: The **Corruption Error (CE)** of *5 detectors* on *WOD-C (Det3D)*. **Bold**: Best in col. Underline: Second best in col. Dark : Best in row. Red : Worst in row.

Method	mCE ↓	Fog	Wet	Snow	Move	Beam	Cross	Echo	Sensor
CenterPP [82]	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
SECOND [77]	121.4	117.9	126.5	127.5	113.4	121.3	127.8	123.7	113.5
P-Pillars [32]	127.5	120.8	135.2	129.7	115.2	123.0	151.7	131.6	113.1
PVRCNN [53]	104.9	110.1	104.2	95.7	101.3	110.7	101.8	106.0	109.4
PV++ [55]	**91.6**	**95.7**	**88.3**	**90.1**	**93.2**	**92.5**	**88.9**	**90.8**	**93.2**

side an instance range have less impact on detecting the object. However, the segmentation task is to identify the semantic meaning of each point in the point cloud. Such a task discrepancy is affecting the model's robustness across different corruptions. From Fig. 2 we find that 3D detectors tend to be more robust to point-level variations, such as *motion blur* and *crosstalk*. These two corruptions likely yield noise offsets that are out of the grid size; while these point translations could easily be misclassified by the 3D segmentation models. On the contrary, the 3D segmentors are more steady to environmental changes like *fog*, *wet ground*, and *snow*. From hindsight, we believe that a sophisticated combination of the detection and segmentation tasks would be a viable solution for building more robust and reliable 3D perception systems against different corruptions.

O-5: Augmentation & Regularization Effects - *The recent out-of-context augmentation (OCA) techniques improve 3D robustness by large margins; the flexible rasterization strategies help learn more robust features*. The in-context augmentations (ICAs), *i.e.*, flip, scale, and rotation, are commonly used in 3D detectors and segmentors. Although these techniques help boost perception accuracy, they are less effective in improving robustness. Recent works [44, 31, 71] proposed OCAs with the goal of further enhancing model performance on the "clean" sets. We implement these augmentations on baseline models and test their effectiveness on our robustness evaluation sets, as shown in Fig. 4 (b) & (d). Since corrupted data often deviate from the training distribution, the model will inevitably degrade under OoD scenarios. OCAs that mix and swap regions without maintaining the consistency of scene layouts are yielding much lower CE scores across all corruptions,

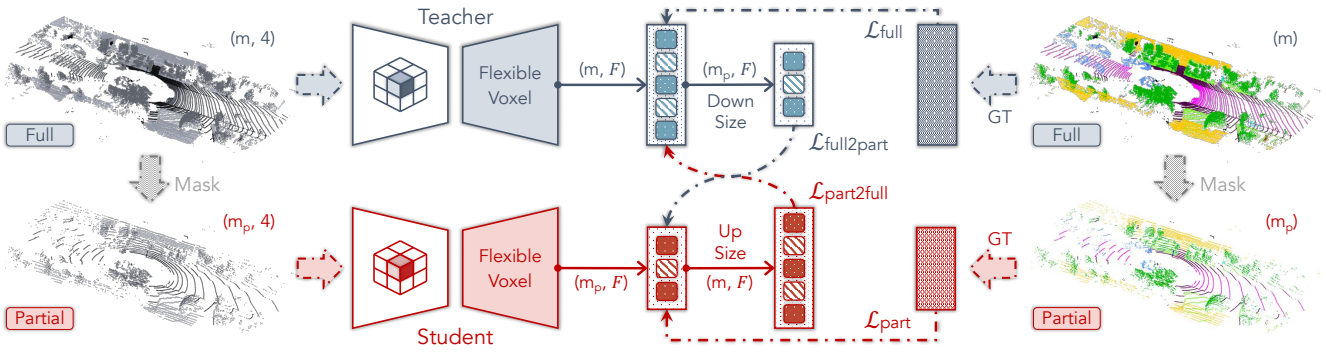


Figure 5: The proposed density-insensitive training framework. The "full" and "partial" point clouds are fed into the teacher branch and student branch, respectively, for feature learning, while the latter is generated by randomly masking the original point cloud. To encourage cross-density consistency, we calculate the *completion* and *confirmation* losses which measure the distances of sub-sampled teacher's prediction and interpolated student's prediction between the other branch's outputs.

except for *wet ground*, where the loss of ground points restricts the effectiveness of scene mixing. Another key factor that influences the robustness (especially for voxel- and point-voxel fusion-based methods) is representation capacity, *i.e.*, voxel size. As shown in Fig. 4 (a) & (c), the 3D segmentors under translations within small regions (*motion blur*) favor a larger voxel size to suppress the global translations; conversely, they are more robust against outliers (*fog*, *snow*, and *crosstalk*) given more fine-grained voxelizations to eliminate the local variations. For 3D detectors, a consensus is formed toward using a higher voxelization resolution, and improvements are constantly achieved across all corruption types in the benchmark.

5. Boosting Corruption Robustness

Motivated by the above observations, we propose two novel techniques to enhance the robustness against corruptions. We conduct experiments on the *SemanticKITTI-C* dataset without loss of generality and include more results on other datasets in the Appendix.

Flexible Voxelization. The widely used sparse convolution [62] requires the formal transformation of the point coordinates $\mathbf{p}_k = (p_k^x, p_k^y, p_k^z)$ into a sparse voxel. This process is often formulated as follows:

$$\mathbf{v}_k = (v_k^x, v_k^y, v_k^z) = \texttt{floor}((\frac{p_k^x}{l^x}), (\frac{p_k^y}{l^y}), (\frac{p_k^z}{l^z})) \,, \quad (12)$$

where l^x, l^y, and l^z denote the voxel size along each axis and are often set as fixed values. As discussed in Fig. 4 (a) & (c), the model tends to show an erratic resilience under different corruptions, *e.g.*, favor a larger voxel size for *motion blur* while is more robust against *fog*, *snow*, and *crosstalk* with a smaller voxel size. To pursue better generalizability among all corruptions, we switch the naive constant into a dynamic alternative $l_{\text{dv}} = (l^x \pm \text{dv}^x, l^y \pm \text{dv}^y, l^z \pm \text{dv}^z)$,

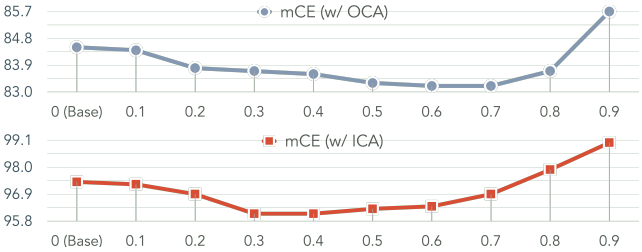


Figure 6: Ablation study on the masking ratio β for models: **[top]** trained *w/* OCA and **[bottom]** trained *w/* ICA.

where dv^x, dv^y, dv^z are the offsets sampled from the continuous uniform distribution with an interval γ.

Density-Insensitive Training. The natural corruptions often cause severe occlusion, attenuation, and reflection of light impulses, resulting in the unavoidable loss of LiDAR points in certain regions around the ego-vehicle [59]. For example, the *wet ground* absorbs energy and loses points on the surfaces [20]; the potential *incomplete echo* and *beam missing* caused by reflection or dust and insects occlusion may lead to serious object failure [83]. The 3D perception models that suffer from such OoD scenarios bear the risk of being involved in safety-critical issues. It is worth noting that such degradation is not compensable via either adjusting the voxel size or applying OCA (see Fig. 4). Inspired by recent masking-based representation methods [22, 14, 43, 24], we propose a robust finetuning framework (see Fig. 5) that tends to be less sensitive to density variations. Specifically, we design a two-branch structure – a teacher net $\mathcal{G}_\theta^{\text{tea}}$ and a student net $\mathcal{G}_\theta^{\text{stu}}$ – that takes a pair of high- and low-density point clouds (x and $\tilde{x}$) as the input, where the sparser one is generated by randomly masking the points from the original point cloud with a ratio β. Note that here we use the random mask to sub-sample

the given point clouds rather than simulating a specific corruption type defined in our benchmark, since the corruption "pattern" in the actual scenario is often hard to predict. The loss functions of the k-th sample from the "full" view and the "partial" view are calculated as follows:

$$\mathcal{L}_{\text{full}} = \mathcal{L}_{\text{task}}(y_k, \mathcal{G}_\theta^{\text{tea}}(x_k)) \ , \ \ \mathcal{L}_{\text{part}} = \mathcal{L}_{\text{task}}(\tilde{y}_k, \mathcal{G}_\theta^{\text{stu}}(\tilde{x}_k)), \tag{13}$$

where y_k and $\tilde{y}_k$ are original and masked ground-truths, respectively. $\mathcal{L}_{\text{task}}$ denotes the task-specific loss, *e.g.*, RPN loss for detection and cross-entropy loss for segmentation.

To encourage cross-consistency between the high- and low-density branches, we calculate $\mathcal{L}_{\text{part2full}}$ and $\mathcal{L}_{\text{full2part}}$, where the former is to mimic dense representations from sparse inputs (completion) and the latter is to pursue local agreements (confirmation). The completion loss is calculated as the distance between the teacher net's prediction of the "full" input and the interpolated student net's prediction of the "partial" input, which can be calculated as follows:

$$\mathcal{L}_{\text{part2full}} = || \ \mathcal{G}_\theta^{\text{tea}}(x), \ \texttt{interp}(\mathcal{G}_\theta^{\text{stu}}(\tilde{x})) \ ||_2^2 \ . \tag{14}$$

Similarly, the confirmation loss for pursuing local agreements can be calculated as follows:

$$\mathcal{L}_{\text{full2part}} = || \ \texttt{subsample}(\mathcal{G}_\theta^{\text{tea}}(x)), \ \mathcal{G}_\theta^{\text{stu}}(\tilde{x}) \ ||_2^2 \ . \tag{15}$$

The final objective is to optimize the summation of the above loss functions, *i.e.*, $\mathcal{L} = \mathcal{L}_{\text{full}} + \mathcal{L}_{\text{part}} + \alpha_1 \mathcal{L}_{\text{part2full}} + \alpha_2 \mathcal{L}_{\text{full2part}}$, where α_1 and α_2 are the weight coefficients.

Implementation Details. We ablate each component and show the results in Tab. 7. Specifically, γ is set as 0.02 in our experiments, along with a mask ratio $\beta = 0.4$ for models *w/* ICA and $\beta = 0.6$ for models *w/* OCA. We initialize both teacher and student networks with the same baseline model and finetune our framework for 6 epochs in total. The weight coefficients are set as 50 and 100, respectively.

Experimental Analysis. Despite its simplicity, we found this framework is conducive to mitigating robustness degradation from corruptions. The simple modification on voxel partition can boost the corruption robustness by large margins; it reduces 2.6% mCE and 1.5% mCE upon the two baselines, respectively. Then, we incorporate the cross-consistency learning between "full" and "partial" views. Among all variants, the one with both completion ($\mathcal{L}_{\text{part2full}}$) and confirmation ($\mathcal{L}_{\text{full2part}}$) objectives achieves the best possible results in terms of mCE and mRR.

We also show an ablation study of the masking ratio β in Fig. 6. We observe that there is often a trade-off between the model's robustness and the proportion of information occlusion; a ratio between 0.3 to 0.6 tends to yield lower mCE (better robustness). It is worth noting that both flexible voxelization and density-insensitive training will slightly lower the task-specific accuracy on the "clean" sets, as shown in the last column of Tab. 7. We conjecture that such an out-of-context consistency regularization will likely relieve the 3D perception model from overfitting the training distribution and in return, become more robust against unseen scenarios from the OoD distribution.

Table 7: Ablation study on: **[left]** in-context (ICA) and out-of-context (OCA) augmentations; **[middle]** voxelization strategies; and **[right]** density-insensitive training losses.

Method	ICA	OCA	Size	$\mathcal{L}_{\text{part2full}}$	$\mathcal{L}_{\text{full2part}}$	mCE ↓	mRR ↑	Clean
Base [12]	✓		Fixed			100.0	81.9	62.8
Ours - (1)	✓		Flexible			97.4	84.2	**62.9**
Ours - (2)	✓		Flexible	✓		96.4	85.1	62.7
Ours - (3)	✓		Flexible	✓	✓	**96.1**	**85.6**	62.7
Base [12]		✓	Fixed			86.0	84.7	**69.2**
Ours - (4)		✓	Flexible			84.5	86.8	68.2
Ours - (5)		✓	Flexible	✓		83.8	88.1	67.9
Ours - (6)		✓	Flexible	✓	✓	**83.2**	**89.7**	68.1

6. Discussion and Conclusion

In this work, we established a comprehensive evaluation benchmark dubbed *Robo3D* for probing and analyzing the robustness of LiDAR-based 3D perception models. We defined eight distinct corruption types with three severity levels on four large-scale autonomous driving datasets. We systematically benchmarked and analyzed representative 3D detectors and segmentors to understand their resilience under real-world corruptions and sensor failure. Several key insights are drawn from aspects including sensor setups, data representations, task particularity, and augmentation effects. To pursue better robustness, we proposed a cross-density consistency training framework and a simple yet effective flexible voxelization strategy. We hope this work could lay a solid foundation for future research on building more robust and reliable 3D perception models.

Potential Limitation. Although we benchmarked a wide range of corruptions that occur in the real world, we do not consider cases that are coupled with multiple corruptions at the same time. Besides, we do not include models that take multi-modal inputs, which could form future directions.

Acknowledgements. This research is part of the programme DesCartes and is supported by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. This study is supported by the Ministry of Education, Singapore, under its MOE AcRF Tier 2 (MOE-T2EP20221-0012), NTU NAP, and under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). This study is also supported by the National Key R&D Program of China (No. 2022ZD0161600).

References

[1] Fatima Albreiki, Sultan Abughazal, Jean Lahoud, Rao Anwer, Hisham Cholakkal, and Fahad Khan. On the robustness of 3d object detectors. *arXiv preprint arXiv:2207.10205*, 2022.

[2] Antonio Alliegro, Francesco Cappio Borlino, and Tatiana Tommasi. Towards open set 3d learning: A benchmark on object point clouds. *arXiv preprint arXiv:2207.11554*, 2022.

[3] Eduardo Arnold, Omar Y Al-Jarrah, Mehrdad Dianati, Saber Fallah, David Oxtoby, and Alex Mouzakitis. A survey on 3d object detection methods for autonomous driving applications. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3782–3795, 2019.

[4] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Juergen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *IEEE/CVF International Conference on Computer Vision*, pages 9297–9307, 2019.

[5] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11682–11692, 2020.

[6] Mario Bijelic, Tobias Gruber, and Werner Ritter. A benchmark for lidar sensors in fog: Is detection breaking down? In *IEEE Intelligent Vehicles Symposium*, pages 760–767, 2018.

[7] Lara Brinon-Arranz, Tiana Rakotovao, Thierry Creuzet, Cem Karaoguz, and Oussama El-Hamzaoui. A methodology for analyzing the impact of crosstalk on lidar measurements. *IEEE Sensors*, 8:1–4, 2021.

[8] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11621–11631, 2020.

[9] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. Invisible for both camera and lidar: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *IEEE Symposium on Security and Privacy*, pages 176–194, 2021.

[10] Prithvijit Chattopadhyay, Judy Hoffman, Roozbeh Mottaghi, and Aniruddha Kembhavi. Robustnav: Towards benchmarking robustness in embodied navigation. In *IEEE/CVF International Conference on Computer Vision*, pages 15691–15700, 2021.

[11] Huixian Cheng, Xianfeng Han, and Guoqiang Xiao. Cenet: Toward concise and efficient lidar semantic segmentation for autonomous driving. In *IEEE International Conference on Multimedia and Expo*, 2022.

[12] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019.

[13] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds. In *International Symposium on Visual Computing*, pages 207–222, 2020.

[14] Xiaoyi Dong, Yinglin Zheng, Jianmin Bao, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. Maskclip: Masked self-distillation advances contrastive language-image pretraining. *arXiv preprint arXiv:2208.12262*, 2022.

[15] Yinpeng Dong, Caixin Kang, Jinlai Zhang, Zijian Zhu, Yikai Wang, Xiao Yang, Hang Su, Xingxing Wei, and Jun Zhu. Benchmarking robustness of 3d object detection to common corruption. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1022–1032, 2023.

[16] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[17] Whye Kit Fong, Rohit Mohan, Juana Valeria Hurtado, Lubing Zhou, Holger Caesar, Oscar Beijbom, and Abhinav Valada. Panoptic nuscenes: A large-scale benchmark for lidar panoptic segmentation and tracking. *IEEE Robotics and Automation Letters*, 7:3795–3802, 2022.

[18] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[19] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12):4338–4364, 2020.

[20] Martin Hahner, Christos Sakaridis, Mario Bijelic, Felix Heide, Fisher Yu, Dengxin Dai, and Luc Van Gool. Lidar snowfall simulation for robust 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16364–16374, 2022.

[21] Martin Hahner, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Fog simulation on real lidar point clouds for 3d object detection in adverse weather. In *IEEE/CVF International Conference on Computer Vision*, pages 15283–15292, 2021.

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2021.

[23] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

[24] Georg Hess, Johan Jaxing, Elias Svensson, David Hagerman, Christoffer Petersson, and Lennart Svensson. Masked autoencoders for self-supervised learning on automotive point clouds. *arXiv preprint arXiv:2207.00531*, 2022.

[25] Keli Huang, Botian Shi, Xiang Li, Xin Li, Siyuan Huang, and Yikang Li. Multi-modal sensor fusion for auto driving perception: A survey. *arXiv preprint arXiv:2202.02703*, 2022.

[26] Maximilian Jaritz, Tuan-Hung Vu, Raoul de Charette, Emilie Wirbel, and Patrick Pérez. xmuda: Cross-modal unsupervised domain adaptation for 3d semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12605–12614, 2020.

[27] Christoph Kamann and Carsten Rother. Benchmarking the robustness of semantic segmentation models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8828–8838, 2020.

[28] Oğuzhan Fatih Kar, Teresa Yeo, Andrei Atanov, and Amir Zamir. 3d common corruptions and data augmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8963–18974, 2022.

[29] Lingdong Kong, Youquan Liu, Runnan Chen, Yuexin Ma, Xinge Zhu, Yikang Li, Yuenan Hou, Yu Qiao, and Ziwei Liu. Rethinking range view representation for lidar segmentation. *arXiv preprint arXiv:2303.05367*, 2023.

[30] Lingdong Kong, Niamul Quader, and Venice Erin Liong. Conda: Unsupervised domain adaptation for lidar segmentation via regularized domain concatenation. In *IEEE International Conference on Robotics and Automation*, pages 9338–9345, 2023.

[31] Lingdong Kong, Jiawei Ren, Liang Pan, and Ziwei Liu. Lasermix for semi-supervised lidar semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21705–21715, 2023.

[32] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019.

[33] Shuangzhi Li, Zhijie Wang, Felix Juefei-Xu, Qing Guo, Xingyu Li, and Lei Ma. Common corruption robustness of point cloud detectors: Benchmark and enhancement. *arXiv preprint arXiv:2210.05896*, 2021.

[34] Xin Li, Tao Ma, Yuenan Hou, Botian Shi, Yucheng Yang, Youquan Liu, Xingjiao Wu, Qin Chen, Yikang Li, Yu Qiao, and Liang He. Logonet: Towards accurate 3d object detection with local-to-global cross-modal fusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17524–17534, 2023.

[35] Xin Li, Botian Shi, Yuenan Hou, Xingjiao Wu, Tianlong Ma, Yikang Li, and Liang He. Homogeneous multi-modal feature fusion and interaction for 3d object detection. In *European Conference on Computer Vision*, pages 691–707, 2022.

[36] Xiaoyan Li, Gang Zhang, Hongyu Pan, and Zhenhua Wang. Cpgnet: Cascade point-grid fusion network for real-time lidar semantic segmentation. In *IEEE International Conference on Robotics and Automation*, pages 11117–11123, 2022.

[37] Zhichao Li, Feng Wang, and Naiyan Wang. Lidar r-cnn: An efficient and universal 3d object detector. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7546–7555, 2021.

[38] Venice Erin Liong, Thi Ngoc Tho Nguyen, Sergi Widjaja, Dhananjai Sharma, and Zhuang Jie Chong. Amvnet: Assertion-based multi-view fusion network for lidar semantic segmentation. *arXiv preprint arXiv:2012.04934*, 2020.

[39] Tao Ma, Xuemeng Yang, Hongbin Zhou, Xin Li, Botian Shi, Junjie Liu, Yuchen Yang, Zhizheng Liu, Liang He, Yu Qiao, et al. Detzero: Rethinking offboard 3d object detection with long-term sequential point clouds. *arXiv preprint arXiv:2306.06023*, 2023.

[40] Jiageng Mao, Yujing Xue, Minzhe Niu, Haoyue Bai, Jiashi Feng, Xiaodan Liang, Hang Xu, and Chunjing Xu. Voxel transformer for 3d object detection. In *IEEE/CVF International Conference on Computer Vision*, pages 3164–3173, 2021.

[41] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S. Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019.

[42] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4213–4220, 2019.

[43] Chen Min, Dawei Zhao, Liang Xiao, Yiming Nie, and Bin Dai. Voxel-mae: Masked autoencoders for pre-training large-scale point clouds. *arXiv preprint arXiv:2206.09900*, 2022.

[44] Alexey Nekrasov, Jonas Schult, Or Litany, Bastian Leibe, and Francis Engelmann. Mix3d: Out-of-context data augmentation for 3d scenes. In *International Conference on 3D Vision*, pages 116–125, 2021.

[45] Won Park, Nan Liu, Qi Alfred Chen, and Z. Morley Mao. Sensor adversarial traits: Analyzing robustness of 3d object detection sensor fusion models. In *IEEE International Conference on Image Processing*, pages 484–488, 2019.

[46] Tyson Govan Phillips, Nicky Guenther, and Peter Ross McAree. When the dust settles: The four behaviors of lidar in the presence of fine airborne particulates. *Journal of Field Robotics*, 34:985–1009, 2017.

[47] Gilles Puy, Alexandre Boulch, and Renaud Marlet. Using a waffle iron for automotive point cloud semantic segmentation. *arXiv preprint arXiv:2301.10100*, 2023.

[48] Haibo Qiu, Baosheng Yu, and Dacheng Tao. Gfnet: Geometric flow network for 3d point cloud semantic segmentation. *Transactions on Machine Learning Research*, 2022.

[49] Jiawei Ren, Liang Pan, and Ziwei Liu. Benchmarking and analyzing point cloud classification under corruptions. In *International Conference on Machine Learning*, pages 18559–18575, 2022.

[50] Giulio Rossolini, Federico Nesti, Gianluca D'Amico, Saasha Nair, Alessandro Biondi, and Giorgio Buttazzo. On the real-world adversarial robustness of real-time semantic segmentation models for autonomous driving. *arXiv preprint arXiv:2201.01850*, 2022.

[51] Alvari Seppänen, Risto Ojala, and Kari Tammi. Adverse weather denoising from adjacent point clouds. *IEEE Robotics and Automation Letters*, 8:456–463, 2022.

[52] Guangsheng Shi, Ruifeng Li, and Chao Ma. Pillarnet: High-performance pillar-based 3d object detection. *arXiv preprint arXiv:2205.07403*, 2022.

[53] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn:

Point-voxel feature set abstraction for 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10529–10538, 2020.

[54] Shaoshuai Shi, Li Jiang, Jiajun Deng, Zhe Wang, Chaoxu Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. *International Journal of Computer Vision*, pages 1–21, 2022.

[55] Shaoshuai Shi, Li Jiang, Jiajun Deng, Zhe Wang, Chaoxu Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. *International Journal of Computer Vision*, 2022.

[56] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–779, 2019.

[57] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(8):2647–2664, 2020.

[58] Weijing Shi and Raj Rajkumar. Point-gnn: Graph neural network for 3d object detection in a point cloud. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1711–1719, 2020.

[59] Jungil Shin, Hyunsuk Park, and Taejung Kim. Characteristics of laser backscattering intensity to detect frozen and wet surfaces on roads. *Journal of Sensors*, 2019.

[60] Jiachen Sun, Qingzhao Zhang, Bhavya Kailkhura, Zhiding Yu, Chaowei Xiao, and Z. Morley Mao. Benchmarking robustness of 3d point cloud recognition against common corruptions. *arXiv preprint arXiv:2201.12296*, 2022.

[61] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, and Benjamin Caine. Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020.

[62] Haotian Tang, Zhijian Liu, Xiuyu Li, Yujun Lin, and Song Han. Torchsparse: Efficient point cloud inference engine. In *Conference on Machine Learning and Systems*, 2022.

[63] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *European Conference on Computer Vision*, pages 685–702, 2020.

[64] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *IEEE/CVF International Conference on Computer Vision*, pages 6411–6420, 2019.

[65] James Tu, Mengye Ren, Sivabalan Manivasagam, Ming Liang, Bin Yang, Richard Du, Frank Cheng, and Raquel Urtasun. Physically realizable adversarial examples for lidar object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13716–13725, 2020.

[66] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[67] Jiahang Wang, Sheng Jin, Wentao Liu, Weizhong Liu, Chen Qian, and Ping Luo. When human pose estimation meets robustness: Adversarial algorithms and benchmarks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11855–11864, 2021.

[68] Yi Wei, Zibu Wei, Yongming Rao, Jiaxin Li, Jie Zhou, and Jiwen Lu. Lidar distillation: bridging the beam-induced domain gap for 3d object detection. In *European Conference on Computer Vision*, pages 179–195, 2022.

[69] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In *IEEE International Conference on Robotics and Automation*, pages 1887–1893, 2018.

[70] Bichen Wu, Xuanyu Zhou, Sicheng Zhao, Xiangyu Yue, and Kurt Keutzer. Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In *IEEE International Conference on Robotics and Automation*, pages 4376–4382, 2019.

[71] Aoran Xiao, Jiaxing Huang, Dayan Guan, Kaiwen Cui, Shijian Lu, and Ling Shao. Polarmix: A general data augmentation technique for lidar point clouds. In *Advances in Neural Information Processing Systems*, 2022.

[72] Aoran Xiao, Jiaxing Huang, Dayan Guan, Fangneng Zhan, and Shijian Lu. Transfer learning from synthetic to real lidar point cloud for semantic segmentation. In *AAAI Conference on Artificial Intelligence*, pages 2795–2803, 2022.

[73] Shaoyuan Xie, Zichao Li, Zeyu Wang, and Cihang Xie. On the adversarial robustness of camera-based 3d object detection. *arXiv preprint arXiv:2301.10766*, 2023.

[74] Jianyun Xu, Ruixiang Zhang, Jian Dou, Yushi Zhu, Jie Sun, and Shiliang Pu. Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In *IEEE/CVF International Conference on Computer Vision*, pages 16024–16033, 2021.

[75] Xu Yan, Jiantao Gao, Chaoda Zheng, Chao Zheng, Ruimao Zhang, Shuguang Cui, and Zhen Li. 2dpass: 2d priors assisted semantic segmentation on lidar point clouds. In *European Conference on Computer Vision*, 2022.

[76] Xu Yan, Chaoda Zheng, Zhen Li, Shuguang Cui, and Dengxin Dai. Benchmarking the robustness of lidar semantic segmentation models. *arXiv preprint arXiv:2301.00970*, 2023.

[77] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

[78] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11040–11048, 2020.

[79] Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Std: Sparse-to-dense 3d object detector for point cloud. In *IEEE/CVF International Conference on Computer Vision*, pages 1951–1960, 2019.

[80] Maosheng Ye, Rui Wan, Shuangjie Xu, Tongyi Cao, and Qifeng Chen. Efficient point cloud segmentation with

geometry-aware sparse networks. In *European Conference on Computer Vision*, pages 196–212, 2022.

[81] Chenyu Yi, Siyuan Yang, Haoliang Li, Yap peng Tan, and Alex Kot. Benchmarking the robustness of spatial-temporal models against corruptions. In *Advances in Neural Information Processing Systems*, 2021.

[82] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11784–11793, 2021.

[83] Kaicheng Yu, Tang Tao, Hongwei Xie, Zhiwei Lin, Zhongwei Wu, Zhongyu Xia, Tingting Liang, Haiyang Sun, Jiong Deng, Dayang Hao, Yongtao Wang, Xiaodan Liang, and Bing Wang. Benchmarking the robustness of lidar-camera fusion for 3d object detection. *arXiv preprint arXiv:2205.14951*, 2022.

[84] Tunhou Zhang, Mingyuan Ma, Feng Yan, Hai Li, and Yiran Chen. Pids: Joint point interaction-dimension search for 3d point cloud. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1298–1307, 2023.

[85] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. *arXiv preprint arXiv:2003.14032*, 2020.

[86] Yiming Zhao, Lin Bai, and Xinming Huang. Fidnet: Lidar point cloud semantic segmentation with fully interpolation decoding. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4453–4458. IEEE, 2021.

[87] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018.

[88] Zixiang Zhou, Xiangchen Zhao, Yu Wang, Panqu Wang, and Hassan Foroosh. Centerformer: Center-based transformer for 3d object detection. In *European Conference on Computer Vision*, pages 496–513, 2022.

[89] Lifa Zhu, Changwei Lin, Cheng Zheng, and Ninghua Yang. Point-voxel adaptive feature abstraction for robust point cloud classification. *arXiv preprint arXiv:2210.15514*, 2022.

[90] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9939–9948, 2021.