Robustifying Point Cloud Networks by Refocusing

Meir Yossef Levi, Guy Gilboa
Viterbi Faculty of Electrical and Computer Engineering
Technion - Israel Institute of Technology, Haifa, Israel
me.levi@campus.technion.ac.il ; guy.gilboa@ee.technion.ac.il

Abstract

The ability to cope with out-of-distribution (OOD) corruptions and adversarial attacks is crucial in real-world safety-demanding applications. In this study, we develop a general mechanism to increase point clouds neural networks robustness based on focus analysis. Recent studies have revealed the phenomenon of Overfocusing, which leads to a performance drop. When the network is primarily influenced by small input regions, it becomes less robust and prone to misclassify under noise and corruptions. However, quantifying overfocusing is still vague and lacks clear definitions. Here, we provide a mathematical definition of ***focus****,* ***overfocusing*** *and* ***underfocusing****. The notions are general, but in this study, we specifically investigate the case of 3D point clouds. We observe that corrupted sets result in a biased focus distribution compared to the clean training set. We show that as focus distribution deviates from the one learned in the training phase - classification performance deteriorates. We thus propose a parameter-free* ***refocusing*** *algorithm that aims to unify all corruptions under the same distribution. We validate our findings on a 3D zero-shot classification task, achieving SOTA in robust 3D classification on ModelNet-C dataset, and in adversarial defense against Shape-Invariant attack.*

1. Introduction

In recent years, significant research efforts have been dedicated to understanding the underlying mechanisms of neural networks for producing accurate and reliable results. One important area of study is explainable AI (XAI), which aims to determine where the network "looks" within an image to establish classification [2, 3, 37]. It is well known that networks may rely on spurious cues [14] or shortcuts [7] for prediction. For instance, a cow may be predicted based on a green pasture rather than cow features, influenced by dataset bias. XAI algorithms typically propagate gradient computations to provide a heatmap, showing regions in the image that mostly contributed to the network's decision. While numerous explainable approaches exist in various fields, methods to effectively analyze these heatmaps remain unclear. Consequently, there are limited tools available for analyzing XAI maps, particularly those fast enough to be embedded for decision-making during the inference phase. We propose to analyze XAI heatmaps through the lens of focus analysis.

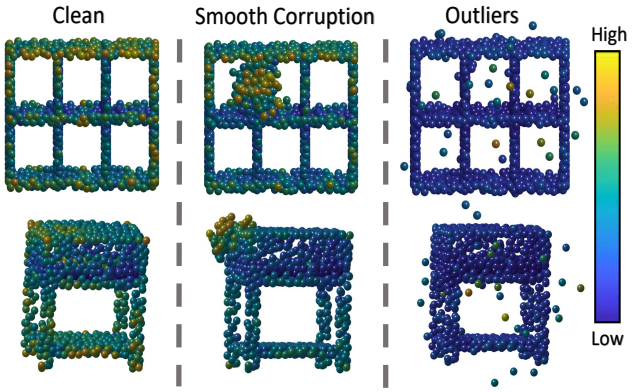


Figure 1. **Outliers and smooth corruptions often draw high influence.** Samples color coded according to influence. The measurement of influence, well distributed according to meaningful features in uncorrupted point clouds, is redistributed mainly to outliers in the corrupted version.

By measuring the influential regions (using existing XAI tools), it is possible to investigate the relations between the *focus* of the network and its performance. Although a definition of attention concentration based on entropy was introduced in [8], the term focus, in a general context, remains somewhat ambiguous and not well-defined. Intuitively, a focused network is influenced by only a few prominent input data points, while an unfocused one relies on input data spread throughout the domain. See Fig. 2 for point cloud examples of high and low focus.

Recent studies have shown that many networks tend to overfocus, making decisions based only on a few highly localized input regions [10]. This results in less stable performance and lacks robustness when statistics change with

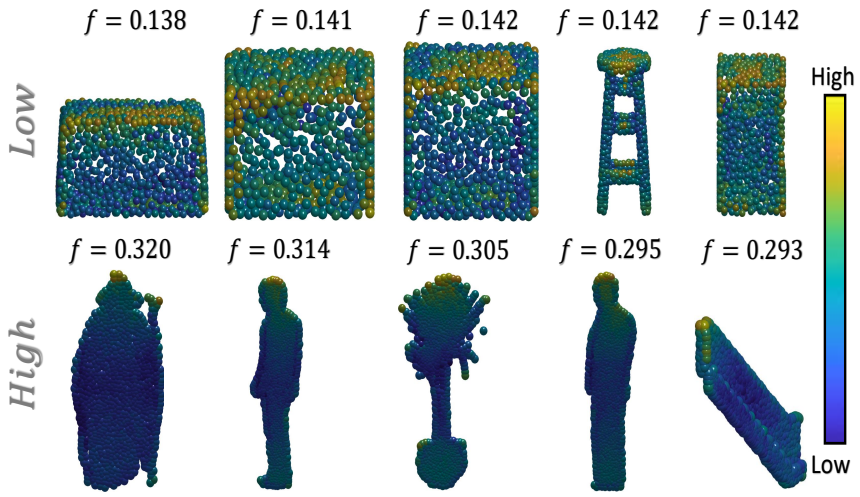


Figure 2. **High and low focus examples.** Samples resulting in low focus distribute influence across a wide spatial range of points, while in the case of high focus, influence becomes concentrated within specific regions. Samples containing flat areas, with some extent of symmetry against the center of the shape, contribute to a decrease in focus. It is possible that samples predominantly spanning a 2D plane prompt the network to prioritize attention towards distinctive regions. Points are color-coded by influence.

respect to the training phase. We investigate classification robustness in the context of 3D point clouds, examining the interplay between focus and robustness to corruptions and to adversarial attacks. We first define a general notion of the network's focus based on normalized entropy. We then analyze the focus distribution and its changes under various corruptions, using a recently proposed corrupted point cloud benchmark, ModelNet-C [35]. Our findings reveal that the clean dataset, used during training, has a distinct focus distribution for which the network is optimal. Each corruption type induces a different unique focus distribution. The general trend is that corruptions involving outliers cause overfocus, while those involving occluded parts cause underfocus, compared to the uncorrupted data (See Fig. 3). This leads to significant performance degradations.

We propose a new learning procedure that reduces the variance of the focus distribution under corruptions. In a nutshell, we train the network to perform a more challenging task, relying on less influential input points. This results in a less focused network during training. Subsequently, by applying the same filtering during inference, which may contain corruptions, we achieve a focus distribution which is more aligned to the training phase. A more stable network is obtained, with improved robustness to out-of-distribution (OOD) corruptions, effectively balancing overall performance on clean samples. This generic idea can enhance the robustness of various point cloud classification networks. We demonstrate it on DGCNN [44], RPC [35], and GDANet [52].

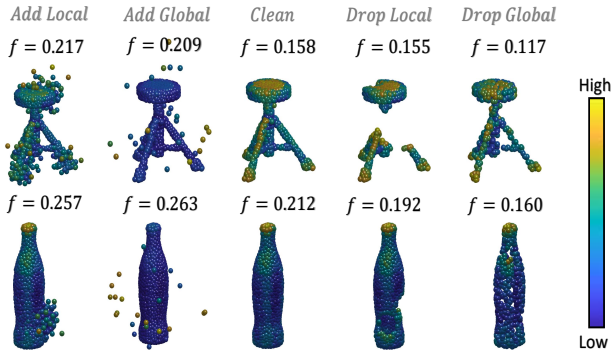


Figure 3. **Variation of focus across different corruptions.** Influence maps on corrupted samples from ModelNet-C [35] using DGCNN [44]. The presence of outliers predominantly increases focus, while occluded parts decrease it. Points are color-coded according to influence. We denote by f the focus of the network for that sample, as defined in Eq. (3).

1.1. Why rely on less influential inputs?

Our approach might seem counterintuitive, as one would expect prominent regions to contribute most to high-quality class discrimination. However, relying on less influential points offers several important benefits:

1. **Calibrated focus.** Our analysis demonstrates that the network performs optimally for data within the focus distribution of the training phase (Fig. 6). The proposed learning procedure of refocusing by screening out the influential points is highly stable, yielding a similar focus distribution for OOD samples as for the clean set, as illustrated in Fig. 4.

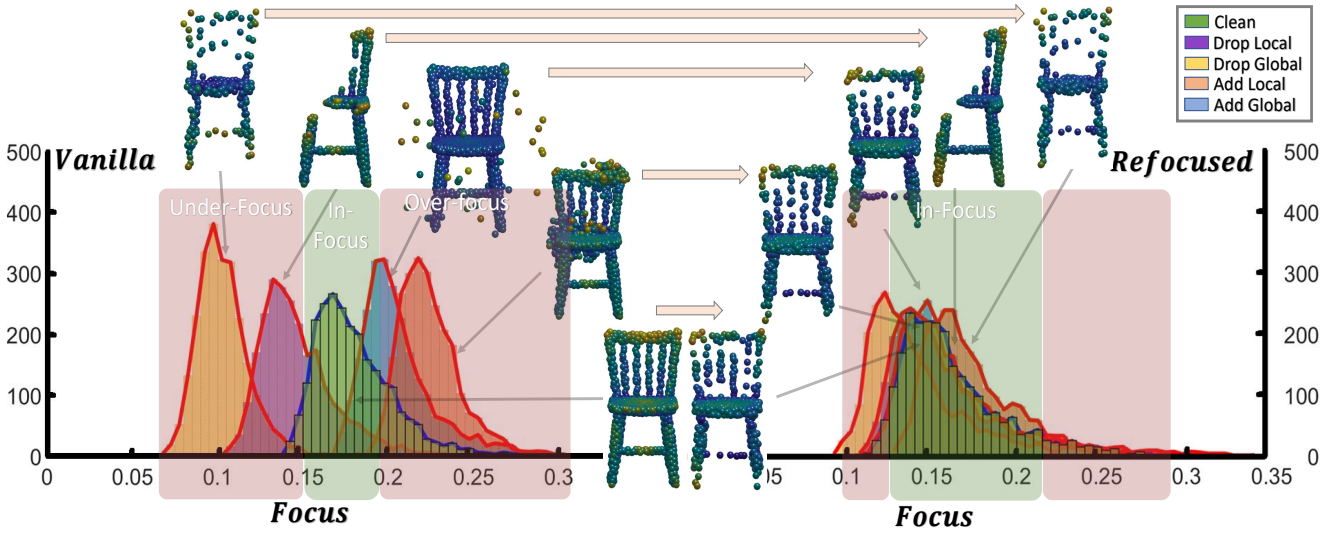


Figure 4. **Refocusing.** Left - Focus distribution on DGCNN [44] based on the clean set of ModelNet40 [47] and on corrupted sets (ModelNet-C [35]). Corrupted samples deviate from the in-focus region. Right - Screening out influential points align the focus distribution, expanding the in-focus region at the expense of under- and over-focus regions. In the chair example shown, one can observe the network is influenced by similar points after refocusing, resembling roughly the same influence distribution across the shape.

2. **Better resilience to corruptions.** Corruptions are often perceived as influential features by the network, leading to a significant performance drop, as shown in Fig. 1. Our approach exhibits implicit significant filtering capacity of outliers. In Tab. 1, we conducted a comparison between filtering the most influential and filtering the less influential points to analyze the trade-off between accuracy on the clean set and on the corrupted set. We find that relying on the less influential points significantly enhances robustness to outliers at the expense of only a slight performance drop on the clean set (i.e. for DGCNN [44], a significant improvement in the Mean Corruption Error, reducing it from 1.000 to 0.688).
3. **Preserving clean data performance.** Sub-sampling, known for enhancing robustness [22, 25], may cause a slight performance drop. However, this can be compensated through ensemble methods [22], which, overall, can surpass vanilla performance, as validated in Tab. 2. Integrating our approach with the ensemble classification method EPiC [22] yields competitive results on clean samples while significantly improving robustness.

Our main contributions are:

1. We provide a general definition for the network's focus and for over- and under-focusing. A comprehensive analysis is performed relating corruptions to the focus distribution.
2. We propose a refocusing scheme, which is parameter-free and can be applied to any point-cloud classification network. It offers a more robust and reliable learning strategy, handling OOD corruptions and adversarial attacks better, without sacrificing overall accuracy.
3. We demonstrate our approach in *Robust Classification* and *Adversarial Defense*, achieving state-of-the-art results.

	Accuracy	
	Clean	Add-Global [35]
Least Influential	91.0%	**90.3%**
Most Influential	91.1%	43.2%

Table 1. **Test accuracy of most vs. least influential inputs.** While both approaches perform comparably on the clean set, using less influential points is much more robust to outliers. Trained and evaluated on 600 least or most influential.

2. Related Work

Focus in neural networks. Focus has been investigated until now mainly in the context of attention maps of transformers. The studies of [5] and [29] have highlighted the tendency of attention maps in certain network layers to rely heavily on a few dominant tokens. Guo et al. [10] coined this phenomenon as *Token Overfocusing* and established a correlation with corruptions. While there have been visual demonstrations of overfocused attention maps, we lack a clear and standardized definition of focus, which may be applied to any type of network. Ghader et al. [8] introduced *Attention Concentration* defined mathematically using attention entropy. This definition has allowed for the analysis of attention concentration in different parts of sentences

in natural language processing (NLP) [43], and the investigation of differences between supervised and unsupervised training in terms of attention entropy [30]. However, it is primarily designed for attention maps in transformers and cannot be applied to general neural network architectures. Additionally, it does not rely on normalized entropy, which plays a vital role when dealing with varying numbers of input elements.

3D robust classification. Point cloud classification [9, 26, 28, 32, 33, 44, 50–52, 56] is vital for autonomous driving [4, 57] and robotics [42]. However, research on robustness against corruptions is relatively scarce. PointNet [32] introduced the concept of *critical points*, which is a subset of points that remain active after the last pooling layer. We note that outliers are often misinterpreted as influential or critical. Supervised [6, 19] and unsupervised [55] 3D sorting strategies have been proposed to better sample point clouds for downstream tasks. These approaches prone to underperform at the presence of out-of-distribution (OOD) corruptions since they prioritize highly influential points. Our observations indicate outliers are highly likely to be sampled by these methods. Several studies [34, 53] offered learnable outlier removal for adaptive sampling in Euclidean space. ModelNet-C dataset [35] introduced real-world corruptions, involving outliers or missing points (which can be caused by occlusions) from 3D point clouds, either globally or locally. They also proposed Robust Point-cloud Classifier (RPC) [35], an algorithm which is a combination of the most robust modules from typical classification networks, achieving state-of-the-art performance on ModelNet-C. Recently, EPiC [22] proposed an ensemble approach combining different sampling schemes, outperforming RPC. However, such ensemble methods are relatively resource-intensive.

3D adversarial attacks. Designing classification networks which are robust against adversarial attacks, particularly in 3D settings, is significant. Numerous 3D adversarial attack methods have emerged in recent years [13, 17, 21, 24, 39, 41, 45, 49, 54, 61–63]. These attacks primarily focus on perturbing points, emphasizing imperceptible manipulations. Our proposed influence measure and point filtering approach can be employed for adversarial defense. Shape-Invariant Attack [15] introduces a sensitivity map consistent across diverse neural networks, sliding points along the tangent plane based on this map. Point Cloud Saliency Maps [62] analyze gradient loss when shifting points to the spherical center to determine importance. The vulnerability of critical points [32] has been used to design several attack strategies, such as [45, 49, 54]. We compare our proposed defensive scheme against the state-of-the-art Shape-Invariant Attack [15], highlighting that even imperceptible perturbations can alter point influence, demonstrating the generality of our approach.

3D adversarial defense. Advanced 3D augmentation techniques like PointWolf [18] and RSMix [20] enhance network robustness against corruptions [35]. Adversarial Training (AT) techniques [12, 16, 23, 31, 40, 54, 59] intentionally introduce perturbations during training to defend against malicious attacks, but they require prior knowledge, therefore not robust for OOD corruptions. PointGuard [25] and PointCert [58] propose certified defense schemes using point-cloud sub-sampling and majority voting. However, their ensemble strategy is highly demanding computationally. Point filtering techniques include Simple Random Sampling (SRS) [64], which removes input points randomly, and Statistical Outlier Removal (SOR) [36], which filters points far from their nearest neighbors. SOR performs well against outliers (with known distributions) but may fail on smooth corruptions. Dup-Net [64] combines denoising and upsampling, significantly increasing inference time. LPF-Defense [12] focuses on low-frequency features using spherical harmonics transformation. IF-Defense [48] optimizes surface distortion, requiring a training phase during inference. In [23] prediction derivatives are calculated to obtain per-point importance, facing scalability challenges with large networks. Our work aims to address these challenges. In our comparison we use Shape-Invariant attack [15] and LPF-Defense [12] as baselines.

3. Focus Definition and Refocusing Method

Let $X \in \mathbb{R}^{N\times d}$ be the input data consisting of N elements $X_i \in \mathbb{R}^d$ in arbitrary dimension d. Let $F : \mathbb{R}^{N\times d} \to \mathbb{R}^m$ be a classification neural network for m classes. We denote by $I_F(X) : \mathbb{R}^{N\times d} \to \mathbb{R}^N$ an influence score. This measure attempts to quantify, for a given X, the amount of influence of each input element on the output of the network F. The term *influence* may be defined in various ways with multiple approaches to compute it, such as based on attention mechanisms or in the general case by using XAI methods. We denote by I_F^i the i^{th} element of $I_F(X)$. It is further assumed that the influence score is normalized, such that it has only non-negative values with a unit sum, that is, $I_F^i \geq 0$, $\forall i = \{1, .., N\}$, and $\sum_{i=1}^N I_F^i = 1$.

3.1. Focus

Let p be a distribution of N elements with p_i denoting the probability of each element. We remind that $p_i \geq 0, \forall i$, and $\sum_{i=1}^N p_i = 1$. Given some probability distribution p, a general measure for uniformity of that distribution is entropy,

$$H(p) := -\sum_{i=1}^{N} p_i \ln(p_i). \tag{1}$$

It is a non-negative function, H is low when the distribution has sharp peaks and its value increases as the distribution becomes more even. To obtain a normalized measure, in the

Model	Approach	WolfMix[18, 20]		Un-Augmented	
		Clean↑	mCE↓	Clean↑	mCE↓
DGCNN[44]	Vanilla	93.2%	0.590	92.6%	1.000
	EPiC [22]	92.1%	0.529	93.0%	0.669
	PointGuard [25]	81.9%	1.154	83.8%	1.165
	Refocusing **(Ours)**	91.4%	0.560	91.6%	0.688
	EPiC & Refocusing **(Ours)**	92.9%	**0.484**	93.4%	**0.557**
RPC[35]	Vanilla	93.3%	0.601	93.0%	0.863
	EPiC [22]	92.7%	0.501	93.6%	0.750
	PointGuard [25]	83.2%	1.067	86.9%	1.051
	Refocusing **(Ours)**	91.2%	0.562	91.6%	0.728
	EPiC & Refocusing **(Ours)**	92.9%	**0.476**	93.2%	**0.616**
GDANet[52]	Vanilla	93.4%	0.571	93.4%	0.892
	EPiC [22]	92.5%	0.530	93.6%	0.704
	PointGuard [25]	83.2%	1.059	84.8%	1.132
	Refocusing **(Ours)**	91.8%	0.528	91.4%	0.718
	EPiC & Refocusing **(Ours)**	92.8%	**0.493**	93.4%	**0.587**

Table 2. **Comparison on ModelNet-C [35], WolfMix augmented and augmented free.** Our approach is on-par with EPiC with extremely faster inference time (see Fig. 7). Combining our approach as extra sampling strategy in EPiC based on RPC achieves SOTA results in terms of robustness. Lower mean Corruption Error (mCE), values and on-par performance on uncorrupted (clean) set, ModelNet40 [47].

range $[0, 1]$, we use normalized entropy. This measure divides the entropy by the maximum possible entropy for that sample: $H_n := H/H_{max}$, where $H_{max} = \max_p\{H(p)\}$. It is well known that entropy is maximized for the uniform distribution, $p_i = 1/N, \forall i$. Plugging this in H_{max} and rearranging yields the following expression for the normalized entropy [46],

$$H_n(p) = \frac{H(p)}{\ln(N)}. \tag{2}$$

We can now define the focus of a network.

Definition 3.1 (Focus of a network) *Given a network F, an input X and an associated normalized influence measure $\hat{I}_F(X)$, the focus of the network, denoted f, is defined by*

$$f(X) := 1 - H_n(\hat{I}_F(X)). \tag{3}$$

Let us state some basic properties of $f(X)$.

Proposition 3.1 (Focus properties) *For any network F, input X of any size N and normalized influence $I_F(X)$, the focus $f(X)$ has the following properties:*

1. $f(X) \in [0, 1]$.
2. $f(X) = 1$ *iff* $\exists i$, $I^i_F = 1$, $I^j_F = 0$, $\forall j \neq i$.
3. $f(X) = 0$ *iff* $I^i_F = \frac{1}{N}$, $\forall i$.

The proof follows directly from the properties of normalized entropy. Consequently, we obtain a general definition of focus within the range $[0, 1]$, enabling easy comparison across different network settings and input sizes. To provide a better intuition for the extremities of the focus measure distributed on a 3D shape, we visualize samples from ModelNet40 [47] with high and low focus values in Fig. 2.

3.1.1 Why normalized entropy?

As mentioned in [46],"To obtain a measure of uncertainty that can be compared across distributions, actual uncertainty must be divided by the maximum possible uncertainty." In this paper, our primary focus is on 3D classification; however, we formulate the definition in a broader context, such that it can be extended to other domains. For an entropy measure, the following relation holds:

$$-\sum_{1}^{k_1} \frac{1}{k_1} \ln\left(\frac{1}{k_1}\right) > -\sum_{1}^{k_2} \frac{1}{k_2} \ln\left(\frac{1}{k_2}\right).$$

For all $k_1 > k_2 > 0$. This implies that the maximal (not normalized) entropy of a vector with more elements is higher. In 3D analysis, when the influence measure is evenly distributed, a larger point cloud has larger entropy. We would like a measure which is invariant to the point cloud size. Hence, normalized entropy is employed.

3.2. Focus distribution, over- and under-focusing

We would like to analyze the network's focus behavior under different datasets, introducing additional notions. Let $T \in \mathbb{R}^{M_T \times N \times d}$ represent a training set with M_T data instances, and $S \in \mathbb{R}^{M_S \times N \times d}$ denote a test set with M_S data

instances. Random samples from these sets are denoted as X^T and X^S, respectively. The empirical mean is denoted by $E[\cdot]$. For a given set Q, let $\mu_Q = E[f(X^Q)]$ and $\sigma_Q = \sqrt{E[(f(X^Q) - \mu_Q)^2]}$.

Algorithm 1 Refocusing (Inference)

Require: $X, params_{refocused}$
$model_{refocused} \leftarrow params_{refocused}$ ▷ Load Pretrained
$X_f = model_{refocused}(X)$ ▷ First forward-pass
$I(X) = Eq.(4)$
$\hat{I} = \frac{I}{\Sigma_{i=1}^{N}(I)}$
$f = Eq.(3)$ ▷ Calculate f
$K = \lfloor (1-f) \cdot N \rfloor$ ▷ Adaptive Threshold
▷ Select K lowest influential points
$X_{sampled} = SelectLowest(X, \hat{I}, K)$
$P = model_{refocused}(X_{sampled})$ ▷ Second forward-pass
$Class = \arg\max(P)$

We define an over-focused sample as X^S such that $f(X^S) \geq \mu_T + \alpha \cdot \sigma_T$ and an under-focused sample as X^S such that $f(X^S) \leq \mu_T - \beta \cdot \sigma_T$, where α and β are tunable parameters corresponding to T. Practically, this approach allows us to identify OOD regions, where samples significantly deviate from the learned focus distribution. Detecting such regions is crucial, as performance degradation is associated with data residing outside the training distribution (See Fig. 6). By incorporating prior knowledge derived from the statistics obtained during training, we gain valuable insights into potential challenges the model may encounter when faced with unfamiliar data at inference.

3.3. Refocusing

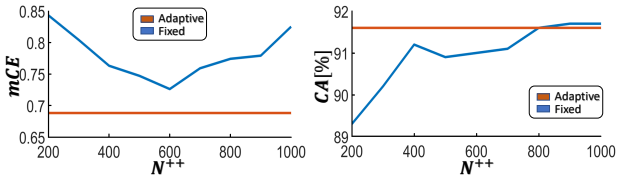
Our method relies on filtering the most influential points identified by a given influence map, as outlined in Algorithm 1. To achieve this, we seek an influence evaluation that is computationally efficient, given its integration during inference. The literature on XAI for point-cloud classification broadly falls into three categories: 1) Iterative processes [38, 62]; 2) Dedicated explainable architectures [1, 60]; 3) Utilizing gradients [11, 27]. However, these approaches either consume significant computational time or lack the capacity to explain certain architectures. Consequently, we choose an explainable method that is a variation of [55]. In essence, the influence measure prioritizes importance based on the frequency of appearance in the global feature vector. Specifically, it quantifies the count of features with the highest values compared to all other points. The influence measure is defined as:

$$I_F(X(j,\cdot)) = \sum_{k=1}^{\mathcal{K}} \mathbb{I}(j == \arg\max_n(X_f(n,k))), \quad (4)$$

where $\mathbb{I}$ is an indicator function (equal to 1 when true and 0 otherwise), $X_f \in \mathbb{R}^{N \times \mathcal{K}}$ is a matrix where each row corresponds to a per-point learned feature, and $\mathcal{K}$ is the size of the feature vector. The normalized influence is thus $\hat{I}_F(X(j,\cdot)) = \frac{I_F(X(j,\cdot))}{\sum_{i=1}^{N} I_F(X(i,\cdot))}$.

3.3.1 Refocusing - "Reign of the less influential"

In Fig. 6 the success rate is shown as a function of focus. We see typical narrow range for the clean set and a much wider range for the corrupted sets. Based on our observation that outliers strongly affect the influence map, it is intuitive to diminish corruption byproducts by discarding the most influential points. Thus, the influence is redistributed among the remaining points. This action should filter out corruptions and align the focus closer to the narrow region of the clean set. Introduction or removal of points shifts the focus distribution from the distribution learned during training. Outliers cause over-focusing, whereas missing points yield under-focusing. However, after cropping most influential points, the focus distribution is aligned, as can be seen in Fig. 4. Therefore, we term our process as *refocusing*.



(a) **Mean Corruption Error (mCE) Vs. Number of retained points.**

(b) **Clean Accuracy Vs. Number of retained points.**

Figure 5. **Adaptive vs. fixed threshold.** Our adaptive threshold demonstrates superior robustness compared to any fixed threshold. The adaptive, parameter-free threshold yields a single result, represented as a straight (orange) line in the plot.

3.3.2 Adaptive threshold

We argue that samples containing outliers should be subjected to more aggressive filtering, compared to samples that have missing points. In fact, it is not clear whether the latter case should be sampled at all. This raises the general question: *How many points should be retained?* In information theory, *normalized entropy* [46], also referred to as *efficiency* can resolve this. One can think of maximal entropy as the most efficient representation, where normalized entropy is a measure of relative efficiency. In the context of 3D classification, setting equal contribution for any input point is equivalent to the most efficient representation. Thus, we advocate using normalized entropy as a criterion for determining the ratio of remaining points during the filtering process. We set the remaining number of points to

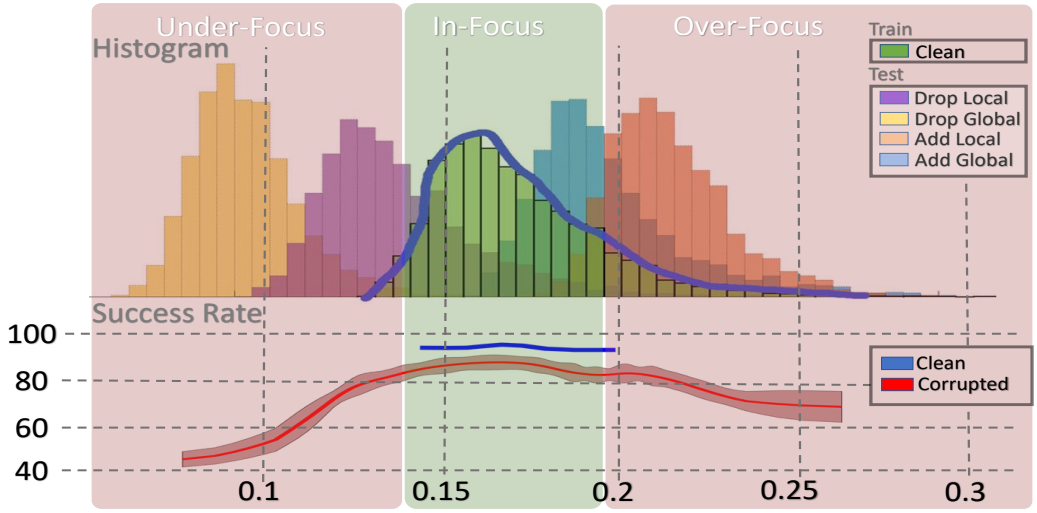


Figure 6. **In-focus, Under-focus, and Over-focus.** Top - Histogram of focus values for the clean set, ModelNet40 [47], defining the in-focus region inside the standard deviation. Histograms collected from corrupted sets in ModelNet-C [35] are clearly out of the training distribution. The trend indicates that the appearance of outliers correlates with over-focusing, while the absence of points correlates with under-focusing. Bottom - Success rate of clean (blue) and corrupted (red) sets. A clear performance drop is observed in the over-focus and under-focus regions.

$$N^{++} = \lfloor (1-f) \cdot N \rfloor.$$

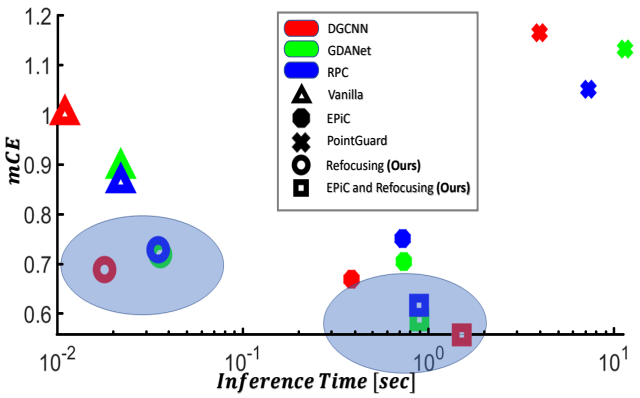


Figure 7. **Robustness vs. inference time (log-scale).** Our method is as fast as vanilla networks, with outstanding robustification, on-par with EPiC [22]. Combining our approach as extra sampling strategy achieves SOTA mCE (both highlighted in blue).

We evaluate various fixed values of remaining points, compare them to our adaptive threshold, and plot the accuracy on uncorrupted samples and robustness to corrupted ones in a classification task (see Fig. 5). The results strongly support our proposal of employing an adaptive threshold rather than a fixed one. The utilization of our suggested selective filtering approach significantly improves the mean corruption error (mCE) compared to any fixed number of sampling points. Moreover, it preserves good performance in terms of success rate on clean samples. Our refocusing algorithm by sampling is summarized in Algorithm 1 for inference; the training procedure is detailed in the supplementary.

4. Applications and Experiments

We now show how our proposed refocusing method can be used for robust classification and adversarial defense.

4.1. Robust classification

Benchmark. ModelNet-C [35] is a variation of ModelNet-40 [47] designed to assess robustness to out-of-distribution (OOD) data. It introduces seven types of corruptions (jitter, scale, rotate, add-global, add-local, drop-global, and drop-local), each with five difficulty levels. A unified calculation mechanism, referred to as *mean Corruption Error (mCE)*, is used to measure robustness. Lower mCE scores indicate better performance. Please refer to [35] for more details.

During training, a point-cloud classification network is trained on the clean set only, adapted to accept a wide range of sampled points (256-1024). The same basic network is used for querying the influence map and for the actual prediction on the filtered sample. Thus, this process can be thought of as a self-restraining process. The network provides a mapping of the influential inputs. After refocusing, the same network is used for classification, based on the least influential points. More details and a pseudocode of the training procedure appear in the supplementary. Inference procedure is described in details in Algorithm 1. It includes dual forward-pass, and simple and fast extra cal-

Defense	DGCNN [44]		PointNet[32]		GDANet[52]	
	ASR↓	AQ↑	ASR↓	AQ↑	ASR↓	AQ↑
Undefended	99.3	106.7	99.8	18.9	99.8	18.9
SRS(50%) [64]	78.4	566.3	94.0	190.9	78.1	595.4
SRS(30%) [64]	68.6	790.3	97.6	93.5	72.4	714.0
SOR [36]	75.6	795.6	78.4	592.9	69.9	913.2
LPF-Defense[12]	47.8	1148.0	98.2	123.1	52.6	1071.4
Refocusing **(Ours)**	**37.5**	**1376.18**	**72.0**	**730.4**	**34.6**	**1425.5**

Table 3. **Adversarial defenses from shape-invariant attack [15] on ModelNet40 [47].** Attack success rate (ASR, measured in percents) is consistently the lowest and mean query cost (AQ, measured in average time) is the highest, over all examined networks, compared to all other defense methods. Note that for DGCNN and GDANet ASR is extremely decreased.

culations. To demonstrate the efficiency of the proposed method, Fig. 7 depicts a comparison to other robust networks, by plotting mCE vs. inference time. We train three different networks with refocusing, showing a substantial improvement in robustness (lower mCE), while being competitive in terms of accuracy on clean samples. To further mitigate accuracy on clean samples, and for even better performance against corruptions, we used refocusing as an extra global sampling strategy, such that combining with EPiC [22] ensemble method, yields SOTA results on this dataset (technical explanation of embedding refocus in EPiC are provided in the supplementary).

4.2. Adversarial defense

Another major threat for point cloud classification networks is adversarial attacks. It has been shown that main classification networks are vulnerable for this attacks, even for barely distinguishable perturbed clouds [15, 62]. In these cases, the classification network has no knowledge regarding the manipulation, thus, there is a clear advantage for OOD robust approaches. We applied our method (as described in Alg. 1) as a defense scheme against Shape-Invariant Attack [15] and evaluated it against several OOD defenses. Our evaluation includes a comparison with an undefended backbone model, along with the trivial defense of Simple Random Sampling (SRS), which randomly removes 30% and 50% of input points. Additionally, we assess more sophisticated defenses such as Statistical Outlier Removal (SOR) [36] and LPF-Defense [12]. Our method achieves substantial improvements, reducing the *attack success rate* (ASR) to a limited 37.5% (compare to 47.8% using LPF-Defense [12]) when embedded to DGCNN [44]. We examine the case where DGCNN is functioning as both the surrogate model and the attacked model, to eliminate transferability issues. The results are shown in Tab. 3.

5. Discussion and Conclusion

The analysis of focus introduces a deeper understanding of neural network performance. It is intriguing to measure over- and under-focus characteristics in various domains, including NLP, audio, and image processing. This understanding can pave the way for a wide range of applications. In the supplementary material, we provide a very preliminary example of how facilitating refocusing or parts of the algorithm can aid in outliers removal. The idea can further extend, for instance, for developing over- or under-focus adversarial attacks, yielding specific focus values. Another potential path is guided adversarial training, which extends the focus range to the one exposed during training.

In this study, we introduced a novel perspective on point cloud neural network behavior through the analysis of focus. We proposed a definition for a network's focus, over-focus, and under-focus, which can be extended beyond 3D point clouds. We observed a strong correlation between corruptions and focus distribution. The presence of outliers predominantly increases focus, while occluded parts have the opposite effect. To enhance the network's ability to process corrupted data, we proposed a robust algorithm aimed at screening out the most influential input elements. Filtering mitigates the impact of outliers and aligns the focus distribution, resulting in improved robustness against OOD corruptions, with only a marginal degradation in accuracy for clean data.

Our method is computationally efficient, making it applicable to time-demanding applications. Experimental results on robust classification and adversarial defense tasks showcase the effectiveness of our approach. We achieved state-of-the-art results for both robust zero-shot classification on the ModelNet-C [35] dataset and for adversarial defense against Shape-Invariant attacks [15].

Acknowledgment: We acknowledge support by the Israel Science Foundation (Grants No. 1472/23), by the Ministry of Science and Technology (Grant No. 5074/22) and by the Ollendorff Minerva Center.

References

[1] Arnold, N.I., Angelov, P., Atkinson, P.M.: An improved explainable point cloud classifier (xpcc). IEEE Transactions on Artificial Intelligence **4**(1), 71–80 (2022) 6

[2] Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), e0130140 (2015) 1

[3] Chefer, H., Gur, S., Wolf, L.: Transformer interpretability beyond attention visualization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 782–791 (2021) 1

[4] Chen, X., Ma, H., Wan, J., Li, B., Xia, T.: Multi-view 3d object detection network for autonomous driving. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1907–1915 (2017) 4

[5] Cordonnier, J.B., Loukas, A., Jaggi, M.: On the relationship between self-attention and convolutional layers. arXiv preprint arXiv:1911.03584 (2019) 3

[6] Dovrat, O., Lang, I., Avidan, S.: Learning to sample. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2760–2769 (2019) 4

[7] Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. Nature Machine Intelligence **2**(11), 665–673 (2020) 1

[8] Ghader, H., Monz, C.: What does attention in neural machine translation pay attention to? arXiv preprint arXiv:1710.03348 (2017) 1, 3

[9] Guo, M.H., Cai, J.X., Liu, Z.N., Mu, T.J., Martin, R.R., Hu, S.M.: Pct: Point cloud transformer. Computational Visual Media **7**, 187–199 (2021) 4

[10] Guo, Y., Stutz, D., Schiele, B.: Robustifying token attention for vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17557–17568 (2023) 1, 3

[11] Gupta, A., Watson, S., Yin, H.: 3d point cloud feature explanations using gradient-based methods. In: 2020 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2020) 6

[12] H., N., K., N., A., E., S., K.: Lpf-defense: 3d adversarial defense based on frequency analysis. Plos one, 18(2), e0271388. (2023) 4, 8

[13] Hamdi, A., Rojas, S., Thabet, A., Ghanem, B.: Advpc: Transferable adversarial perturbations on 3d point clouds. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16. pp. 241–257. Springer (2020) 4

[14] Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D.: Natural adversarial examples. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15262–15271 (2021) 1

[15] Huang, Q., Dong, X., Chen, D., Zhou, H., Zhang, W., Yu, N.: Shape-invariant 3d adversarial point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15335–15344 (2022) 4, 8

[16] J., H., J., Y., C., Q., Y., A., C., L., C., B.: Pointacl: Adversarial contrastive learning for robust point clouds representation under adversarial attack. In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1-5). IEEE. (2023) 4

[17] Kim, J., Hua, B.S., Nguyen, T., Yeung, S.K.: Minimal adversarial examples for deep learning on 3d point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7797–7806 (2021) 4

[18] Kim, S., Lee, S., Hwang, D., Lee, J., Hwang, S.J., Kim, H.J.: Point cloud augmentation with weighted local transformations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 548–557 (2021) 4, 5

[19] Lang, I., Manor, A., Avidan, S.: Samplenet: Differentiable point cloud sampling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7578–7588 (2020) 4

[20] Lee, D., Lee, J., Lee, J., Lee, H., Lee, M., Woo, S., Lee, S.: Regularization strategy for point cloud via rigidly mixed sample. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15900–15909 (2021) 4, 5

[21] Lee, K., Chen, Z., Yan, X., Urtasun, R., Yumer, E.: Shapeadv: Generating shape-aware adversarial 3d point clouds. arXiv preprint arXiv:2005.11626 (2020) 4

[22] Levi, M.Y., Gilboa, G.: Epic: Ensemble of partial point clouds for robust classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 14475–14484 (October 2023) 3, 4, 5, 7, 8

[23] Liu, D., Yu, R., Su, H.: Extending adversarial attacks and defenses to deep 3d point cloud classifiers. In: 2019 IEEE International Conference on Image Processing (ICIP). pp. 2279–2283. IEEE (2019) 4

[24] Liu, D., Yu, R., Su, H.: Adversarial shape perturbations on 3d point clouds. In: Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16. pp. 88–104. Springer (2020) 4

[25] Liu, H., Jia, J., Gong, N.Z.: Pointguard: Provably robust 3d point cloud classification. In: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition. pp. 6186–6195 (2021) 3, 4, 5

[26] Ma, X., Qin, C., You, H., Ran, H., Fu, Y.: Rethinking network design and local geometry in point cloud: A simple residual mlp framework. arXiv preprint arXiv:2202.07123 (2022) 4

[27] Matrone, F., Paolanti, M., Felicetti, A., Martini, M., Pierdicca, R.: Bubblex: An explainable deep learning framework for point-cloud classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **15**, 6571–6587 (2022) 6

[28] Mesika, A., Ben-Shabat, Y., Tal, A.: Cloudwalker: Random walks for 3d point cloud shape analysis. Computers & Graphics **106**, 110–118 (2022) 4

[29] Pan, Z., Zhuang, B., He, H., Liu, J., Cai, J.: Less is more: Pay less attention in vision transformers. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 2035–2043 (2022) 3

[30] Park, B., Choi, J.: Explanation on pretraining bias of finetuned vision transformer. arXiv preprint arXiv:2211.15428 (2022) 4

[31] Q., L., Q., L., W., N., A., L.A.: Pagn: perturbation adaption generation network for point cloud adversarial defense. Multimedia Systems, 28(3), 851-859. (2022) 4

[32] Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017) 4, 8

[33] Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017) 4

[34] Rakotosaona, M.J., La Barbera, V., Guerrero, P., Mitra, N.J., Ovsjanikov, M.: Pointcleannet: Learning to denoise and remove outliers from dense point clouds. In: Computer graphics forum. vol. 39, pp. 185–203. Wiley Online Library (2020) 4

[35] Ren, J., Pan, L., Liu, Z.: Benchmarking and analyzing point cloud classification under corruptions. In: International Conference on Machine Learning. pp. 18559–18575. PMLR (2022) 2, 3, 4, 5, 7, 8

[36] Rusu, R.B., Marton, Z.C., Blodow, N., Dolha, M., Beetz, M.: Towards 3d point cloud based object maps for household environments. Robotics and Autonomous Systems **56**(11), 927–941 (2008) 4, 8

[37] Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017) 1

[38] Tan, H., Kotthaus, H.: Surrogate model-based explainability methods for point cloud nns. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2239–2248 (2022) 6

[39] Tan, H., Kotthaus, H.: Explainability-aware one point attack for point cloud neural networks. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 4581–4590 (2023) 4

[40] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., McDaniel, P.: Ensemble adversarial training: Attacks and defenses. arXiv preprint arXiv:1705.07204 (2017) 4

[41] Tsai, T., Yang, K., Ho, T.Y., Jin, Y.: Robust adversarial objects against deep learning models. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 954–962 (2020) 4

[42] Varley, J., DeChant, C., Richardson, A., Ruales, J., Allen, P.: Shape completion enabled robotic grasping. In: 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 2442–2447. IEEE (2017) 4

[43] Vig, J., Belinkov, Y.: Analyzing the structure of attention in a transformer language model. arXiv preprint arXiv:1906.04284 (2019) 4

[44] Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (tog) **38**(5), 1–12 (2019) 2, 3, 4, 5, 8

[45] Wicker, M., Kwiatkowska, M.: Robustness of 3d deep learning in an adversarial setting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11767–11775 (2019) 4

[46] Wilcox, A.R.: Indices of qualitative variation. Tech. rep., Oak Ridge National Lab., Tenn. (1967) 5, 6

[47] Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015) 3, 5, 7, 8

[48] Wu, Z., Duan, Y., Wang, H., Fan, Q., Guibas, L.J.: If-defense: 3d adversarial point cloud defense via implicit function based restoration. arXiv preprint arXiv:2010.05272 (2020) 4

[49] Xiang, C., Qi, C.R., Li, B.: Generating 3d adversarial point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9136–9144 (2019) 4

[50] Xiang, T., Zhang, C., Song, Y., Yu, J., Cai, W.: Walk in the cloud: Learning curves for point clouds shape analysis. In: Proceedings of the IEEE/CVF Interna-

tional Conference on Computer Vision. pp. 915–924 (2021) 4

[51] Xu, M., Ding, R., Zhao, H., Qi, X.: Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3173–3182 (2021)

[52] Xu, M., Zhang, J., Zhou, Z., Xu, M., Qi, X., Qiao, Y.: Learning geometry-disentangled representation for complementary understanding of 3d object point cloud. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 3056–3064 (2021) 2, 4, 5, 8

[53] Yan, X., Zheng, C., Li, Z., Wang, S., Cui, S.: Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5589–5598 (2020) 4

[54] Yang, J., Zhang, Q., Fang, R., Ni, B., Liu, J., Tian, Q.: Adversarial attack and defense on point sets. arXiv preprint arXiv:1902.10899 (2019) 4

[55] Yang, P., Snoek, C.G., Asano, Y.M.: Self-ordering point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15813–15822 (2023) 4, 6

[56] Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., Lu, J.: Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19313–19322 (2022) 4

[57] Yue, X., Wu, B., Seshia, S.A., Keutzer, K., Sangiovanni-Vincentelli, A.L.: A lidar point cloud generator: from a virtual world to autonomous driving. In: Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval. pp. 458–464 (2018) 4

[58] Zhang, J., Jia, J., Liu, H., Gong, N.Z.: Pointcert: Point cloud classification with deterministic certified robustness guarantees. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9496–9505 (2023) 4

[59] Zhang, J., Chen, L., Ouyang, B., Liu, B., Zhu, J., Chen, Y., Meng, Y., Wu, D.: Pointcutmix: Regularization strategy for point cloud classification. Neurocomputing **505**, 58–67 (2022) 4

[60] Zhang, M., You, H., Kadam, P., Liu, S., Kuo, C.C.J.: Pointhop: An explainable machine learning method for point cloud classification. IEEE Transactions on Multimedia **22**(7), 1744–1755 (2020) 6

[61] Zhao, Y., Wu, Y., Chen, C., Lim, A.: On isometry robustness of deep 3d point cloud models under adversarial attacks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1201–1210 (2020) 4

[62] Zheng, T., Chen, C., Yuan, J., Li, B., Ren, K.: Pointcloud saliency maps. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1598–1606 (2019) 4, 6, 8

[63] Zhou, H., Chen, D., Liao, J., Chen, K., Dong, X., Liu, K., Zhang, W., Hua, G., Yu, N.: Lg-gan: Label guided adversarial network for flexible targeted attack of point cloud based deep networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10356–10365 (2020) 4

[64] Zhou, H., Chen, K., Zhang, W., Fang, H., Zhou, W., Yu, N.: Dup-net: Denoiser and upsampler network for 3d adversarial point clouds defense. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1961–1970 (2019) 4, 8