ART-Point: Improving Rotation Robustness of Point Cloud Classifiers via Adversarial Rotation

Ruibin Wang[1], Yibo Yang[2,*], Dacheng Tao[2]
[1] Key Laboratory of Machine Perception (MOE), School of Artificial Intelligence, Peking University
[2] JD Explore Academy, China
{robin_wang, ibo}@pku.edu.cn, dacheng.tao@gmail.com

Abstract

Point cloud classifiers with rotation robustness have been widely discussed in the 3D deep learning community. Most proposed methods either use rotation invariant descriptors as inputs or try to design rotation equivariant networks. However, robust models generated by these methods have limited performance under clean aligned datasets due to modifications on the original classifiers or input space. In this study, for the first time, we show that the rotation robustness of point cloud classifiers can also be acquired via adversarial training with better performance on both rotated and clean datasets. Specifically, our proposed framework named ART-Point regards the rotation of the point cloud as an attack and improves rotation robustness by training the classifier on inputs with Adversarial RoTations. We contribute an axis-wise rotation attack that uses back-propagated gradients of the pre-trained model to effectively find the adversarial rotations. To avoid model over-fitting on adversarial inputs, we construct rotation pools that leverage the transferability of adversarial rotations among samples to increase the diversity of training data. Moreover, we propose a fast one-step optimization to efficiently reach the final robust model. Experiments show that our proposed rotation attack achieves a high success rate and ART-Point can be used on most existing classifiers to improve the rotation robustness while showing better performance on clean datasets than state-of-the-art methods.

1. Introduction

A very basic requirement for point cloud classification is expecting the network to obtain stable predictions on inputs undergoing rigid transformations since such transformations do not change the shape of the object, let alone change its semantic meanings. This basic requirement is even more important in practical applications. For example, when a robot is identifying and picking up an object, the object is usually in an unknown pose. However, many studies [7, 17, 51] have shown that most existing point cloud classifiers can be easily attacked by simply rotating the inputs. To use these classifiers we require to align all input objects which is a very expensive and time-consuming process. To this end, how to improve the robustness of point cloud classifiers to arbitrary rotations, becomes a very popular and necessary research topic.

In order to make the network robust to rotated inputs, most existing works can be classified into three categories: (1) **Rotation Augmentation Methods** attempt to augment the training data using rotations and have been widely used in the earlier point cloud classifiers [29, 30, 39]. However, data augmentation can hardly be applied to improve model robustness to arbitrary rotations due to the astronomical number of rotated data [49]. (2) **Rotation-Invariance Methods** propose to convert the input point clouds into geometric descriptors that are invariant to rotations. Typical invariant descriptors can be the distance and angles between local point pairs [4, 8, 47, 48] or point norms [17, 49] and principal directions [47] calculated from global coordinates. (3) **Rotation-Equivariance Methods** try to solve the rotation problem from the perspective of model architectures. For example, [5, 27, 37, 40] use convolution with steerable kernel bases to construct rotation-equivariant networks and [7, 34, 50] modify existing networks with equivariant operations. While both methods (2) and (3) can effectively improve model robustness to arbitrary rotations, they either require time-consuming pre-processing on inputs or need complex architectural modifications, which will result in limited performance on clean aligned datasets.

In this paper, we try to explore a new technical route for the rotation robustness problem in point clouds. Our method is inspired by adversarial training [22], a typical defense method to improve model robustness to attacks. The idea of adversarial training is straightforward: it augments training data with adversarial examples in each training loop. Thus adversarially trained models behave more

*Corresponding author.

normally when facing adversarial examples than standardly trained models. Adversarial training has shown its great effectiveness in improving model robustness to image or text perturbations [9, 11, 21, 33, 44], while keeping a strong discriminative ability. In 3D point clouds, [18, 35, 36] also successfully leverage adversarial training to defend against point cloud perturbations such as random point shifting or removing. However, using adversarial training to improve the rotation robustness of point cloud classifiers has rarely been studied.

To this end, by regarding rotation as an attack, we develop the ART-Point framework to improve the rotation robustness by training networks on inputs with **A**dversarial **R**o**T**ations. Like the general framework of adversarial training, ART-Point forms a classic min-max problem, where the max step finds the most aggressive rotations, on which the min step is performed to optimize the network parameters for rotation robustness. For the max step, we propose an axis-wise rotation attack algorithm to find the most offensive rotating samples. Compared with the existing rotation attack algorithm [51] that directly optimizes the transformation matrix, our method optimizes on the rotation angles which reduces the optimization parameters, while ensuring that the attack is pure rotation to serve for the adversarial training. For the min step, we follow the training scheme of the original classifier to retrain the network on the adversarial samples. To overcome the problem of over-fitting on adversarial samples caused by label leaking [15], we construct a rotation pool that leverages the transferability of adversarial rotations among point cloud samples to increase the diversity of training data. Finally, inspired by ensemble adversarial training [38], we contribute a fast one-step optimization method to solve the min-max problems. Instead of alternately optimizing the min-max problem until the model converges, the one-step method can quickly reach the final robust model with competitive performance.

Compared with the rotation-invariant and equivariant methods, the ART-Point framework aims to optimize network parameters such that the converged model is naturally robust to both arbitrary and adversarial rotations, without the necessity of either geometric descriptor extractions or architectural modifications that may impede the model to learn discriminative features. So our resulting robust model better inherits the original performance on the clean (aligned) datasets. It has no constraint on the model design and can be integrated on most point cloud classifiers.

In experiments, we mainly verify the effectiveness of our methods under two datasets ModelNet40 [42] and ShapeNet16 [46]. We adopt PointNet [29], PointNet++ [30] and DGCNN [39] as the basic classifiers. Firstly, compared with the existing rotation attack method [51], our proposed attack achieves a higher attack success rate. Then, compared with existing rotation robust classifiers, our best model (ART-DGCNN) shows a more robust performance on randomly rotated datasets. Meanwhile, our methods generally show less accuracy reduction on clean aligned datasets. Beyond arbitrary rotations, the resulting models also show a solid defense against adversarial rotations.[1] Our contributions can be summarized as follows:

- For the first time, we successfully improve the rotation robustness of point cloud classifiers from the perspective of model attack and defense. Our proposed framework, ART-Point, enjoys fewer architectural modifications than previous rotation-equivariant methods and requires no descriptor extractions on input data.
- We propose an axis-wise rotation attack algorithm to efficiently find the most aggressive rotated samples for adversarial training. A rotation pool is designed to avoid over-fitting of models on adversarial samples. We also contribute a fast one-step optimization to solve the min-max problem.
- We validate our method on two datasets with three point cloud classifiers. The results show that our attack algorithm achieves a higher attack success rate than existing methods. Moreover, the proposed ART-Point framework can effectively improve model rotation robustness allowing the model to defend against both arbitrary and adversarial rotations, while hardly affecting model performance on clean data.

2. Related Work

2.1. Rotation Robust Point Cloud Classifiers

Rotation Augmentation. The initial work of the point cloud classifier [29, 30, 39] adopt rotation augmentation during training to improve rotation robustness. Nevertheless, rotation augmentation can only result in models robust to a small range of angles. More recently, to obtain models robust to arbitrary rotation angles, both rotation-invariance and rotation-equivariance methods are proposed.

Rotation-invariance methods extract rotation-invariant descriptors from point clouds as model inputs. For example, [4, 8, 28, 48] cleverly construct distances and angles from local point pairs. [17, 47, 49] further extend local invariant descriptors with global invariant contexts. In addition to using invariant descriptors with a clear geometric meaning, [20, 28, 31] also design invariant convolutions to automatically learn various descriptors for processing.

Rotation-equivariance methods expect the learned features to rotate correspondingly with the input thus resulting in rotation robust models. Most of these works usually rely on rotation-equivariant convolutions [5, 6, 10, 14, 27, 37, 40]

[1] Code address: https://github.com/robinwang1/ART-Point.

to construct equivariant networks. Other works like [7, 34, 50] attempt to modify modules in existing point cloud classifiers [29, 30, 39] to make them rotation-equivariant.

However, these methods usually require specific descriptors or network modules which will reduce the performance of the classifier on the aligned datasets. Our study differs from these methods in that we try to obtain a robust model by optimizing the parameters without changing the input space or network architectures.

2.2. Adversarial Training

Adversarial Training [13, 22] has been proved to be the most effective technique against adversarial attacks [23, 26, 32], receiving considerable attention from the research community. Unlike other defense strategies, adversarial training aims to enhance the robustness of models intrinsically [1]. This property makes adversarial training widely used in various fields to improve the robustness of the model, including image recognition [11, 12, 33, 44], text classification [9, 21, 24, 25], relation extraction [41] etc. In 3D point clouds classification, adversarial training can also be effectively used. For example, [18] employs adversarial training to improve the model robustness to point shifting perturbation by training on both clean and adversarially perturbed point clouds. [36] presents an in-depth study showing how adversarial training behaves in point cloud classification. However, existing works only focus on improving the model's robustness to perturbations of random point shifting or removing [12, 16, 19, 43, 45, 52].

Recently, [51] designs a rotation attack algorithm for existing point cloud classifiers. Yet it does not provide detailed strategies to defense the rotation attack. As a comparison, we design a new attack algorithm that enjoys a higher attack success rate. More importantly, it serves for our adversarial training framework that generates model naturally defending against both arbitrary and adversarial rotations.

3. Methods

In this section, we first provide a brief review of adversarial training (Sect. 3.1). Then, we reformulate the adversarial training objective under rotation attack of point clouds (Sect. 3.2). Next, we propose attack (Sect. 3.3) and defense (Sect. 3.4) algorithms to obtain good solutions to the reformulated objective. Finally, we provide a one-step optimization to fast reach a robust model (Sect. 3.5).

3.1. Preliminaries on Adversarial Training

Let us first consider a standard classification task with an underlying data distribution $\mathcal{D}$ over inputs $p \in \mathbb{R}^d$ and corresponding labels $q \in [k]$. The goal then is to find model parameters θ that minimize the risk $\mathbb{E}_{(p,q)\sim\mathcal{D}}[L(\theta, p, q)]$, where $L(\theta, p, q)$ is a suitable loss function. To improve the model robustness, we wish no perturbations are possible to fool the network, which gives rise to the following formulation:

$$\min_{\theta} \rho(\theta), \quad \text{where} \quad \rho(\theta) = \mathbb{E}_{(p,q)\sim\mathcal{D}}[L(\theta, p+\delta, q)], \tag{1}$$

where $p+\delta$ refers to the perturbed samples generated by introducing perturbations $\delta \in \mathcal{S}$ on input data p. $\mathcal{S}$ refers to the allowed perturbation set. Eq. (1) reflects the basic idea of data augmentations.

In contrast, adversarial training improves model robustness more efficiently. By the in-depth study of the landscape of adversarial samples, [22] finds the concentration phenomenon of different adversarial samples, which suggests that training on the most aggressive adversary yields robustness against all other concentrated adversaries. This gives rise to the formulation of adversarial training which is a saddle point problem:

$$\min_{\theta} \rho(\theta), \quad \text{where} \quad \rho(\theta) = \mathbb{E}_{(p,q)\sim\mathcal{D}}[\max_{\delta\in\mathcal{S}} L(\theta, p+\delta, q)]. \tag{2}$$

The saddle point problem can be viewed as the composition of an inner maximization problem and an outer minimization problem, where the inner maximization problem is finding the worst-case samples for the given model, and the outer minimization problem is to train a model robust to adversarial samples. Compared with data augmentation, adversarial training searches for the best solution to the worst-case optimum and can improve the model robustness to perturbations in larger ranges [22].

3.2. Problem Formulation

Our main goal is to improve the robustness of the point cloud classifiers to rotation attacks through the adversarial training framework. We reformulate Eq. (2) by specifying the perturbation to be the point cloud rotation as follows:

$$\min_{\theta} \rho(\theta), \quad \text{where} \quad \rho(\theta) = \mathbb{E}_{(p,q)\sim\mathcal{D}}[\max_{R\in SO(3)} L(\theta, Rp, q)], \tag{3}$$

where $p \in \mathbb{R}^{n\times 3}$ refers to an input point cloud of size n and $q \in [k]$ is the corresponding class label. θ is the parameters of point cloud classifiers such as PointNet [29] or DGCNN [39]. Rp refers to the adversarial samples generated by using matrix R to rotate the input p and $SO(3)$ is the group of all rotations around the origin of $\mathbb{R}^3$ Euclidean space. We set the rotation $R \in SO(3)$ to ensure the objective is to make the model robust to arbitrary rotations.

As discussed in [22], one key element for obtaining a good solution to Eq. (3) is using the strongest possible adversarial samples to train the networks. Following this principle, we first propose a novel rotation attack method that enjoys satisfactory attack success and thus better serves for the adversarial training to improve model robustness.

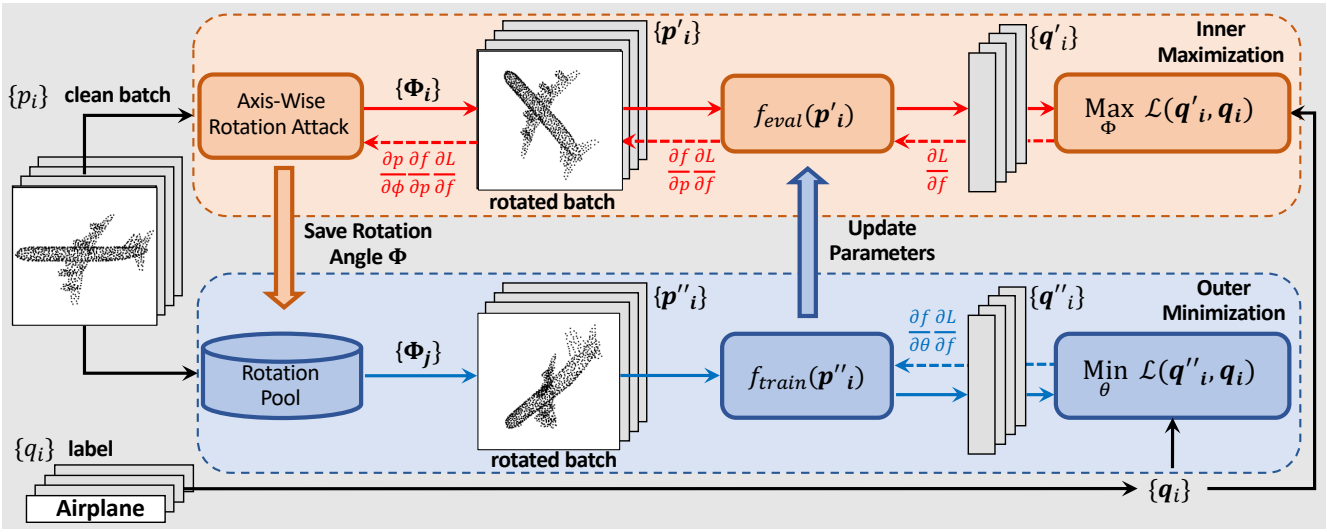

Figure 1. The general pipeline of our adversarial training approach. In the upper branch, the network takes a clean batch (aligned object) as inputs and finds the most aggressive attack angles by maximizing the classification loss of the eval model. The attack angles will be stored by class in the rotation pool. In the lower branch, the network samples angles from the rotation pool to produce adversarial point clouds for re-training the classifier to obtain the rotation robust model. The red and blue dashed lines respectively indicate routes of the backward gradient in two optimization tasks and point to the final optimized parameters. In the real implementations, the one-step optimization will construct the rotation pool by attacking multiple eval models, while the iterative optimization will update the parameter of the eval model by parameters of the latest re-trained model in each min-max iterations.

3.3. Attack—Inner Maximization

For the inner maximization problem, we expect a strong rotation attack algorithm that can find the most aggressive samples inducing high classification loss. A previous study [51] introduced two rotation attack methods, Thompson Sampling Isometry (TSI) attack and Combined Targeted Restricted Isometry (CTRI) attack, for generating adversarial rotations. However, they can hardly be used in adversarial training for the following reasons: (1) the TSI attack is a black-box attack, which has no direct access to the classifier parameters and thus can hardly be used to find samples inducing high loss. (2) CTRI attack is a white-box attack and one can use parameter information to search the most aggressive samples. Yet, in CTRI, there is no strict constraint for the matrix to be a pure rotation, which leads to adversarial samples with non-rigid deformation. To this end, we propose a novel white-box attack that finds the most aggressive samples while guaranteeing that the attack is pure rotation.

Gradient Descent on Angles. Firstly, to ensure the attack is pure rotation, we propose to optimize the attack by gradient descent on rotating angles. Specifically, for an n-point cloud $p = [x_i, y_i, z_i], i = 1...n$, we consider vectors $\Phi = [\phi_x, \phi_y, \phi_z]$ with 3 parameters denoting rotation angles along three axes. Rotating points along z axis by δ will increase the loss L by $\frac{\partial L}{\partial \phi_z}\delta$, which can then be calculated under the spherical coordinate, by the chain rule as:

$$\begin{aligned}\frac{\partial L}{\partial \phi_z} &= \sum_{i=1}^{n}\left(\frac{\partial x_i}{\partial \phi_z}\frac{\partial L}{\partial x_i} + \frac{\partial y_i}{\partial \phi_z}\frac{\partial L}{\partial y_i} + \frac{\partial z_i}{\partial \phi_z}\frac{\partial L}{\partial z_i}\right) \\ &= \sum_{i=1}^{n}\left(-y_i\frac{\partial L}{\partial x_i} + x_i\frac{\partial L}{\partial y_i}\right),\end{aligned} \tag{4}$$

where, $\frac{\partial L}{\partial x} = \nabla_x L(\theta, p, q)$ and $\frac{\partial L}{\partial y} = \nabla_y L(\theta, p, q)$ are gradients back-propagated on point coordinates. For the rest of the rotation axes, $\frac{\partial L}{\partial \phi_x}$ and $\frac{\partial L}{\partial \phi_y}$ can also be calculated in the same way. Based on Eq. (4), we can iteratively optimize the angles by gradient descent to obtain adversarial rotations that induce high loss. Finally, the rotation matrix is generated from optimized angles as $R = R_{\phi_z}R_{\phi_y}R_{\phi_x}$, where R_{ϕ_x} corresponds to the rotation matrix that rotates ϕ_x degrees around x axis. More derivations about the gradient calculation and rotation matrix construction will be provided in the supplementary.

Axis-Wise Attack. In order to efficiently find the most aggressive rotations, based on the angle gradients, we further propose an axis-wise mechanism. Specifically, we subdivide a rotation in SO(3) into rotations around three axes for optimization. By doing so, each time we can choose the most aggressive axis to rotate, resulting in stronger attacks. We approximate the loss change ratio of a specific axis by $|\frac{\partial L}{\partial \phi}|$, which reflects the influence of rotating around a certain axis on final losses. Next, we select the most influenced

Algorithm 1 Axis-Wise Rotation Attack

Require: Point cloud input p, label q and model parameters θ, loss function $L(\theta, p, q)$, number of iterations T, step size α, initial rotation angles $\Phi = [\phi_x, \phi_y, \phi_z]$ and corresponding rotation matrix $R = R_{\phi_x} R_{\phi_y} R_{\phi_z}$.

for $t = 0$ to T **do**
  Compute the gradients on coordinates:
  $\frac{\partial L}{\partial p^{(t)}} = [\frac{\partial L}{\partial x^{(t)}}, \frac{\partial L}{\partial y^{(t)}}, \frac{\partial L}{\partial z^{(t)}}]$.
  Compute the gradients on angles by Eq. (4).
  Determining the target axis by Eq. (5).
  Attack the target axis by Eq. (7).
  Update the rotation matrix:
  $R^{(t+1)} = R_{\phi_x^{(t+1)}} R_{\phi_y^{(t+1)}} R_{\phi_z^{(t+1)}}$
  Obtain the attacked point clouds: $p^{(t+1)} = R^{(t+1)} p$
end for
Output $R^{(T)}, p^{(T)}$

axis

$$\xi^* = \text{argmax}_\xi |\frac{\partial L}{\partial \phi_\xi}|, \xi \in [x, y, z], \tag{5}$$

and attack the axis by rotating one step in the opposite direction of gradient descent:

$$\phi_{\xi^*}^{(t+1)} = \phi_{\xi^*}^{(t)} + \alpha \text{sign}(\frac{\partial L}{\partial \phi_{\xi^*}}). \tag{6}$$

Compared with simultaneously optimizing on all three axes, the axis-wise attack can specify a gentler change of the rotation angles in each attack step.

Implementation Details. In the real implementations, we adopt several other general settings to find adversarial samples. Firstly, we use the Projected Gradient Descent (PGD) [22] to optimize angles. Compared with the normal gradient descent, PGD ensures that the optimized angles can be constrained into certain scopes:

$$\phi_{\xi^*}^{(t+1)} = \text{Proj}_{[-\pi,\pi]}(\phi_{\xi^*}^{(t)} + \alpha \text{sign}(\frac{\partial L}{\partial \phi_{\xi^*}})). \tag{7}$$

In our case, we set the projected scope as $[-\pi, \pi]$ to avoid the discontinuity caused by the periodicity of rotation. Then, instead of cross-entropy, we follow [43, 51] to adopt CW loss [3] to modify the cross-entropy as a more powerful adversarial objective to generate stronger adversary. Finally, to make sure that the generated adversary can be more evenly distributed among $[-\pi, \pi]$, we adopt a random start strategy. For each input point cloud, we will initialize it with a random rotation angle, then continue to attack along with the initialization angles. The proposed axis-wise rotation attack algorithm is illustrated in Algorithm (1).

3.4. Defense—Outer Minimization

On the defense side, we use Stochastic Gradient Descent (SGD) [2] to re-train the model on the adversarial samples.

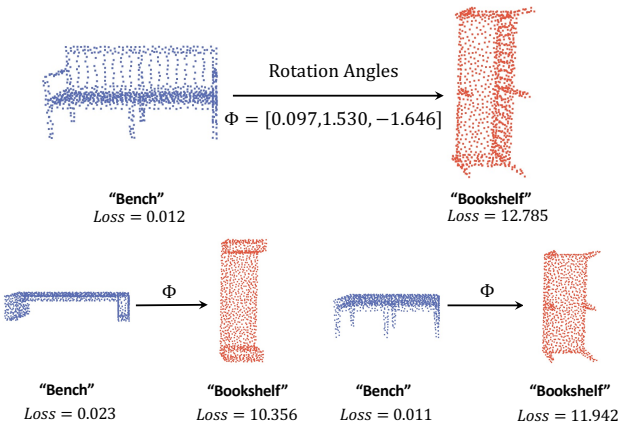

Figure 2. Transferability of adversarial rotations among samples in the same categories. The adversarial rotation found on one sample in "Bench" can be applied to other samples of the same category to induce high loss and mislead the model to classify them into a wrong category "Bookshelf".

During experiments, we find that for the original training set $\mathcal{A}$ and its attacked set $\mathcal{B}$ with rotations, directly training on set $\mathcal{B}$ can easily lead to model over-fitting. This behavior is known as label leaking [15] and stems from the fact that the gradient-based attack produces a very restricted set of adversarial examples that the network can overfit. The problem can be even worse on the smaller training set, in our case, ModelNet40 [42]. To solve the label leaking caused over-fitting problems, we propose to increase the training data with more kinds of adversarial rotations. A simple solution is to construct the training set $\mathcal{B}$ with multiple attack $\mathcal{B} = [\text{attack}_1(\mathcal{A}), \text{attack}_2(\mathcal{A}), \ldots, \text{attack}_i(\mathcal{A})]$. However, multiple attacks can be very time-consuming. To this end, we construct a rotation pool to increase the diversity of training data in a more efficient manner.

Rotation Pool. As shown in Fig. (4), we observe that the adversarial rotation found on one sample has a strong transferability on other samples of the same category. Based on this observation, instead of saving the rotated samples, we suggest saving the rotation angles produced on each sample by class to construct a rotation pool:

$$\mathcal{R} = [\{\Phi_{i,1}\}_{i=1}^{n_1}, \cdots, \{\Phi_{i,k}\}_{i=1}^{n_k}, \cdots, \{\Phi_{i,K}\}_{i=1}^{n_K}], \tag{8}$$

where $\Phi_{i,k}$ is the rotation found on sample i of category k. We will save the rotations corresponding to all n_k samples in the category k and traverse all K categories to construct the final rotation pool $\mathcal{R}$. During defense training, we only need to sample rotations from the rotation pool according to the category to transform the input into adversaries. Thanks to the transferability, the adversarial samples generated by the rotation pool can also induce high classification loss. Experiments in Sect. 4.5 also confirm that the rotation pool can effectively solve the over-fitting problem.

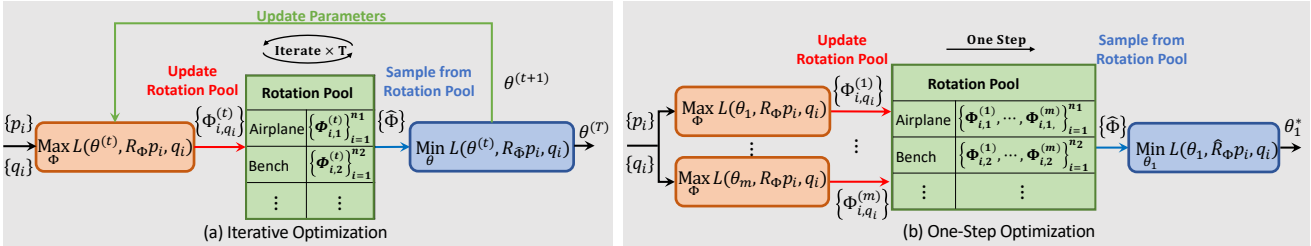

Figure 3. Comparison of different optimizations. For the iterative optimization (a), model with parameters θ will be repeatedly optimized on the min-max problem T times until converging to a robust parameter θ^T. In contrast, the proposed one-step optimization (b) constructs the rotation pool by attacking m different models and requires only one step to obtain robust parameters of the targeted model.

Iterative Optimization. In order to solve the minimization problem, *i.e.* Eq. (3), in adversarial training to reach the final robust models, an iterative optimization scheme is usually adopted. Specifically, in the first iteration, we will attack the pre-trained classifier to initialize the rotation pool and then re-train the classifier on adversarial samples generated from the rotation pool towards a robust model. In the following iterations, we will attack the latest robust model to update the rotation pool iteratively:

$$\Phi_{i,q_i}^{(t)} = \max_{\Phi} L(\theta^{(t)}, R_\Phi p_i, q_i), \tag{9}$$

where $\theta^{(t)}$ refers to the parameters of robust model after t iterations, R_ϕ is the rotation matrix of random start angles ϕ and q_i is the class label corresponding to input sample p_i. $\Phi_{i,q_i}^{(t)}$ refers to the rotation found on sample i of category q_i in the t-th iteration. We then re-train the classifier on the adversaries generated from the updated pool $\mathcal{R}^{(t)}$ to reach a more robust model. The process will be repeated until the model converges to the most robust state.

3.5. One-Step Optimization

The naive implementation above requires multiple iterations on both the attack and defense sides. Though obtaining robust models, the whole process is extremely time-consuming. Inspired by the ensemble adversarial training (EAT) [38], we further propose an efficient one-step optimization to reach the robust model with lower training cost.

Specifically, instead of iterating multiple times for obtaining more aggressive samples, EAT proposes to introduce the adversarial examples crafted on other stronger static pre-trained models. Intuitively, as adversarial samples transfer between models, perturbations crafted on the more robust model are good approximations for the maximization problem of the target model. We follow this principle to solve the minimization problem Eq. (3) in one step. Concretely, we not only attack the target classifier but attack more robust classifiers to construct a larger rotation pool:

$$\Phi_{i,q_i}^{(m)} = \max_{\Phi} L(\theta_m, R_\Phi p_i, q_i), \tag{10}$$

where, θ_m refers to the parameters of model m and $\Phi_{i,q_i}^{(m)}$ is the adversarial rotation generated by attacking model m. By attacking m models, the resulting rotation pool has m times more aggressive rotations than the iterative optimization does. For defense, similar to the iterative optimization, we use the adversarial rotation sampled from the rotation pool to re-train the target model. Compared with the iterative manner, the one-step optimization achieves competitive results with faster training progress. Hence, we select the one-step optimization as the default implementation of our ART-Point framework. The comparison between the two optimization methods is shown in Fig. (3). Detailed implementations and comparison experiments will be provided in the supplementary.

4. Experiments

4.1. Experiment Setup

Datasets. We evaluate our methods on two classification datasets ModelNet40 [42] and ShapeNet16 [46]. ModelNet40 contains 12,311 meshed CAD models from 40 categories. ShapeNet16 is a larger dataset which contains 16,881 shapes from 16 categories. For both datasets, we follow the official train and test split scheme and use the same data pre-processing as in [29, 30, 39] where each model is uniformly sampled with 1,024 points from the mesh faces and rescaled to fit into the unit sphere.

Models. We select three point cloud classifiers to evaluate our method, including PointNet [29], a pioneer network that processes points individually, PointNet++ [30], a hierarchical feature extraction network and DGCNN [39], a graph-based feature extraction network. These classifiers lack robustness to rotation. By verifying these classifiers, we show that ART-Point can be applied to various learning architectures to improve rotation robustness.

Evaluations. In order to comprehensively compare the rotation robustness of different models, we design three evaluation protocols: (1) Attack. The test set is adversarially rotated by the proposed attack algorithm for evaluating model defense. (2) Random. The test set is randomly

Method	ModelNet40		
	Attack	Random	Clean
PointNet [29] (RA)	55.6	74.4	76.7
PointNet++ [30] (RA)	58.9	80.1	82.3
DGCNN [39] (RA)	65.6	85.7	87.6
ART-PointNet (**Ours**)	85.6(30.0↑)	84.3(9.9↑)	85.5(8.8↑)
ART-PointNet++ (**Ours**)	90.1(31.2↑)	87.5(7.4↑)	88.6(6.3↑)
ART-DGCNN (**Ours**)	**91.5**(25.9↑)	**90.5**(4.8↑)	**91.3**(3.7↑)
Method	ShapeNet16		
	Attack	Random	Clean
PointNet [29] (RA)	66.4	87.3	89.5
PointNet++ [30] (RA)	70.5	89.7	92.1
DGCNN [39] (RA)	74.4	90.5	94.3
ART-PointNet (**Ours**)	96.9(30.5↑)	95.1(7.8↑)	96.2(6.7↑)
ART-PointNet++ (**Ours**)	97.8(27.3↑)	96.3(6.6↑)	97.5(5.4↑)
ART-DGCNN (**Ours**)	**98.4**(24.0↑)	**97.7**(7.2↑)	**98.1**(3.8↑)

Table 1. Comparing three evaluation protocols under ModelNet40 [42] and ShapeNet16 [46] for classifiers trained via rotation augmentation (RA) and adversarial rotation (ART).

rotated for evaluating model rotation robustness. (3) Clean. The test set is unchanged for evaluating the discriminative ability under aligned data. Moreover, we use the attack success rate to evaluate our attack algorithm. The attack success rate is calculated as the percentage of correctly predicted samples in the test set before and after the attack.

4.2. Comparison with Rotation Augmentation

We first compare the effectiveness of the proposed ART-Point with rotation augmentation (RA) for improving model rotation robustness. For classifiers using rotation augmentation, we will train them with randomly rotated inputs. In Tab. (1), we illustrate the comparison results under ModelNet40 [42] and ShapeNet16 [46]. From the table, several observations can be obtained. Firstly, compared with rotation augmentation, the proposed ART-Point results in models performing better under all protocols. Such performance improvements can be consistently observed on all three classifiers under both datasets. Secondly, under the attacked test set, the classification accuracy of model trained using ART-point is significantly higher than model trained with RA. (maximum increase: 31.2%). This is mainly because that rotation augmentation can hardly defend against adversarial rotations found using model gradient information. In contrast, our method shows stronger defense to adversarial rotations. We will further test the defense ability of our method under different rotation attacks in Sect. 4.4. Both observations suggest that the proposed ART-Point is a more effective method to improve the rotation robustness of point cloud classifiers than rotation augmentation.

Method	ModelNet40		
	Attack	Random	Clean
Classifiers Using Invariant Descriptors			
SFCNN [31]	90.1	90.1	90.1
RI-Conv [48]	86.5	86.4	86.5
ClusterNet [4]	87.1	87.1	87.1
RI-Framework [17]	89.4	89.3	89.4
Classifiers with Equivariant Architectures			
TFN [37]	87.6	87.6	87.6
REQNN [34]	74.4	74.1	74.4
VN-PointNet [7]	77.2	77.2	77.2
VN-DGCNN [7]	90.2	90.2	90.2
EPN [5]	88.3	88.3	88.3
Ours			
ART-PointNet	85.6	84.3	85.5
ART-PointNet++	90.1	87.5	88.6
ART-DGCNN	**91.5**	**90.5**	**91.3**

Table 2. Comparing three evaluation protocols under ModelNet40 [42] for various rotation robust classifiers.

4.3. Comparison with Rotation Robust Classifiers

We further compare robust models trained by ART-Point with existing rotation robust classifiers, including [4,17,31,48] that convert point clouds into rotation invariant descriptors and [5,7,34,37] that design rotation-equivariant architectures, to further illustrate appealing properties of our method. Rotation robust classifiers will be trained on random rotated inputs. The comparison results based on all protocols under ModelNet40 [42] are shown in Tab. (2). Firstly, our best model ART-DGCNN outperforms all equivariant or invariant methods under three evaluation protocols, which indicates its stronger robustness over rotations. Secondly, both equivariant or invariant methods perform similarly under all protocols, which is undesirable, since the clean test set should more easily be classified by the model. This is mainly because that these methods obtain rotation robustness by separating the pose information from point clouds via modifications on input space or model architectures. In contrast, ART-Point uses original classifiers for training on adversarial samples in 3D space, the resulting model not only better inherits the performance of original classifiers on clean sets but shows great defense on the attacked test set.

4.4. Attack and Defense

Beyond rotation robustness, our method provides a complete set of tools for attack and defense on point cloud classifiers. To verify the proposed attack algorithm, we compare the attack success rate of our method with other rotation attacks proposed in [51]. Meanwhile, we also show the defense ability of classifiers trained with ART-Point. The

Models	Rotation Attack Algorithm		
	TSI [51]	CTRI [51]	Ours
PointNet [29]	96.92	99.44	99.54
PointNet++ [30]	91.31	97.93	98.96
DGCNN [39]	89.81	97.99	98.51
ART-PointNet (**Ours**)	9.71	11.13	12.78
ART-PointNet++ (**Ours**)	4.31	6.60	7.92
ART-DGCNN (**Ours**)	3.14	5.33	6.62

Table 3. Comparing attack success rate (%) of several attack algorithms on different classifiers under ModelNet40 [42].

Methods	Loss	Acc.	Methods	Loss	Acc.
Random	5.13	74.4	w/o RP	12.72	55.8
TSI [51]	7.35	79.5	RP(pn1)	10.19	82.9
CTRI [51]	8.87	82.1	RP(pn1,pn2)	12.01	82.6
Ours (step=1)	7.65	81.5	RP(pn1,dg)	12.55	83.1
Ours (step=5)	9.57	82.8	RP(pn2,dg)	13.03	84.0
Ours (step=10)	**13.49**	**84.3**	RP(pn1,pn2,dg)	**13.49**	**84.3**

Table 4. The average loss of adversarial samples generated by different methods and accuracy of corresponding adversarial training. RP(pn1) refers to the rotation pool generated by attacking PointNet [29]. pn2 and dg refer to PointNet++ [30] and DGCNN [39].

results are illustrated in Tab. (3). In the first three rows, we report the attack success rate of different attack algorithms on classifiers trained using clean samples. As can be seen, compared with the other two rotation attacks, our attack achieves the highest success rate on all three classifiers. In the last three rows, we further report the attack success rate on classifiers trained using ART-Point. As can be seen, ART-Point improves model defense against rotation attacks.

4.5. Ablation Study

Finally, we conduct ablation studies to prove the effectiveness of our designs in ART-Point. All ablation experiments are conducted on the PointNet [29] classifier and evaluated under randomly rotated test sets[1].

Different Attacks. We use adversarial samples generated by different rotation attacks for adversarial training and investigate the impact on the robustness of the resulting models. We adopt several attacks to generate adversarial samples that induce different loss values, including the random rotation attack, attacks in [51] and our attacks with different steps. In the left column of Tab. (4), we illustrate the average classification loss of samples produced by different attacks and results of adversarial training using corresponding samples.

Rotation Pool. We verify the necessity of constructing the rotation pool. We compare the results of adversarial training with, without rotation pools and constructing rotation pools from different models. As shown in the right column of Tab. (4), although adversarial training without rotation pool generates samples inducing high loss values, the final result is worse than training with rotation pool due to the over-fitting caused by label leaking [15].

[1]More ablation studies on descent step, rotation angle, and attack step size can be found in the supplementary material.

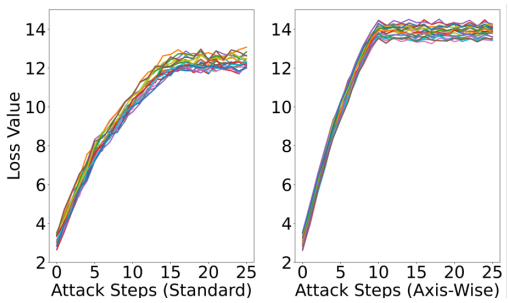

Figure 4. Averaged loss values of attacked samples produced by standard attack and axis-wise attack under different attack steps.

Axis-Wise Attack. We compare our proposed axis-wise rotation attack with the standard attack algorithm, which simultaneously optimizes three angles in one gradient descent. We mainly follow [22] to show the average loss value of attacked samples in each step. We restart the attack 20 times with random angle initialization. The comparison results are shown in Fig. (4). As can be seen, the axis-wise mechanism enables the attack algorithm to find more aggressive rotated samples.

4.6. Discussions of Limitations and Society Impact

Since our method is mainly based on adversarial training, one limitation is that we need to obtain a fully trained model with accessible parameters in the first place. Meanwhile, the rotating attack algorithm may be exploited for attacking point cloud based 3D object detection systems, which is a potential negative societal impact.

5. Conclusion

In this paper, we propose ART-Point to improve the rotation robustness of point cloud classifiers via adversarial training. ART-Point consists of an axis-wise rotation attack and a defense method with the rotation pool mechanism. It can be adopted on most existing classifiers with fast one-step optimization to obtain rotation robust models. Experiments show that the novel rotation attack achieves a high attack success rate on most point cloud classifiers. Moreover, our best model ART-DGCNN shows great robustness to arbitrary and adversarial rotations and outperforms existing state-of-the-art rotation robust classifiers.

Acknowledgment This work is supported by the Major Science and Technology Innovation 2030 "Brain Science and Brain-like Research" key project (No. 2021ZD0201402 and 2021ZD0201405).

References

[1] Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. Recent advances in adversarial training for adversarial robustness. *arXiv preprint arXiv:2102.01356*, 2021. 3

[2] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010. 5

[3] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017. 5

[4] Chao Chen, Guanbin Li, Ruijia Xu, Tianshui Chen, Meng Wang, and Liang Lin. Clusternet: Deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4994–5002, 2019. 1, 2, 7

[5] Haiwei Chen, Shichen Liu, Weikai Chen, Hao Li, and Randall Hill. Equivariant point network for 3d point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14514–14523, 2021. 1, 2, 7

[6] Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. *arXiv preprint arXiv:1801.10130*, 2018. 2

[7] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas Guibas. Vector neurons: A general framework for so (3)-equivariant networks. *arXiv preprint arXiv:2104.12229*, 2021. 1, 3, 7

[8] Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 602–618, 2018. 1, 2

[9] Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. Hotflip: White-box adversarial examples for text classification. *arXiv preprint arXiv:1712.06751*, 2017. 2, 3

[10] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68, 2018. 2

[11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016. 2, 3

[12] Chengyue Gong, Tongzheng Ren, Mao Ye, and Qiang Liu. Maxup: Lightweight adversarial training with data augmentation improves neural network training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2474–2483, 2021. 3

[13] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 3

[14] Lingshen He, Yuxuan Chen, Yiming Dong, Yisen Wang, Zhouchen Lin, et al. Efficient equivariant network. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[15] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016. 2, 5, 8

[16] Itai Lang, Uriel Kotlicki, and Shai Avidan. Geometric adversarial attacks and defenses on 3d point clouds. *arXiv preprint arXiv:2012.05657*, 2020. 3

[17] Xianzhi Li, Ruihui Li, Guangyong Chen, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. A rotation-invariant framework for deep point cloud analysis. *IEEE Transactions on Visualization and Computer Graphics*, 2021. 1, 2, 7

[18] Daniel Liu, Ronald Yu, and Hao Su. Extending adversarial attacks and defenses to deep 3d point cloud classifiers. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2279–2283. IEEE, 2019. 2, 3

[19] Hongbin Liu, Jinyuan Jia, and Neil Zhenqiang Gong. Pointguard: Provably robust 3d point cloud classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6186–6195, 2021. 3

[20] Min Liu, Fupin Yao, Chiho Choi, Ayan Sinha, and Karthik Ramani. Deep learning 3d shapes using alt-az anisotropic 2-sphere convolution. In *International Conference on Learning Representations*, 2018. 2

[21] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742*, 2017. 2, 3

[22] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 1, 3, 5, 8

[23] Pratyush Maini, Eric Wong, and Zico Kolter. Adversarial robustness against the union of multiple perturbation models. In *International Conference on Machine Learning*, pages 6640–6650. PMLR, 2020. 3

[24] Takeru Miyato, Andrew M Dai, and Ian Goodfellow. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*, 2016. 3

[25] John X Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. *arXiv preprint arXiv:2005.05909*, 2020. 3

[26] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. *arXiv preprint arXiv:2010.00467*, 2020. 3

[27] Adrien Poulenard and Leonidas J Guibas. A functional approach to rotation equivariant non-linearities for tensor field networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13174–13183, 2021. 1, 2

[28] Adrien Poulenard, Marie-Julie Rakotosaona, Yann Ponty, and Maks Ovsjanikov. Effective rotation-invariant point cnn with spherical harmonics kernels. In *2019 International Conference on 3D Vision (3DV)*, pages 47–56. IEEE, 2019. 2

[29] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference*

on computer vision and pattern recognition, pages 652–660, 2017. 1, 2, 3, 6, 7, 8

[30] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017. 1, 2, 3, 6, 7, 8

[31] Yongming Rao, Jiwen Lu, and Jie Zhou. Spherical fractal convolutional neural networks for point cloud recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 452–460, 2019. 2, 7

[32] Lukas Schott, Jonas Rauber, Matthias Bethge, and Wieland Brendel. Towards the first adversarially robust neural network model on mnist. *arXiv preprint arXiv:1805.09190*, 2018. 3

[33] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *arXiv preprint arXiv:1904.12843*, 2019. 2, 3

[34] Wen Shen, Binbin Zhang, Shikun Huang, Zhihua Wei, and Quanshi Zhang. 3d-rotation-equivariant quaternion neural networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 531–547. Springer, 2020. 1, 3, 7

[35] Jiachen Sun, Yulong Cao, Christopher B Choy, Zhiding Yu, Anima Anandkumar, Zhuoqing Morley Mao, and Chaowei Xiao. Adversarially robust 3d point cloud recognition using self-supervisions. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[36] Jiachen Sun, Karl Koenig, Yulong Cao, Qi Alfred Chen, and Zhuoqing Mao. On the adversarial robustness of 3d point cloud classification. 2020. 2, 3

[37] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018. 1, 2, 7

[38] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017. 2, 6

[39] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics*, 38(5):1–12, 2019. 1, 2, 3, 6, 7, 8

[40] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco Cohen. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *arXiv preprint arXiv:1807.02547*, 2018. 1, 2

[41] Yi Wu, David Bamman, and Stuart Russell. Adversarial training for relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1778–1783, 2017. 3

[42] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 2, 5, 6, 7, 8

[43] Chong Xiang, Charles R Qi, and Bo Li. Generating 3d adversarial point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9136–9144, 2019. 3, 5

[44] Cihang Xie and Alan Yuille. Intriguing properties of adversarial training at scale. *arXiv preprint arXiv:1906.03787*, 2019. 2, 3

[45] Jiancheng Yang, Qiang Zhang, Rongyao Fang, Bingbing Ni, Jinxian Liu, and Qi Tian. Adversarial attack and defense on point sets. *arXiv preprint arXiv:1902.10899*, 2019. 3

[46] Li Yi, Vladimir G Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (ToG)*, 35(6):1–12, 2016. 2, 6, 7

[47] Zhiyuan Zhang, Binh-Son Hua, Wei Chen, Yibin Tian, and Sai-Kit Yeung. Global context aware convolutions for 3d point cloud understanding. In *2020 International Conference on 3D Vision (3DV)*, pages 210–219. IEEE, 2020. 1, 2

[48] Zhiyuan Zhang, Binh-Son Hua, David W Rosen, and Sai-Kit Yeung. Rotation invariant convolutions for 3d point clouds deep learning. In *2019 International Conference on 3D Vision (3DV)*, pages 204–213. IEEE, 2019. 1, 2, 7

[49] Chen Zhao, Jiaqi Yang, Xin Xiong, Angfan Zhu, Zhiguo Cao, and Xin Li. Rotation invariant point cloud classification: Where local geometry meets global topology. *arXiv preprint arXiv:1911.00195*, 2019. 1, 2

[50] Yongheng Zhao, Tolga Birdal, Jan Eric Lenssen, Emanuele Menegatti, Leonidas Guibas, and Federico Tombari. Quaternion equivariant capsule networks for 3d point clouds. In *European Conference on Computer Vision*, pages 1–19. Springer, 2020. 1, 3

[51] Yue Zhao, Yuwei Wu, Caihua Chen, and Andrew Lim. 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*, pages 1201–1210, 2020. 1, 2, 3, 4, 5, 7, 8

[52] Tianhang Zheng, Changyou Chen, Junsong Yuan, Bo Li, and Kui Ren. Pointcloud saliency maps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1598–1606, 2019. 3