PointCRT: Detecting Backdoor in 3D Point Cloud via Corruption Robustness

Shengshan Hu[*][†][‡][§]
hushengshan@hust.edu.cn
School of Cyber Science and Engineering, Huazhong University of Science and Technology

Wei Liu[*][†][‡][§]
weiliu73@hust.edu.cn
School of Cyber Science and Engineering, Huazhong University of Science and Technology

Minghui Li
minghuili@hust.edu.cn
School of Software Engineering, Huazhong University of Science and Technology

Yechao Zhang[*][†][‡][§]
ycz@hust.edu.cn
School of Cyber Science and Engineering, Huazhong University of Science and Technology

Xiaogeng Liu[*][†][‡][§]
liuxiaogeng@hust.edu.cn
School of Cyber Science and Engineering, Huazhong University of Science and Technology

Xianlong Wang[*][†][‡][§]
wxl99@hust.edu.cn
School of Cyber Science and Engineering, Huazhong University of Science and Technology

Leo Yu Zhang
leo.zhang@deakin.edu.au
School of Information Technology, Deakin University

Junhui Hou
jh.hou@cityu.edu.hk
Department of Computer Science, City University of Hong Kong

ABSTRACT

Backdoor attacks for point clouds have elicited mounting interest with the proliferation of deep learning. The point cloud classifiers can be vulnerable to malicious actors who seek to manipulate or fool the model with specific backdoor triggers. Detecting and rejecting backdoor samples during the inference stage can effectively alleviate backdoor attacks. Recently, some black-box test-time backdoor sample detection methods have been proposed in the 2D image domain, without any underlying assumptions about the backdoor triggers. However, upon examination, we have found that these detection techniques are not effective for 3D point clouds. As a result, there is a pressing need to bridge the gap for the development of a universal approach that is specifically designed for 3D point clouds.

In this paper, we propose the first test-time backdoor sample detection method in 3D point cloud without assumption to the backdoor triggers, called **Point** Clouds **C**orruption **R**obustness **T**est (**PointCRT**). Based on the fact that the corruption robustness of clean samples remains relatively stable across various backdoor models, we propose the corruption robustness score to map the features into high-dimensional space. The corruption robustness score is a vector evaluated by label consistency, whose element is the minimum severity level of corruption that changes the label prediction of the victim model. Then, the trigger is identified by detecting the abnormal corruption robustness score through a nonlinear classification. The comprehensive experiments demonstrate PointCRT deals with all cases with the average *AUC* over 0.934 and F1 score over 0.864, with the enhancement of 18%-28% on ModelNet40. Our codes are available at: https://github.com/CGCL-codes/PointCRT.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; **Computer vision**.

KEYWORDS

Deep Learning, Backdoor Detection, 3D Point Clouds

ACM Reference Format:
Shengshan Hu, Wei Liu, Minghui Li, Yechao Zhang, Xiaogeng Liu, Xianlong Wang, Leo Yu Zhang, and Junhui Hou. 2023. PointCRT: Detecting Backdoor in 3D Point Cloud via Corruption Robustness. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3581783.3612456

1 INTRODUCTION

With the stellar progress of deep learning, 3D point clouds arise in a wealth of applications, like autonomous driving, augmented reality, robotics, etc [16]. As a result, the development of robust and efficient deep learning algorithms for 3D point clouds has become a crucial research area. Especially, adversarial machine learning has achieved remarkable advancements spurring an arms race between attacks and defenses [15, 20, 48].

Meanwhile, Badnets [10] leads to the realization that the backdoor attacks (or trojan attacks) become another non-negligible

[*] National Engineering Research Center for Big Data Technology and System
[†] Services Computing Technology and System Lab
[‡] Hubei Key Laboratory of Distributed System Security
[§] Hubei Engineering Research Center on Big Data Security

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3612456

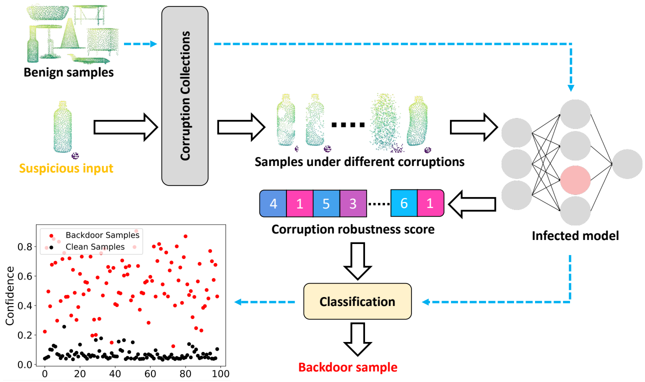


Figure 1: An overview of PointCRT

security threat. Backdoor attacks are intended to deceive the victim model for those samples that contain the engineered trigger but do not have any impact on the model's efficacy with respect to benign data. Hence, it poses a formidable obstacle for users to discern the surreptitious insertion of a backdoor trigger within the model [4, 12, 18, 42]. Numerous defense countermeasures against the backdoor attacks have been proffered in the image domain [2, 4, 7, 39, 41]. From the various defenses against backdoor attacks, one of the pivotal ideas is to detect the injected trigger, as the trigger provided a *shortcut* that builds up a mapping between the trigger and target label [8]. Apparently, detecting a trigger sample at the inference stage is equivalent to removing the hazard.

Recently, the significant advances achieved by test-time backdoor detection methods have garnered widespread attention. These methods aim to detect backdoor samples at test time (inference stage) of the victim models and filter out the potential malicious samples [3, 7]. In real scenarios, the defender has no prior knowledge of the trigger designs or victim model structure including the logits outputs, and no access to the training process of the victim model, also known as the black-box setting. Lately, SCALE-UP [13] and TeCo [29] are proposed to detect backdoor samples in this setting and only require the hard-label outputs. Unfortunately, these 2D image detection methods are encountering significant hurdles in 3D point clouds, the backdoor attack on 3D deep learning is nascent but extremely intractable. One of the major problems comes from the flexible representation of point clouds making the backdoor pattern totally different from images [21]. The pixel representation cannot be considered equal to the coordinates of a point, let alone the image patch. Another problem is the point cloud classifiers as they use symmetric functions to process the unordered point clouds, leading to different designs of backdoor triggers [9, 22, 33]. For example, some detection methods fail when meeting transformation-based triggers like rotation [6, 21], which will be further discussed in Sec. 5.2.

In this paper, we attempt to design a universal black-box backdoor sample detection method tailored for 3D point clouds without any prior knowledge or assumption of the triggers and victim models. In such a strong strict setting, we are only able to obtain a small quantity of clean data that is the same distribution as the test dataset, which will be used to distinguish whether the suspicious input is stamped with a backdoor trigger, shown in Fig. 1. Under the circumstances of restricted access, we opt to pre-process the input samples, with the aim of highlighting the characteristic attributes of the triggers. The resistance of point cloud backdoor attacks under different pre-processing has been discussed in prior work [6]. However, it has only delved into basic pre-processing techniques and no further consideration on detecting backdoor samples. On the other hand, corruption is a more comprehensive method that can be utilized for measuring robustness [35, 38]. The existing point cloud corruption benchmarks have analyzed the point cloud classifiers' robustness under various corruptions. Inspired by [11, 23, 29], we observe that the backdoored infected models have the consistency of corruption robustness on clean data while performing differently on the backdoor data with different triggers. It should be further pointed out that this phenomenon is not entirely consistent with the observation in 2D images as the spacial backdoor pattern and data representation in 3D point clouds [26]. Typically, the robustness of backdoor samples is considered to be more robust than benign samples in 2D images, while this relationship is much more complex in 3D point clouds. The interaction-based triggers and transformation-based triggers have negative and positive effects on enhancing the robustness of the original samples, respectively. Therefore, it cannot simply be divided by a linear separation, as discussed in Sec. 4.2.

Based on that, we propose **Point** Clouds **C**orruption **R**obustness **T**est (**PointCRT**) by applying several corruptions to point clouds with growing severity and obtain the *corruption robustness score* (CRS) via the hard-label prediction consistency. The CRS represents resistance to the maximum severity level under different corruptions, with the maintenance of the model prediction. Based on the observation that the clean samples have stable CRS on different backdoored models, we can determine the backdoor samples by a (curved) hyperplane. Without consideration of the geometry, evaluating the corruption robustness of backdoor samples has great advantages for transformation-based triggers that don't change the structure of the input. We compare PointCRT with the state-of-the-art (SOTA) test-time detection methods proposed for 2D images. The experiments demonstrate it is impractical to directly apply these detection methods from 2D images to 3D point clouds, while PointCRT can achieve remarkable results. We also validate our method against data augmentation during backdoor training and evaluate the transferability under unseen backdoor attacks.

In a nutshell, we make the following contributions:

- We propose PointCRT, the first test-time black-box backdoor sample detection method for 3D point clouds, which can detect the backdoor trigger without any assumption of the trigger or requirement of the victim model.
- We first observe the discrepancy influence on the robustness of backdoor samples between interaction-based triggers and transformation-based triggers.
- Extensive experiments on multiple 3D point clouds benchmark datasets delineate that our approach achieves superior performance in detecting backdoor attacks.
- We find that using transformation-based backdoor samples for PointCRT's training has good transferability and effectiveness in detecting other unseen backdoor attacks.

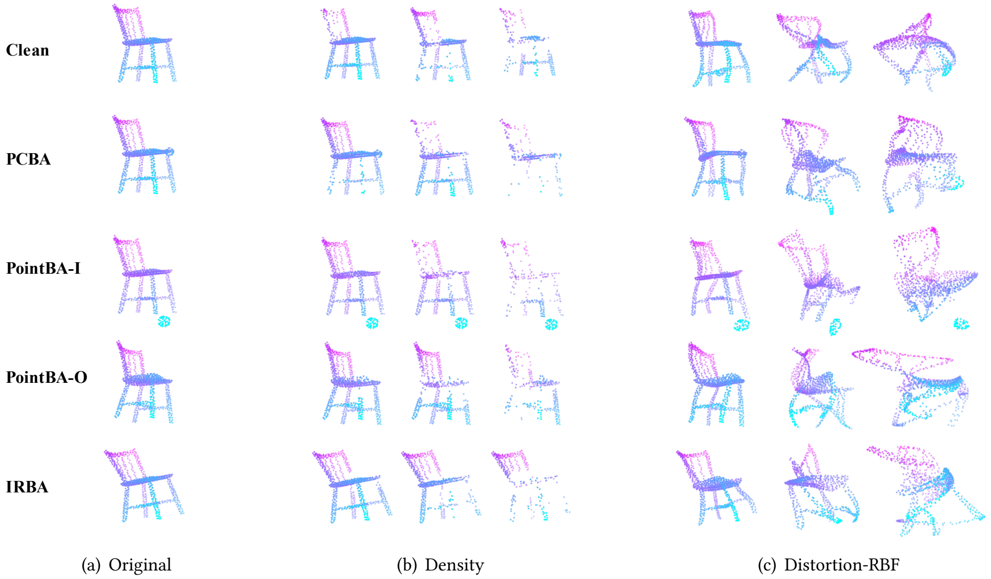


Figure 2: An illustration of one clean sample (first row) and four backdoor samples under corruptions, *i.e.*, radial basis function (RBF)-based distortion and density with the severity levels of 1,3,5. Although the backdoor samples implanted by PointBA-O and IRBA have similar structures to the clean one, the geometry is still changed slightly or dramatically after corruption.

2 RELATED WORK

2.1 Adversarial Attacks and Defenses on 3D Point Cloud

The first adversarial attack algorithm in 3D point clouds is proposed in [48], which have inspired several adversarial attack methods such as point shifting [24–26, 30, 45], point adding [51] and point dropping [53, 54]. The rotation-based attack reveals the vulnerability of the point cloud classifiers to isometry transformations [43, 52]. AdvPC [15] creates transferable adversarial examples which can exploit the data distribution. Si-Adv [19], as the first query-based black box attack in point clouds, limits displacement of the point on its tangent plane. PointCA [17] is the first adversarial attack against 3D point cloud completion, which indicates the potential harm of adversarial samples in other tasks.

To defend against the above attacks, DUP-Net [54] using the *Statistical Outlier Removal* (SOR) module and upsampling technique has strong robustness to adversarial examples. Simple Random Sampling (SRS) is used to drop potential adversarial points [51]. Other countermeasures including adversarial training [26, 36, 43], data augmentation, and certified robustness [27, 46], have achieved stellar progress. Recently, PointDP [37] uses the diffusion model to purify adversarial examples and still maintain satisfactory robustness even under strong adaptive attacks.

2.2 Backdoor Attacks on 3D Point Cloud

PCBA [49] implants the backdoor trigger by inserting a cluster of points into the appropriate optimized location near the point cloud. Meanwhile, PointBA [21] designs two methods for embedding the triggers, PointBA-I and PointBA-O respectively. PointBA-I also uses an interaction trigger pattern implanting a ball with a fixed radius and place. PointBA-O alters the orientation of the point clouds by rotation without changing the structure of the original point clouds. Besides, the authors extend them to the clean label backdoor attack via the feature disentanglement. With multiple trigger-embedding modules, Poisoning MorphNet [40] follows the clean label backdoor setting generating the sample-adaptive triggers hidden in the high-frequency domain. Considering the robustness to pre-processing technique, IRBA [6] uses the nonlinear and local transformation to obtain a sample-adaptive trigger, which shows resistance to several pre-processing. Recently proposed NRBdoor [5] is the first uniform trigger generation method that can adapt for both point cloud and 3D mesh based on rotation and adding noise.

2.3 Backdoor Defenses on 2D image

Fine-Pruning represents the most straightforward way to alleviate the damage of backdoor attacks, albeit at the expense of clean accuracy [28]. Nevertheless, the process of purifying the backdoor samples may necessitate intricate operations and pristine data, which can be unacceptable in the real scene. Activation clustering [2] is proposed for detecting training data and filtering trigger samples. However, access to the model by the defender may be impeded due to various reasons, including but not limited to property rights protection. STRIP [7] is the first black-box detection method without access to the victim model, while it requires the logits outputs to calculate the entropy. SCALE-UP [13] and TeCo [29] are the latest test-time backdoor trigger detection methods in total black-box settings. Different from previous works, both of them use label consistency to determine the backdoor samples and clean samples from the perspectives of amplification effects and robustness to image transformations. Unfortunately, with overreliance on the structure of the backdoor pattern, these backdoor defenses for the 2D images domain generally are not suitable for point clouds [3].

It should be noticed that the defense in [50] is used to detect whether the classifier is backdoor infected which is not the topic discussed in this work. Accordingly, it is urgently desired to propose a black-box detection scheme for 3D point clouds requiring zero knowledge about the backdoor trigger pattern and victim model.

3 PRELIMINARIES

Our problem can be formulated as follows, let $f_\theta : \mathcal{X} \to \boldsymbol{y}$ be a 3D point cloud classifier parameterized by θ. Here, $\mathcal{X} \in \mathbb{R}^{n\times 3}$ is the 3D point cloud space, the point cloud $\boldsymbol{X} = \{x_i \in \mathbb{R}^3 | i = 1, \cdots, n\} \in \mathcal{X}$ where x_i is the coordination of the i-th point in point cloud and $\boldsymbol{y} = \{1, \cdots, C\}$. Given a model trainer who wants to train a point cloud model for the classification task, there is an adversary who mounts the backdoor attack by substituting a poisoned set $\mathcal{D}_p$ for a small fraction of the training dataset without the model trainer being aware of it. $\mathcal{D}_p = \{(\hat{X}_i, y_t) | i = 1, \cdots, M\}$, where $\hat{X}_i = \mathcal{T}(X_i)$ is the point cloud implanted with a well-designed trigger $\mathcal{T}(\cdot)$ and y_t is the preset target label. The remaining part of training dataset forms the clean dataset $\mathcal{D}_c$, The backdoor attacker's goal is to generate a special trigger that makes the victim model trained on the poisoned dataset predict the specified target label on any input with the trigger, but perform normally on clean samples. The model trainer will get a backdoored model by solving:

$$\min_{\theta} \sum_{(X,y)\in\mathcal{D}_c} \mathcal{L}\left(f_{\boldsymbol{\theta}}(X), y\right) + \sum_{(\hat{X},y_t)\in\mathcal{D}_p} \mathcal{L}\left(f_{\boldsymbol{\theta}}(\hat{X}), y_t\right) \tag{1}$$

where $\mathcal{L}(\cdot,\cdot)$ denotes the loss function.

As the defender, we attempt to find out a backdoor detection method M to distinguish the backdoor samples from clean samples on a backdoored model $f_{\hat{\theta}}$. Here the test time detection method M can be obtained by the following objective function:

$$M = \arg\max_{M} \mathbb{E}_{X\in D_c}\left[(M(X, f_{\hat{\theta}}) = 0) + (M(\mathcal{T}(X), f_{\hat{\theta}}) = 1)\right] \tag{2}$$

4 CORRUPTION ROBUSTNESS TEST

4.1 Corruption Robustness in Point Cloud

Several point clouds robustness benchmarks have been proposed, ModelNet40-C [38], ModelNet-C [35], and Pointcloud-C [35]. These benchmarks use different corruptions with several severity levels to evaluate the robustness of models. In the same way, we can use it to evaluate the corruption robustness of a point cloud by measuring the model's prediction. This leaves a question to us that *whether point clouds with the backdoor trigger are consistent with the clean point clouds in corruption robustness given a backdoored model.* It is the backdoor trigger that links backdoor samples and target labels. When the backdoor trigger is destroyed by certain corruptions, the backdoor sample will be more or less robust than the benign sample. For example, the rotation triggers are easily destroyed by corruptions based on the input coordinates, the backdoored models will classify the corrupted samples normally.

To explore this further, we visualize one of the samples with the two corruptions used in ModelNet40-C. We compare the benign sample with backdoor samples generated by four popular backdoor attacks [6, 21, 49] in point clouds. As we can see in Fig. 2, the first row represents the shape changes of a benign sample without any implanted trigger after applying various corruptions, provided as a reference. The remaining four rows show backdoor samples implanted with various trigger patterns under corruption. Although transformation-based triggers in the last two rows show good imperceptibility, the geometric shapes of the backdoor samples still undergo significant alteration under corruptions with high severity levels. Given a backdoored model, we conjecture that there is a robustness gap between the benign samples and the backdoor samples under different corruptions.

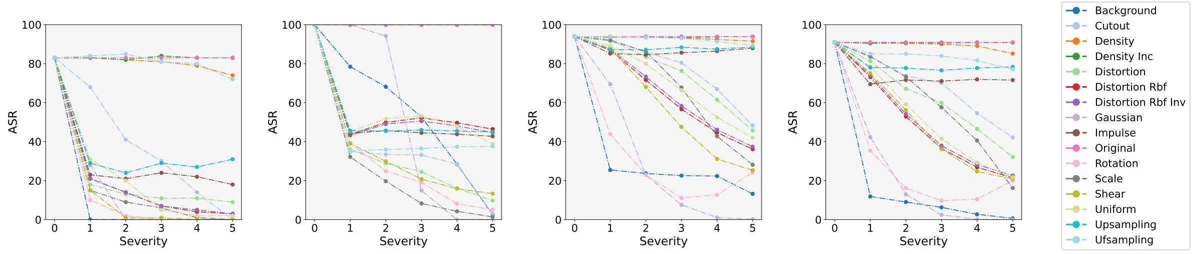
4.2 Evaluation on Corruption Robustness

To further investigate the above assumption, we conduct the Corruption Robustness Test on a victim model under 4 backdoor attacks [6, 21, 49]. Given a point clouds classifier, Corruption Robustness Test imposes a corruption set C_K^N, consisting of K corruption types with N severity levels to the input data, then computes the clean accuracy (ACC) for the clean samples and the attack success rate (ASR) for the samples with the trigger. We utilize 75 corruptions (15 types with 5 severity levels) in [38] to the victim model by 4 different backdoor attacks. As shown in Fig. 3, the shapes and downward trends of these figures differ from one another. We can conduct a rough analysis that the backdoor samples present different corruption robustness with the different types of triggers. The curves of transformation-based triggers, PointBA-O and IRBA have similar shapes but drop steadily compared with the clean samples, which indicates the backdoor sample is more robust than the normal sample from an overall perspective.

Nevertheless, all models show similar performance for the clean point clouds in Fig. 4. Due to the striking resemblance in the shape of the clean samples' ACC curve between models implanted with various backdoor triggers, we can infer that **the corruption robustness of clean data remains stable within a specific range, irrespective of trigger modifications, whereas samples outside this range can be considered as backdoored**. The triggers generated by different backdoor attacks have a profound influence on the corruption robustness of the backdoor samples. Once we are capable of extracting the range of this interval through an algorithmic approach, we can detect all backdoor attacks ideally. Based on that observation, we can distinguish between the backdoor samples and clean samples by comparing the corruption robustness. In the meanwhile, it should be emphasized that this is different from the findings in 2D images. Previous works [7, 23, 29] map the backdoor sample to linearly separable space, as the samples with the trigger are normally more robust than benign samples. However, we have analyzed that interaction-based triggers in point clouds show poor robustness to the most corruptions, the transformation-based triggers enhance the robustness slightly. As a consequence, we need to solve this problem in terms of nonlinear classification.

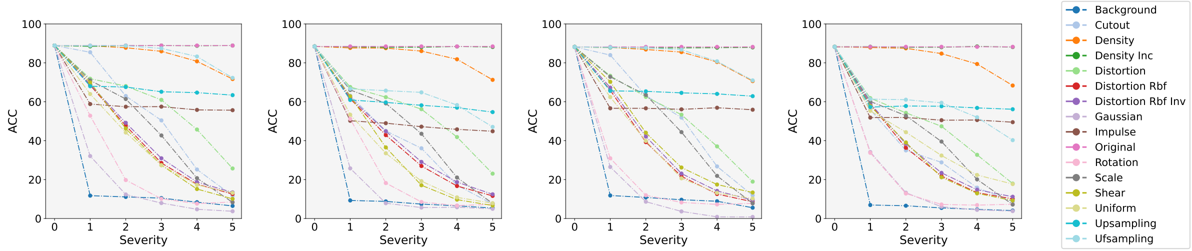
4.3 Corruption Robustness Score

From the above observation, we are going to figure out an algorithm to represent the distinct characteristic of corruption robustness. We are only capable of obtaining the hard-label output of the model, it also ought to be recognized that the ACC (ASR) is calculated for the whole dataset via comparison between the model output



(a) PCBA (b) PointBA-I (c) PointBA-O (d) IRBA

Figure 3: The backdoored PointNet's attack success rate (ASR) for the backdoor sample in ModelNet40 with 4 different backdoor attacks under 15 common corruptions. The four curves have different shapes illustrating that different corruptions can destroy the triggers making ASR descend in different trends.



(a) PCBA (b) PointBA-I (c) PointBA-O (d) IRBA

Figure 4: The backdoored PointNet's accuracy (ACC) for the clean samples in ModelNet40 under 15 common corruptions. The curves are similar to each other, as the inputs are clean without triggers.

and the ground truth (target label), which cannot be available for a suspicious sample.

So we propose *corruption robustness score* (CRS) as the minimal severity that each corruption changes the model prediction to represent the model's robustness to certain corruption C_i:

$$
\begin{aligned}
CRS_i(X) = \arg\min_{s}(f_{\theta^*}(C_i^s(X)) \neq f_{\theta^*}(X)) \\
\text{s.t.} \quad s = 1, \cdots, N \\
i = 1, \cdots, K
\end{aligned}
\tag{3}
$$

where f_{θ^*} is the black-box model. The higher the value is, the more robust the input is to the corruption C_i. It should be underscored that Eq. 3 compares the changes of predicted labels of a sample before and after applying corruption, thus enabling us to solely consider individual samples without requiring other information. The meaning of this formula is equivalent to exploring the slope trends of ACC and ASR, while also providing a more comprehensive evaluation of the robustness of an individual sample under corruptions. After applying all corruption in the set, we can obtain $CRS(X) = \{CRS_i(X)|i = 1, \cdots, K\}$, a K-dimension vector. We can represent the aforementioned observation using the following formula:

$$
\left| d\Big(CRS(\mathcal{T}(X)), CRS(X)\Big) \right| > \gamma \tag{4}
$$

where d is a distance function. However, the issue lies in the fact that the threshold value γ is dependent on the sample X and the type of backdoor trigger $\mathcal{T}(\cdot)$, it is not a fixed value. Therefore, we propose using a binary classifier to directly classify high-dimensional vectors instead of calculating the threshold value for each sample. We feed the CRS vector to a binary classifier $\mathcal{B}$ for the final judgment. Then it can determine the sample with the confidence p if $\mathcal{B}(CRS(X) > p)$ return 1, regard it as a backdoor sample, or we will regard it as a clean sample.

5 EXPERIMENTS

5.1 Experiment Setup

Datasets and models. We conduct our experiments with classic benchmark datasets, ModelNet10 [47], ModelNet40 [47], and ShapeNetPart [1]. ModelNet40 includes 12,311 CAD models from 40 categories, with the split into 9,843 for training and 2,468 for testing. The ModelNet10 is a subset of the ModelNet40 with 10 categories. ShapeNetPart contains 16 categories splitting into 12,128 and 2874 objects for training and testing, respectively. All datasets are sampled into 1024 points uniformly and normalized to [-1, 1]. The victim models include PointNet [33], DGCNN [44], PointNet++ [34], part experiments on CurveNet [32], PCT [14], and Simple View [9].

Implementation details. In this paper, we will adopt 15 corruptions introduced in ModelNet40-C [38] for practical. The 15 corruptions have 5 levels of severity and cover the majority of distortion cases. In order to fully test the robustness of 3D point cloud, we modified the original parameters in ModelNet40-C. We choose the XGBoost (Extreme Gradient Boosting) as our binary classifier. The clean data used for training the binary classifier are collected from the test set and the default sampling ratio is fixed to 10% (*e.g.*, 99 clean samples in ModelNet10, 246 in ModelNet40, 286 in ShapeNetPart) without specific mentioned.

Baselines. In this paper, we focus on the test-time backdoor detection in the black-box setting where defenders can only obtain the hard label of the victim model predictions. The competitors are the latest published methods, SCALE-UP [13], TeCo [29], and we

Table 1: The experiment results on different backdoor attacks, datasets, and models. The last column represents the average performance of a single model under all backdoor attacks. "AVG" stands for the average performance of all backdoor attacks.

Dataset	Model	Attack(⟶) Method(↓)	PCBA F1 score	 AUC	Pointba-I F1 score	 AUC	PointBA-O F1 score	 AUC	IRBA F1 score	 AUC	**AVG** **F1 score**	 **AUC**
ModelNet40	PointNet	STRIP	0.961	0.536	0.510	0.000	0.510	0.283	0.510	0.286	0.623	0.276
		SCALE-UP	0.961	0.412	0.556	0.548	**0.885**	**0.911**	0.510	0.453	0.728	0.581
		TeCo-3D	0.474	0.201	0.496	0.499	0.597	0.803	0.560	0.672	0.532	0.544
		Ours	**0.963**	**0.839**	**0.977**	**0.997**	0.816	0.878	**0.781**	**0.840**	**0.884**	**0.888**
	DGCNN	STRIP	0.961	0.373	0.510	0.217	0.510	0.142	0.510	0.113	0.623	0.211
		SCALE-UP	0.961	0.412	0.974	0.979	0.544	0.534	0.566	0.558	0.761	0.621
		TeCo	0.537	0.215	0.614	0.861	0.552	0.675	0.576	0.739	0.570	0.622
		Ours	**0.961**	**0.759**	**0.996**	**1.000**	**0.808**	**0.870**	**0.844**	**0.914**	**0.902**	**0.886**
	PointNet++	STRIP	0.961	0.170	0.510	0.000	0.510	0.211	0.510	0.173	0.623	0.139
		SCALE-UP	0.961	0.468	0.698	0.696	0.605	0.598	0.652	0.646	0.729	0.602
		TeCo-3D	0.552	0.277	0.658	0.939	0.614	0.869	0.638	0.906	0.615	0.748
		Ours	**0.961**	**0.820**	**0.989**	**0.999**	**0.876**	**0.939**	**0.937**	**0.979**	**0.941**	**0.934**
ShapeNetPart	PointNet	STRIP	0.948	0.391	0.526	0.000	0.563	0.390	0.526	0.405	0.641	0.297
		SCALE-UP	0.531	0.535	0.836	0.899	0.526	0.344	0.526	0.394	0.605	0.543
		TeCo	0.351	0.607	0.628	0.966	0.530	0.659	0.579	0.845	0.522	0.769
		Ours	**0.971**	**0.970**	**0.987**	**0.992**	**0.859**	**0.917**	**0.890**	**0.940**	**0.927**	**0.955**
	DGCNN	STRIP	0.948	0.425	0.606	0.632	0.526	0.276	0.526	0.159	0.652	0.373
		SCALE-UP	0.658	0.686	0.688	0.737	0.554	0.460	0.620	0.619	0.630	0.626
		TeCo-3D	0.340	0.342	0.561	0.836	0.574	0.819	0.551	0.773	0.506	0.692
		Ours	**0.989**	**0.994**	**0.992**	**0.999**	**0.918**	**0.957**	**0.907**	**0.957**	**0.952**	**0.977**
	PointNet++	STRIP	0.948	0.633	0.526	0.000	0.557	0.542	0.526	0.424	0.639	0.400
		SCALE-UP	0.948	0.804	0.952	0.957	0.799	0.875	0.893	0.935	0.898	0.892
		TeCo-3D	0.379	0.774	0.641	0.939	0.581	0.829	0.603	0.888	0.551	0.857
		Ours	**0.989**	**0.992**	**0.962**	**0.989**	**0.932**	**0.980**	**0.943**	**0.980**	**0.956**	**0.985**

Table 2: The characteristics of the evaluated backdoor attacks

	PCBA	PointBA-I	PointBA-O	IRBA
Interaction	✓	✓		
Transformation			✓	✓
Sample-specific				✓

also compared the STRIP [7] which needs more the logits outputs. We modify them to tailor for 3d point clouds, *e.g.*, turning TeCo to TeCo-3D.

Attack settings. Four backdoor attacks are launched to evaluate our defense method, *i.e.*, PCBA [49], PointBA-I [21], PointBA-O [21], and IRBA [6], as shown in Table. 2. All attack methods follow the settings in [6], and the injection ratio is 0.05 of the whole training set. In particular, PCBA requires a (source, target) class pair, we choose ("Night stand", "Table") in ModelNet10, ("Chair", "Toilet") in ModelNet40, and ("Guitar", "Lamp") in ShapeNetPart. What's more, we conduct it as all-to-all attack additionally to simulate the worst scenario that all classes are facing the threat to be stamped triggers, shown in Sec 5.3.

Evaluation metrics. To evaluate the performance of detection method, we use two types of evaluation metrics: *F1 score* and *Area under Receiver Operating Characteristic Curve*, short for AUC, which are widely used in binary classification. With false positive rate (FPR) for the clean samples as the horizontal axis and true positive rate (TPR) for the backdoor samples as the vertical axis, the Receiver Operating Characteristic Curve (ROC) curve can be delineated. The closer AUC is to 1, the better the detection method is to distinguish the backdoor sample and the clean sample. The F1 score can be computed by:

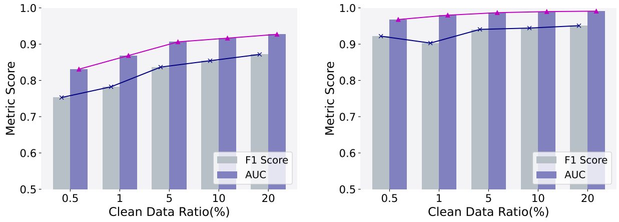
$$F1\ score = \frac{2 \times (precision \times recall)}{precision + recall} \tag{5}$$

Several TPRs and FPRs under different thresholds are used to calculate the AUC, we choose the max F1 score at the same time.

5.2 Performance Evaluation

In Tab 1, we evaluate the performance of PointCRT under different backdoor attacks, datasets, and models[1]. From the table, we can see PointCRT can achieve remarkable performance in most cases with the average *AUC* over 0.938, far surpassing other methods. Here, the SOTA test-time detection methods in 2D images fail to detect backdoor samples in 3D point clouds. The main reason for the unsatisfactory performance of these methods is the data representation gap between 2D images and 3D point clouds. Firstly, STRIP aims to use different clean samples as watermarks to block the possible trigger in 2D images, while in the context of three-dimensional Euclidean space, point clouds exhibit sparsity, which results in gaps when directly overlaying them with one another. The trigger pattern is relatively diminutive compared to point cloud and is less likely to be occluded. Secondly, the postulation in 2D images is not universally applicable to 3D point clouds. Both SCALE-UP and TeCo exhibit significant fluctuations in detection performance across different datasets. In contrast, PointCRT outperforms them and is capable of achieving excellent F1 and AUC scores in almost all circumstances.

[1] During our experiments, we found SimpleView and PCT are robust to backdoor attacks on certain datasets. For example, the PCBA is not strong enough to implant trigger on them, which makes the trigger ineffective and hinders to evaluate.



(a) PointNet / PointBA-O (b) SimpleView / PointBA-I

Figure 5: Illustrations of PointCRT’s performance with different clean samples

Table 3: The comparison of all-to-all attack on ShapeNetPart

Model	PointNet		DGCNN		PointNet++	
Method	F1 score	AUC	F1 score	AUC	F1 score	AUC
SCALE-UP	0.500	0.380	0.500	0.248	0.500	0.341
Ours	**0.888**	**0.943**	**0.951**	**0.984**	**0.929**	**0.977**

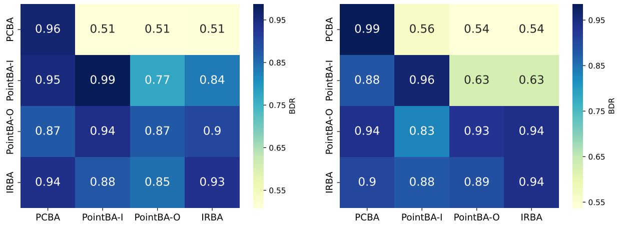
5.3 Ablation Study

Performance on different numbers of clean data. We investigate PointCRT’s performance under different sampling ratios on clean dataset. We illustrate two types of classic triggers’ results on ModelNet40 in Fig. 5. For PointBA-I, we choose SimpleView as the victim model, and for PointBA-O, we select PointNet. It should be highlighted that in extremely small quantities of clean data (*e.g.*, only 12 clean samples in ModelNet40 for PointBA-I), PointCRT still achieves $AUC \geq 0.82$. In the end, we strike a balance between performance and data requirements by setting the sampling ratio of test datasets to 10%, which proves to be effective in meeting our desired outcome.

Performance against all-to-all attacks. As PCBA launches the attack by source-target pair, it is possible for us to mount the all-to-all attack. In this part, we are going to explore the worst case that all classes are embedded with backdoor trigger. We perform the experiment on ShapeNetPart and the target label is set by turning y_i to $y_i + 3$. From the Table. 3, SACLE-UP is completely incapacitated and even starts making the opposite decisions, while PointCRT is still able to maintain its performance.

Resistance to data augmentations. We consider a possible situation that corruption is used for data augmentation in the backdoor training. In this experiment, we apply data augmentation by randomly applying corruption to the sample during training. As shown in Table. 4, there is no significant degradation in the performance, PointCRT shows effective resistance to data augmentation.

Transferability to unseen attacks. We explore the transferability of PointCRT to unseen attacks after the XGBoost is trained on the known attack. To simplify, we use the backdoor detection rate (BDR) as the metric to measure the detecting performance of PointCRT. Fig. 6 shows the method on PointNet++ is transferred to detect the corresponding attack in the same row. For example, the first row in Fig. 6(b) means the victim model is attacked by PCBA, PointCRT detects PCBA’s backdoor samples with the BDR of 0.99, while it only has about 0.54 facing other unseen attacks . To this end, we believe using transformation-based backdoor attacks as the basic method is capable of effectively handling the majority of 3D point cloud backdoor attack scenarios that arise in real-world situations.



(a) ModelNet40 (b) ShapeNetPart

Figure 6: Transferability of PointCRT on unseen backdoor attacks

Table 4: The results of PointCRT on PointNet++ using data augmentations

Dataset	ModelNet10		ModelNet40		ShapeNetPart	
Attack	F1 score	AUC	F1 score	AUC	F1 score	AUC
PCBA	0.924	0.906	0.961	0.805	0.977	0.972
Pointba-I	0.981	0.998	0.949	0.987	0.969	0.995
PointBA-O	0.898	0.953	0.843	0.911	0.910	0.968
IRBA	0.791	0.837	0.831	0.905	0.938	0.987

5.4 Performance on real dataset

KITTI Vision Benchmark Suite [31] is one of the most famous real benchmarks used for autonomous driving. Here, we add our experiment of detecting PCBA on KITTI following PCBA’s original setting. What’s more, we evaluate the performance with different metrics to demonstrate the overall capacity of our approach, *e.g.*, precision and the False Acceptance Rate (FAR). From Table. 5, we can see PointCRT is still able to detect the backdoor samples with average AUC over 0.94. This is reasonable as CRS does not consider the feature of datasets, but rather focuses on the label consistency of individual samples to evaluate their corruption robustness. The experiment results verify our previous observation that the clean sample’s corruption robustness is stable on real datasets.

Table 5: The performance of PointCRT detecting PCBA on KITTI dataset

Model	F1 score(↑)	AUC(↑)	Precision(↑)	FAR(↓)
PointNet	0.850	0.900	0.856	0.141
DGCNN	0.961	0.982	0.979	0.019
PointNet++	0.951	0.951	0.953	0.047

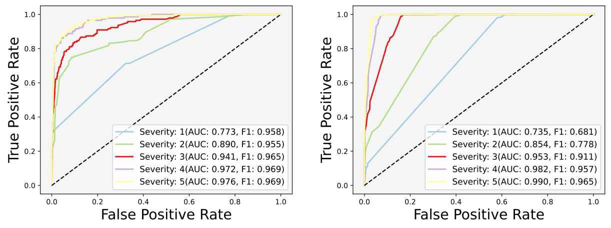
6 DISCUSSION

6.1 Detecting the Backdoored Models

We start by investigating PointCRT on detecting adversarial examples. Adversarial examples are generated by JGBA [30] in ModelNet40, the target label is the same as the all-to-all attack experiment’s setting y_i to $y_i + 3$. All adversarial examples and benign samples apply the corruptions under the default settings as before.

Table 6: The results of PointCRT's performance under different combinations of corruptions. "w/o density" means dropping the density group from the corruption sets.

Corruption Group	PCBA		PointBA-I		PointBA-O		IRBA		**AVG**	
	F1 score	AUC	F1 score	AUC	F1 score	AUC	F1 score	AUC	**F1 score**	**AUC**
3 (select 1 per group)	0.946	0.818	0.901	0.951	0.765	0.793	0.891	0.928	**0.876**	**0.873**
6 (select 2 per group)	0.956	0.879	0.949	0.988	0.850	0.901	0.913	0.959	**0.917**	**0.932**
9 (select 3 per group)	0.955	0.918	0.967	0.993	0.869	0.930	0.914	0.966	**0.926**	**0.952**
w/o density	0.954	0.903	0.949	0.988	0.849	0.910	0.887	0.942	**0.910**	**0.936**
w/o noise	0.947	0.860	0.959	0.991	0.855	0.912	0.914	0.962	**0.919**	**0.931**
w/o transformation	0.959	0.899	0.970	0.993	0.868	0.928	0.927	0.961	**0.931**	**0.945**
all	0.961	0.918	0.979	0.997	0.878	0.940	0.927	0.973	**0.936**	**0.957**



(a) CurveNet/PCBA

(b) PCT/PointBA-I

Figure 7: Comparison of PointCRT's performance under different maximum severity. The performance improves as the maximum severity N increases.

In the end, the PointCRT fails in this toy example, the F1 score and AUC both are 0.5, which means the classifier is random guessing.

The core concept is that in backdoor models, the mapping from trigger to target label is usually stronger than in normal samples. Therefore, a normal model without a backdoor implant can not distinguish between malicious and normal samples, which can explain why PointCRT fails in detecting adversarial examples.

So to detect whether the suspicious model is backdoored, we can obtain the CRS by feeding a clean sample to the suspicious model, and use a pre-trained binary classifier to make a decision. If the suspicious model is not compromised, the classifier may misclassify it as a backdoor sample with the confidence of approximately 50%, but if the model is actually backdoored, the classifier would identify it as a clean sample correctly.

6.2 The Option of Corruptions

One important question still remains to be answered, *i.e.*, can we choose the corruption set arbitrarily? We will answer this question from two perspectives, corruption levels of severity and corruption types.

Firstly, we set up the maximum severity ranging from 1 to 5. For the sake of fairness, we choose a combination with a relatively close attack success rate for backdoor attacks and victim model pairs on ShapeNetPart. As shown in Fig. 7, it consistently demonstrates good performance even at low N values.

Secondly, these corruptions represent different kinds of point cloud attributes as mentioned in [35], and can be grouped into 3 categories: density patterns, noise patterns, and transformation patterns. We randomly selected different combinations of corruptions and investigate these combinations groups on PointNet++ under 4 backdoor attacks. As demonstrated in Table. 6, the quantity of corruption presents a noteworthy impact on PointCRT's efficiency, while the types of corruption exhibit feeble influence. We hypothesize that the certain corruptions may have the equivalent impact on a given backdoor sample, rendering the effect of these corruptions redundant and unable to fully demonstrate robustness, leading to fluctuations in performance.

It is worth noting that, the computational cost of PointCRT is highly related to the above corruptions types K and the max severity level N. Hence, we hold the opinion that excavating unknown corruption serving as a substitute for the part of corruption sets can elevate PointCRT's performance and efficiency.

7 CONCLUSION

In this paper, we propose PointCRT, the first scheme on black-box backdoor sample detection for 3D point clouds. Via applying different corruptions on the inputs, PointCRT evaluates the input's corruption robustness and detects the backdoor samples by their abnormal corruption robustness without any prior knowledge of the trigger patterns. Our experimental results demonstrate the effectiveness of our approach in detecting backdoor attacks in 3D point clouds, addressing the challenges posed by unique and sparse data formats and the imperceptibility of transformation-based triggers.

8 LIMITATIONS AND FUTURE WORKS

When the trigger implants to the victim model badly, the detection performance of PointCRT is also not good. We believe further investigation should be focused on better performance in all cases in the future. Similar to previous methods that require obtaining a threshold through clean data, PointCRT also requires clean samples as a reference. Our next step is to explore how to purify the detected backdoor samples by corruptions to make full use of the dataset without clean data.

ACKNOWLEDGMENTS

Shengshan's work is supported in part by the National Natural Science Foundation of China (Grant No.U20A20177) and Hubei Province Key R&D Technology Special Innovation Project under Grant No.2021BAA032. Minghui's work is supported in part by the National Natural Science Foundation of China (Grant No. 62202186) Minghui Li is the corresponding author.

REFERENCES

[1] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet: An information-rich 3D model repository. *arXiv preprint arXiv:1512.03012* (2015).

[2] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Ben Edwards, Taesung Lee, Ian Molloy, and B. Srivastava. 2018. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728* (2018).

[3] Edward Chou, Florian Tramèr, Giancarlo Pellegrino, and Dan Boneh. 2018. SentiNet: Detecting physical attacks against deep learning systems. *arXiv preprint arXiv:1812.00292* (2018).

[4] Yinpeng Dong, Xiao Yang, Zhijie Deng, Tianyu Pang, Zihao Xiao, Hang Su, and Jun Zhu. 2021. Black-box detection of backdoor attacks with limited information and data. In *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV'21)*. 16462–16471.

[5] Linkun Fan, Fazhi He, Qingchen Guo, Wei Tang, Xiaolin Hong, and Bing Li. 2022. Be careful with rotation: A uniform backdoor pattern for 3D shape. *arXiv preprint arXiv:2211.16192* (2022).

[6] Kuofeng Gao, Jiawang Bai, Baoyuan Wu, Mengxi Ya, and Shutao Xia. 2022. Imperceptible and robust backdoor attack in 3D point cloud. *arXiv preprint arXiv:2208.08052* (2022).

[7] Yansong Gao, Chang Xu, Derui Wang, Shiping Chen, Damith Chinthana Ranasinghe, and Surya Nepal. 2019. STRIP: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference (ACSAC'19)*. 113–125.

[8] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence* 2 (2020), 665 – 673.

[9] Ankit Goyal, Hei Law, Bowei Liu, Alejandro Newell, and Jia Deng. 2021. Revisiting point cloud shape classification with a simple and effective baseline. In *Proceedings of the 2021 International Conference on Machine Learning (ICML'21)*, Vol. 139. 3809–3820.

[10] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2019. BadNets: Evaluating backdooring attacks on deep neural networks. *IEEE Access* 7 (2019), 47230–47244.

[11] Jun Guo, Wei Bao, Jiakai Wang, Yuqing Ma, Xinghai Gao, Gang Xiao, Aishan Liu, Jian Dong, Xianglong Liu, and Wenjun Wu. 2023. A Comprehensive Evaluation Framework for Deep Model Robustness. *Pattern Recognition* (2023).

[12] Junfeng Guo, Ang Li, and Cong Liu. 2022. AEVA: Black-box backdoor detection using adversarial extreme value analysis. In *Proceedings of the 2022 International Conference on Learning Representations (ICLR'22)*.

[13] Junfeng Guo, Yiming Li, Xun Chen, Hanqing Guo, Lichao Sun, and Cong Liu. 2023. SCALE-UP: An efficient black-box input-level backdoor detection via analyzing scaled prediction consistency. In *Proceedings of the 2023 International Conference on Learning Representations (ICLR'23)*,.

[14] Mengao Guo, Junxiong Cai, Zhengning Liu, Taijiang Mu, Ralph Robert Martin, and Shimin Hu. 2020. PCT: Point cloud transformer. *Computational Visual Media* 7 (2020), 187–199.

[15] Abdullah Hamdi, Sara Rojas, Ali K. Thabet, and Bernard Ghanem. 2020. AdvPC: Transferable adversarial perturbations on 3D point clouds. *In 16th European Conference on Computer Vision (ECCV'20)*, 241–257.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*. 770–778.

[17] Shengshan Hu, Junwei Zhang, Wei Liu, Junhui Hou, Minghui Li, Leo Yu Zhang, Hai Jin, and Lichao Sun. 2023. PointCA: Evaluating the robustness of 3D point cloud completion models against adversarial examples. *In Proceedings of the AAAI Conference on Artificial Intelligence(AAAI'23)* 2023 (Jun. 2023), 872–880.

[18] Shengshan Hu, Ziqi Zhou, Yechao Zhang, Leo Yu Zhang, Yifeng Zheng, Yuanyuan He, and Hai Jin. 2022. Badhash: Invisible backdoor attacks against deep hashing with clean label. In *Proceedings of the 30th ACM International Conference on Multimedia (ACM MM'22)*. 678–686.

[19] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Hang Zhou, Weiming Zhang, and Nenghai Yu. 2022. Shape-invariant 3D adversarial point clouds. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22)*. 15314–15323.

[20] Kaidong Li, Ziming Zhang, Cuncong Zhong, and Guanghui Wang. 2022. Robust structured declarative classifiers for 3D point clouds: Defending adversarial attacks with implicit gradients. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22)*. 15273–15283.

[21] Xinke Li, Zhirui Chen, Yue Zhao, Zekun Tong, Yabang Zhao, Andrew Lim, and Joey Tianyi Zhou. 2021. PointBA: Towards backdoor attacks in 3D point cloud. In *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV'21)*. 16472–16481.

[22] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. 2018. PointCNN: Convolution on -transformed points. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NeurIPS'18)*. Curran Associates Inc., 828–838.

[23] Yiming Li, Tongqing Zhai, Baoyuan Wu, Yong Jiang, Zhifeng Li, and Shutao Xia. 2020. Rethinking the trigger of backdoor attack. *arXiv preprint arXiv:2004.04692* (2020).

[24] Qi Liang, Qiang Li, and Song Yang. 2021. LP-GAN: Learning perturbations based on generative adversarial networks for point cloud adversarial attacks. *Image Vis. Comput.* 120 (2021), 104370.

[25] Daizong Liu and Wei Hu. 2023. Imperceptible transfer attack and defense on 3D point cloud classification. *IEEE transactions on pattern analysis and machine intelligence* 45, 4 (2023), 4727–4746.

[26] Daniel Liu, Ronald Yu, and Hao Su. 2019. Extending adversarial attacks and defenses to deep 3D point cloud classifiers. In *Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP'19)*. 2279–2283.

[27] Hongbin Liu, Jinyuan Jia, and Neil Zhenqiang Gong. 2021. PointGuard: Provably robust 3D point cloud classification. In *Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*. 6182–6191.

[28] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2018. Fine-Pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Recent Advances in Intrusion Detection (RAID'18)*. 273–294.

[29] Xiaogeng Liu, Minghui Li, Hao Wang, Shengshan Hu, Dengpan Ye, Hai Jin, Libing Wu, and Chaowei Xiao. 2023. Detecting backdoors during the inference stage based on corruption robustness consistency. In *Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'23)*.

[30] Chengcheng Ma, Weiliang Meng, Baoyuan Wu, Shibiao Xu, and Xiaopeng Zhang. 2020. Efficient joint gradient based attack against SOR defense for 3D point cloud classification. In *Proceedings of the 28th ACM International Conference on Multimedia (MM'20)*. 1819–1827.

[31] Moritz Menze and Andreas Geiger. 2015. Object scene flow for autonomous vehicles. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR' 15)*.

[32] A. A. M. Muzahid, Wanggen Wan, Ferdous Sohel, Lianyao Wu, and Li Hou. 2021. CurveNet: Curvature-based multitask learning deep networks for 3D object recognition. *IEEE/CAA Journal of Automatica Sinica* 8, 6 (2021), 1177–1187.

[33] C. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. 2017. PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'17)*. 77–85.

[34] C. Qi, L. Yi, Hao Su, and Leonidas J. Guibas. 2017. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS'17)*. 5105–5114.

[35] Jiawei Ren, Liang Pan, and Ziwei Liu. 2022. Benchmarking and analyzing point cloud classification under corruptions. *arXiv preprint arXiv:2202.03377* (2022).

[36] Jiachen Sun, Yulong Cao, Christopher Bongsoo Choy, Zhiding Yu, Anima Anandkumar, Zhuoqing Morley Mao, and Chaowei Xiao. 2021. Adversarially robust 3D point cloud recognition using self-supervisions. In *Proceedings of the 34st International Conference on Neural Information Processing Systems (NeurIPS'21)*. 15498–15512.

[37] Jiachen Sun, Weili Nie, Zhiding Yu, Zhuoqing Morley Mao, and Chaowei Xiao. 2022. PointDP: Diffusion-driven purification against adversarial attacks on 3D point cloud recognition. *arXiv preprint arXiv:2208.09801* (2022).

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

[39] Di Tang, XiaoFeng Wang, Haixu Tang, and Kehuan Zhang. 2021. Demon in the variant: Statistical analysis of DNNs for robust backdoor contamination detection. In *Proceedings of the 30th USENIX Security Symposium (USENIX Security'21)*. USENIX Association, 1541–1558.

[40] Guiyu Tian, Wenhao Jiang, Wei Liu, and Yadong Mu. 2021. Poisoning MorphNet for clean-label backdoor attack to point clouds. *arXiv preprint arXiv:2105.04839* (2021).

[41] Sakshi Udeshi, Shanshan Peng, Gerald Woo, Lionell Loh, Louth Rawshan, and Sudipta Chattopadhyay. 2022. Model agnostic defence against backdoor attacks in machine learning. *IEEE Transactions on Reliability* 71, 2 (2022), 880–895.

[42] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y. Zhao. 2019. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP'19)*. 707–723.

[43] Robin Wang, Yibo Yang, and Dacheng Tao. 2022. ART-Point: Improving rotation robustness of point cloud classifiers via adversarial rotation. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22)*. 14351–14360.

[44] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. 2019. Dynamic graph cnn for learning on point clouds. In *Acm Transactions On Graphics (TOG'19)*, Vol. 38. 1–12.

[45] Yuxin Wen, Jiehong Lin, Ke Chen, C. L. Philip Chen, and Kui Jia. 2019. Geometry-aware generation of adversarial point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44 (2019), 2984–2999.

[46] Ziyi Wu, Yueqi Duan, He Wang, Qingnan Fan, and Leonidas J. Guibas. 2020. IF-Defense: 3D adversarial point cloud defense via implicit function based restoration. *arXiv preprint arXiv:2010.05272* (2020).

[47] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the 2015 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'15)*. 1912–1920.

[48] Chong Xiang, C. Qi, and Bo Li. 2018. Generating 3D adversarial point clouds. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*. 9128–9136.

[49] Zhen Xiang, David J. Miller, Siheng Chen, Xi Li, and George Kesidis. 2021. A backdoor attack against 3D point cloud classifiers. In *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV'21)*. 7577–7587.

[50] Zhen Xiang, David J. Miller, Siheng Chen, Xi Li, and George Kesidis. 2022. Detecting backdoor attacks against point cloud classifiers. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'22)*. 3159–3163.

[51] Jiancheng Yang, Qiang Zhang, Rongyao Fang, Bingbing Ni, Jinxian Liu, and Qi Tian. 2019. Adversarial attack and defense on point sets. *arXiv preprint arXiv:1902.10899* (2019).

[52] Yue Zhao, Yuwei Wu, Caihua Chen, and Andrew Lim. 2020. On isometry robustness of deep 3D point cloud models under adversarial attacks. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*. 1198–1207.

[53] Tianhang Zheng, Changyou Chen, Junsong Yuan, Bo Li, and Kui Ren. 2019. PointCloud saliency maps. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV'19)*. 1598–1606.

[54] Hang Zhou, Kejiang Chen, Weiming Zhang, Han Fang, Wenbo Zhou, and Nenghai Yu. 2019. DUP-Net: Denoiser and upsampler network for 3D adversarial point clouds defense. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV'19)*. 1961–1970.