Rethinking Perturbation Directions for Imperceptible Adversarial Attacks on Point Clouds

Keke Tang, *Member, IEEE*, Yawen Shi, Tianrui Lou, Weilong Peng, Xu He, Peican Zhu, *Member, IEEE*, Zhaoquan Gu, *Member, IEEE*, and Zhihong Tian, *Senior Member, IEEE*

***Abstract*—Adversarial attacks have been successfully extended to the field of point clouds. Besides applying the common perturbation guided by the gradient, adversarial attacks on point clouds can be conducted by applying directional perturbations, e.g., along normal and along the tangent plane. In this article, we first investigate whether adversarial attacks with these two orthogonal directional perturbations are more imperceptible than that with the gradient-aware perturbation. Second, we investigate the deeper difference between adversarial attacks with these two directional perturbations, and whether they are applicable to the same scenarios. Third, based on the verification results that the above two directional perturbations have different sensitiveness to curvature, we devise a novel normal-tangent attack (NTA) framework with a hybrid directional perturbation scheme that adaptively chooses the direction according to the curvature of the local shape around the point. Extensive experiments on two publicly available data sets, e.g., ModelNet40 and ShapeNet Part, with classifiers in three representative networks, e.g., PointNet++, DGCNN, PointConv, validate the effectiveness of NTA, and the superiority to the state-of-the-art methods.**

***Index Terms*—Adversarial attack, direction, imperceptible perturbation, point clouds.**

Manuscript received 30 July 2022; revised 29 October 2022; accepted 10 November 2022. Date of publication 15 November 2022; date of current version 7 March 2023. This work was supported in part by the Science and Technology Innovation 2030 "New Generation Artificial Intelligence" Major Project under Grant 2020AAA0107700; in part by the National Natural Science Foundation of China under Grant 62102105, Grant 62073263, Grant U20B2046, and Grant 61902082; in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2020A1515110997, Grant 2022A1515011501, and Grant 2022A1515010138; in part by the Science and Technology Program of Guangzhou under Grant 202002030263, Grant 202102010419, and Grant 202201020229; in part by the Open Project Program of the State Key Lab of CAD&CG, Zhejiang University under Grant A2218; in part by the Guangdong Higher Education Innovation Group under Grant 2020KCXTD007; and in part by the Guangzhou Higher Education Innovation Group under Grant 202032854. *(Keke Tang and Yawen Shi are co-first authors.) (Corresponding authors: Peican Zhu; Zhihong Tian.)*

Keke Tang, Yawen Shi, Tianrui Lou, Xu He, and Zhihong Tian are with the Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou 510006, China (e-mail: tangbohutbh@gmail.com; shiyawen666@gmail.com; loutianrui@gmail.com; heexu976@gmail.com; tianzhihong@gzhu.edu.cn).

Weilong Peng is with the School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou, Guangzhou 510006, China (e-mail: wlpeng@gzhu.edu.cn).

Zhaoquan Gu is with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China, and also with the Department of New Networks, Peng Cheng Laboratory, Shenzhen 518066, China (e-mail: zqgu@gzhu.edu.cn).

Peican Zhu is with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: ericcan@nwpu.edu.cn).

Digital Object Identifier 10.1109/JIOT.2022.3222159

I. Introduction

With the development of deep neural networks (DNNs) [1], [2], [3], [4], [5], researchers race to utilize them for handing 3-D point cloud perception tasks, significantly refreshing the state-of-the-art records [6], [7], [8]. While DNNs make huge performance improvements, they also bring many fatal security risks. One typical risk of DNNs is the vulnerability to adversarial attacks [9], i.e., imperceptible modifications on the input can lead to erroneous predictions of victim DNN models, and thus hinders their deployment in safety-critical applications, e.g., autonomous driving [10] and robotic task planning [11]. Therefore, understanding the rationality of adversarial attacks on DNNs for point clouds is important for preventing this risk.

There have been many attempts in attacking DNN classifiers for 3-D point clouds. According to the strategy to apply adversarial attacks, current solutions can be broadly classified into three categories, i.e., addition-based solutions that attach adversarial points, clusters, and objects [12], deletion-based solutions that discard a small number of salient points [13], and the most commonly adopted perturbation-based solutions. Among them, the perturbation-based solutions can allocate the perturbation to all points in the cloud, and can borrow the knowledge from adversarial attacks in the mature image field [14], and thus is more promising in achieving imperceptible adversarial attacks.

A common way to apply perturbation is to calculate the directions for different points, i.e., the reverse of the gradient directions (GDs) that are expected to make the classification loss larger, and then move these points along them iteratively. Since point clouds and images are different, applying perturbation on point clouds in an exactly the same way as in images may not be a good strategy. Indeed, some pioneering studies have been already aware of it and have attempted to consider it in a more geometrical view. Wen et al. [15] added a curvature consistency term between adversarial and benign point clouds during the perturbation process along gradient-guided directions. Instead of utilizing extra distance loss, Liu and Hu [16] explicitly constrained the perturbation to be along the normal direction (ND) of each point, and Huang et al. [17] restricted them to be on the tangent plane. Since all these solutions with perturbation applied along different directions claim that they achieve imperceptible adversarial attack, see Fig. 1, it thus brings us to the main topic of this article: which is the best perturbation direction for imperceptible adversarial attacks on point clouds?

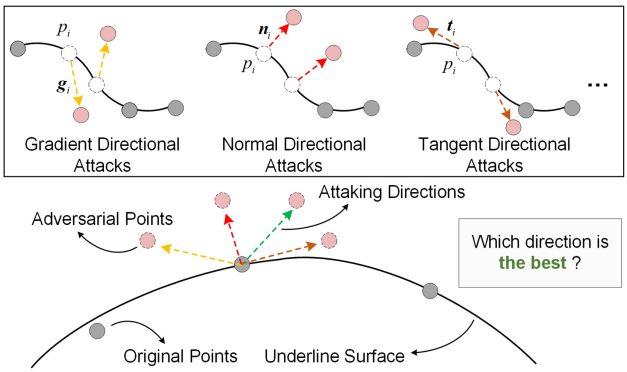


Fig. 1. Adversarial attacks on point clouds can be achieved by applying directional perturbations, e.g., along the direction of gradient ascent, along the normal, and along the tangent plane of the underline surface. Which direction is the best remains to be an open problem.

To answer the above question, we investigate adversarial attacks on DNN classifiers for point clouds in three aspects. First, adversarial attacks via applying perturbation along normal or tangent directions (TDs) differ a lot with the gradient-based attack, e.g., more perturbation steps does not necessarily lead to a higher attack success rate (ASR). Therefore, to avoid the interference of other factors caused by different perturbation directions as much as possible, we relax the strict directional restrictions to the minimization of the projection distance of the gradient, and then conduct extensive comparisons between these configurations. Second, after validating the superiority of applying geometrical directional perturbations, e.g., along normal and along the tangent plane, we further investigate why these two conflicting configurations could both work. Specifically, we compare the performance of attacking along normal and TDs (NA and TA) under different perturbation weights with respect to curvature. The results show that NA works better at regions with larger curvature and TA works better at regions with smaller curvature, and thus prove their rationalities, i.e., zooming in/out and rotating, and indicate their different applicable scenarios. Finally, considering that the applicable scenarios of NA and TA are complementary to each other, we combine these two strategies into a hybrid normal-tangent attack (NTA) by designing a directional controlling module to adaptively choose the perturbation directions according to the curvature. Extensive experiments on two publicly available benchmarks with classifiers in three representative DNN architectures validate that NTA is imperceptible, and outperforms the state-of-the-art methods.

Overall, our contribution is at least threefold.

1) We compare adversarial attacks along three different directions in a fair comparison scheme.
2) We investigate applicable scenarios for adversarial attacks along normal and TDs.
3) We devise a hybrid framework that combines adversarial attacks along normal and TDs.

II. Related Work

A. DNN Models for Point Cloud Classification

DNN models have been widely adopted for handling the point cloud classification task. Since point clouds are irregular, early attempts first convert them into structured grid representations, e.g., projecting into multiview images [18] or rasterizing into 3-D voxel grids [19], and then utilize mature 2-D DNNs models to process them. Until very recently, the pioneering PointNet [20] validated that the structure of multilayer perceptrons (MLPs) followed with maximum pooling can overcome the unorder issue of point clouds, researchers start to explore processing point clouds directly. To solve the problem that PointNet can not recognize fine-grained patterns, PointNet++ [6] are further proposed to capture the fine geometric structure of point clouds by hierarchically applying PointNet to the neighborhood of each point. Besides these MLP-based solutions, another mainstream DNN architectures for point cloud classification include convolution-based KPConv [21], PointCNN [22], and graph-based DGCNN [7], and some of their combinations [23], [24], [25], etc. Since DNN models for 3-D point cloud classification is now a large and very active research field, we refer the readers to the survey papers [8], [26], [27] for more complete reviews. In this article, we investigate applying adversarial attacks on these DNN models.

B. Adversarial Attacks on Point Clouds

The vulnerability of DNNs that an imperceptible perturbation on the input can lead them to make an error prediction is first discovered in the image field [9]. Since then, extensive studies have been conducted on attacking DNN models for image classification [14], [28], [29], [30]. Please refer to [31], [32], and [33] for more complete surveys.

With the popularity of utilizing DNN models in handling point cloud classification, researchers have started to extend adversarial attacks from the field of images to that of point clouds. According to the way to apply adversarial attacks, current solutions can be broadly divided into three categories: 1) addition based; 2) deletion based; and 3) perturbation based. Addition-based solutions [34] perform adversarial attacks by attaching a limited number of synthetic points, clusters, and objects to the point clouds. On the contrary, deletion-based solutions discard a small number of salient points. Wicker and Kwiatkowska [35] proposed to determine the critical points in a random and iterative manner and then generated adversarial examples for attack by deleting the critical points. Yang et al. [12] found key points by calculating the importance scores associated with the labels obtained from the output of the classifier relative to the gradient of the input, and then deleted key points in a similar manner. Instead of deleting the points, Zheng et al. [13] devised a more flexible way that moves the points with high saliency toward the center of the shape, such that these points will not influence the surfaces.

Compared with the above two directions, perturbation-based solutions are more popular. Liu et al. [36] extended the FGSM [14] by adding a l_2-norm as constraint to construct imperceptible adversarial 3-D point clouds. Lee et al. [37] added adversarial noise to the latent space of an auto-encoder, keeping the decoded shape similar to the original one. To achieve better imperceptibility, Kim et al. [38] proposed to perturb a minimal subset of points, instead of all of them.

However, very little work exploited the geometric property of point clouds to improve the imperceptibility of generated adversarial point clouds.

C. Utilizing Geometry Properties for Imperceptibility

Since point clouds are surfaces of 3-D model shapes, geometric properties are critical cues for making adversarial attacks to be imperceptible. Tsai et al. [39] introduced a k-nearest neighbor loss into the C&W framework to ensure the compactness of the local neighborhoods in adversarial point clouds. To restrict the perturbation, Wen et al. [15] enforced the consistency of local curvature between the adversarial points and benign ones.

Different from the above solutions, we focus on another geometric property, i.e., the perturbation direction. Indeed, some recent work has started to investigate it. Liu and Hu [16] and Tang et al. [40] restricted each point move along the ND, while Huang et al. [17] restricted the perturbation to be along the 2-D projection plane. Both works achieved successful adversarial attacks with high imperceptibility. Since the above two directions are totally orthogonal, in this article, we focus on investigating the difference of their rationality and applicable scenarios. Besides, by borrowing the benefits of adversarial attacks in both normal and TDs, we propose a hybrid framework.

III. Review on Directional Adversarial Attacks and Perturbation Directions

A. Directional Adversarial Attacks

Notations: This work considers the setting in a C-category point cloud classification problem. Let $\mathcal{P}$ be an input point cloud, containing a set of unordered points $\{p_i\}_1^n \in \mathbb{R}^{n\times 3}$ sampled from the surface of a 3-D object, where each point $p_i \in \mathbb{R}^3$ contains coordinate positions. We denote n_i as the normal of p_i and denote $F(\cdot)$ as the classifier, e.g., PointNet++, that predicts the category to which $\mathcal{P}$ belongs.

Formulation of Adversarial Attack: Suppose $F(\cdot)$ can originally correctly classify the category of point cloud $\mathcal{P}$

$$y = F(\mathcal{P}), \quad y \in \{1, 2, \ldots, C\} \tag{1}$$

where y denotes the ground truth label of $\mathcal{P}$, adversarial attack aims to find a human-imperceptible perturbation $\triangle$, such that $F(\cdot)$ will make an error prediction on the adversarial point cloud

$$y' = F(\mathcal{P} + \triangle), \quad y' \neq y. \tag{2}$$

Note that the above formulation describes the situation of an untargeted attack, while an targeted attack can be achieved by additionally designating the expected category to be predicted. In this article, we only consider untargeted attack.

We summarize all adversarial attack methods that apply directional perturbation $\triangle$, i.e., along a specific direction (iteratively), as *directional adversarial attacks*.

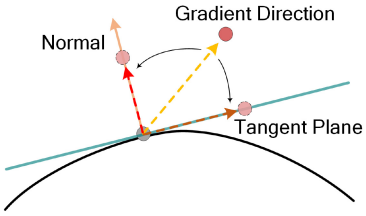


Fig. 2. Demonstration of applying a projection restriction on the normal and TDs to simulate ND and TD.

B. Perturbation Directions of Attacks

According to the perturbation direction, current directional adversarial attacks can be broadly divided into three categories: 1) gradient directional attack; 2) normal directional attack; and 3) tangent directional attack, as seen in Fig. 1. As the name suggests, they apply attack in the following three directions.

Gradient Direction: Gradient directional attacks (GA) apply perturbation along the direction from which the cross-entropy loss for category classification enlarges the most, i.e., the direction of gradient ascent

$$\triangle(p_i) \leftarrow -\text{sign}\left(\frac{\partial L(\mathcal{P}, y)}{\partial p_i}\right) \tag{3}$$

where $L(\mathcal{P}, y)$ indicates the loss of category classification, and $\leftarrow$ indicates "is determined by." Most current adversarial attack solutions for deep point cloud classifiers belong to this category of attacks.

Normal Direction: Normal directional attacks (NA) apply perturbation along the direction of normal of each point

$$\triangle(p_i) \leftarrow n_i. \tag{4}$$

ITA [16] is a representative work of this category of attacks.

Tangent Direction: Tangent directional attacks (TA) apply perturbation along the tangent plane of the local shape of each point

$$\triangle(p_i) \leftarrow \perp n_i \tag{5}$$

where $\perp n_i$ indicates the tangent plane at point p_i. SI-ADV [17] is a representative work of this category of attacks.

C. Open Questions

GA applies perturbation in the view of optimization, which is in the same way as in the image-based adversarial attacks. Differently, NA and TA apply perturbation in the view of geometry, e.g., along normal and TDs. In this article, we will try to answer the below two questions.

Q1: Which is the best among NA, TA, and GA?

Indeed, both the NA and TA solutions, e.g., ITA [16] and SI-ADV [17], have claimed that they achieve more imperceptible adversarial attacks. However, more fair comparisons between all the three solutions are still required.

Q2: What are the applicable scenarios of NA and TA?

Since point clouds represent geometrical shapes, we have expected that the geometric properties of shapes, e.g., perturbation direction, can be exploited to help improve the imperceptibility adversarial attacks. However, since normal

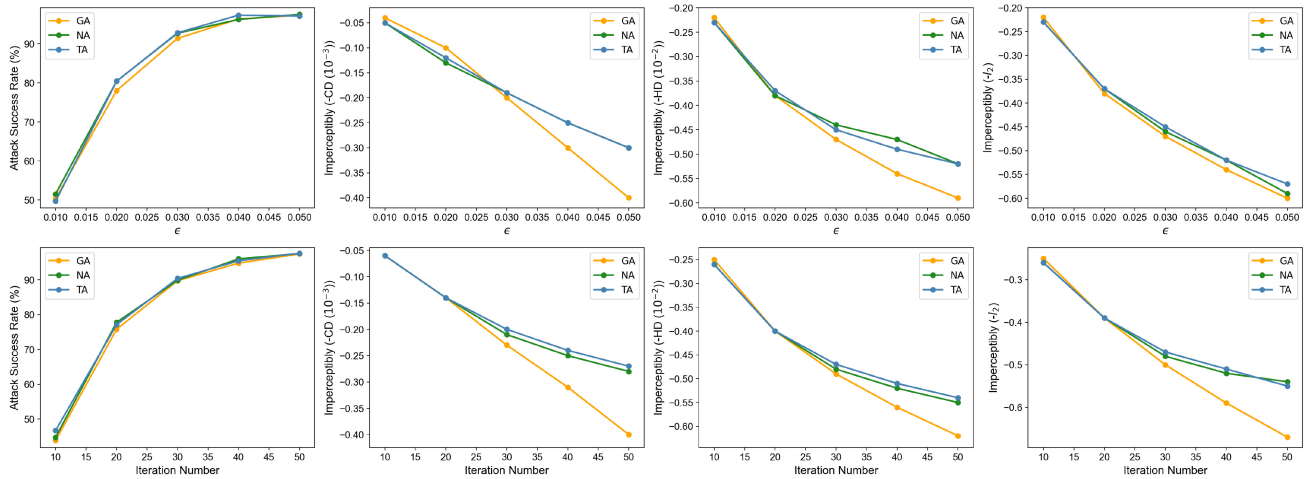


Fig. 3. Comparison of GA, NA, and TA. Top row: ASR and imperceptibility of these three adversarial attack methods with a fixed 80 iteration but different perturbation step sizes. Bottom row: ASR and imperceptibility of these three adversarial attack methods with a fixed perturbation step size of 0.05 but different iterations.

and TDs are orthogonal with each other, if these two seemingly contradictory strategies both work, they should have different applicable scenarios.

IV. Analysis on Different Attack Directions

To answer the two questions raised in Section III-C, we first intentionally devise a fair comparison setting, and then extensively evaluate the performance of adversarial attacks along the three directions, i.e., GD, ND, and TD, and finally investigate their sensitiveness to the curvature.

A. Comparison Setting

To avoid the influence of many other factors, we do not strictly restrict the perturbation of ND and TD to be along normal and TDs. Instead, we simply enforce the projection of perturbation on tangent and NDs to be small. Therefore, the only difference between GA, NA, and TA is whether applying a projection restriction on the normal and TDs (see Fig. 2).

To better illustrate the performance difference between different attack directions, i.e., GD, ND, and TD, we evaluate the three-directional adversarial attacks in two respects. First, we report the ASR and three imperceptibility metrics measured by l_2-norm distance (l_2), Chamfer distance (CD), and Hausdorff distance (HD) with different perturbation step sizes in 80 iterations. Second, we draw the ASR and three imperceptibility metrics during the iterative attack process with a fixed a perturbation step size, e.g., 0.05.

B. Which Is the Best Among GA, NA, and TA?

To answer this question, we compare GA, NA, and TA on attacking PointNet++ trained on ModelNet40. The results drawn in Fig. 3 show that attacking along all three directions can achieve high ASR, e.g., nearly 100%. However, the three imperceptibility metrics of attacking along GA are significantly lower than those of NA and TA, both when setting a large perturbation step size and in the whole iterative attacking process.

Overall, we conclude that NA and TA are comparable, and are both better than GA.

C. Are ND and TD Applicable for the Same Curvature?

In this section, we take curvature as an example to investigate whether adversarial attacks by applying perturbation along ND and TD perform always the same.

Perturb More on Larger Curvature Regions: We first investigate whether attacking along ND and TD are more applicable to be performed at regions with larger curvature. Specifically, instead of applying the same step size for all points, we additionally multiply a curvature-aware weight for perturbation at each point, e.g., p_i

$$w_l(i) = \frac{1}{1 + e^{-t \cdot \mathrm{cur}(p_i)}} \tag{6}$$

where $\mathrm{cur}(p_i)$ is the curvature of p_i and t is a temperature scaling parameter in positive value. The results drawn in Fig. 4 show that both the ASR and imperceptibility of NA are now superior to TA and GA, validating that attacking along NA is more applicable for regions with large curvature.

Perturb More on Smaller Curvature Regions: Similarly, we apply a reversed curvature-aware weight to the perturbation step size at different points

$$w_s(i) = \frac{1}{1 + e^{t \cdot cur(p_i)}}. \tag{7}$$

In this setting, smaller step sizes are applied for regions with larger curvature. The results drawn in Fig. 5 show that both the ASR and imperceptibility of TA are now superior to NA and GA, validating that attacking along TD is more applicable for regions with small curvature.

Discussion: Indeed, the reason why the performance of attacking along ND and TD differs a lot with respect to the area curvature attributes to their rationalities. NA obtains its imperceptibility by implementing the attacks via slightly zooming in or zooming out the object shapes. Therefore, the zooming is more difficult to be aware at regions with

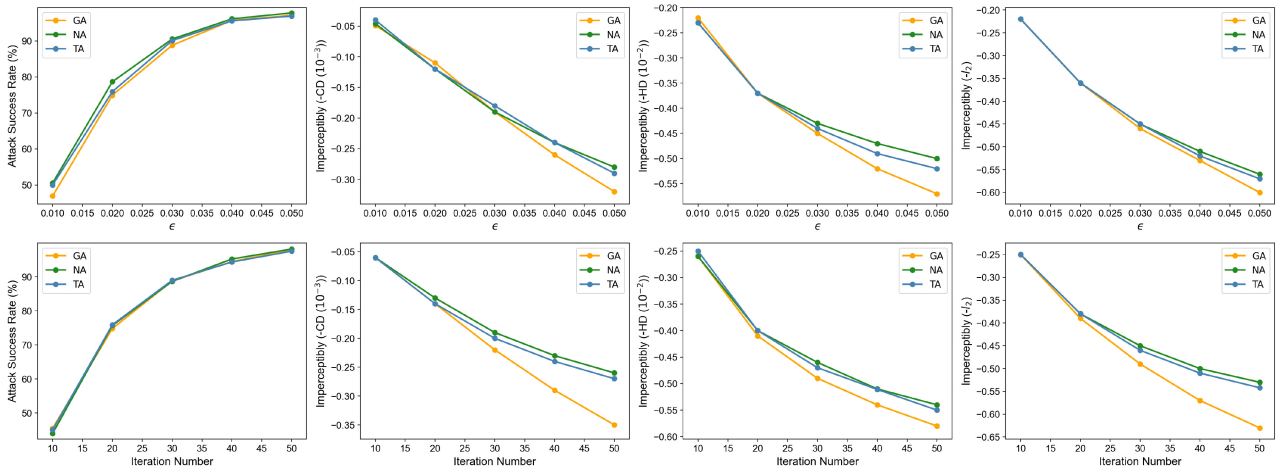


Fig. 4. Comparison of GA, NA, and TA that perturb more on larger curvature regions. Top row: ASR and imperceptibility of these three adversarial attack methods with a fixed 80 iteration but different perturbation step sizes. Bottom row: ASR and imperceptibility of these three adversarial attack methods with a fixed perturbation step size of 0.05 but different iterations.

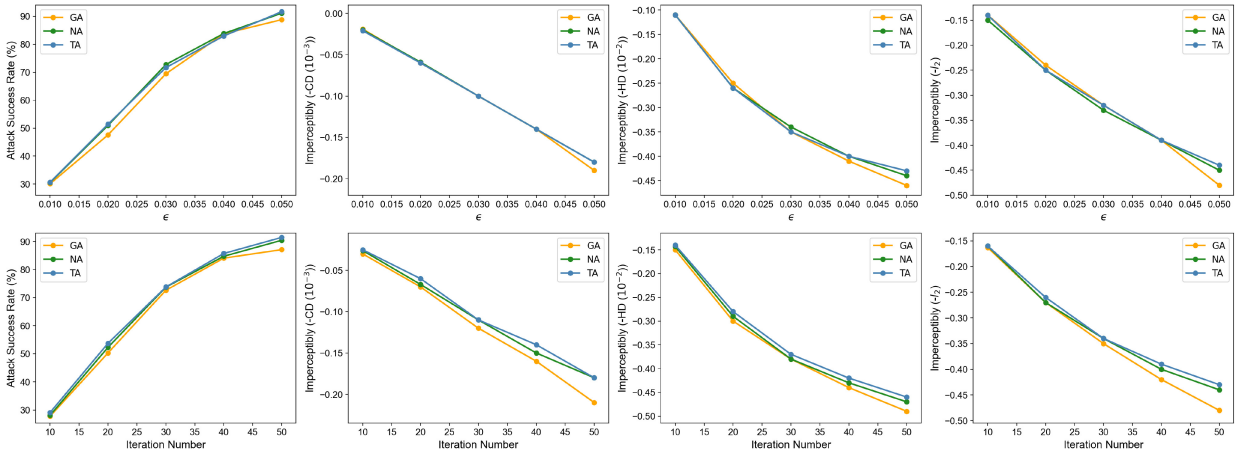


Fig. 5. Comparison of GA, NA, and TA that perturb more on smaller curvature regions. Top row: ASR and imperceptibility of these three adversarial attack methods with a fixed 80 iteration but different perturbation step sizes. Bottom row: ASR and imperceptibility of these three adversarial attack methods with a fixed perturbation step size of 0.05 but different iterations.

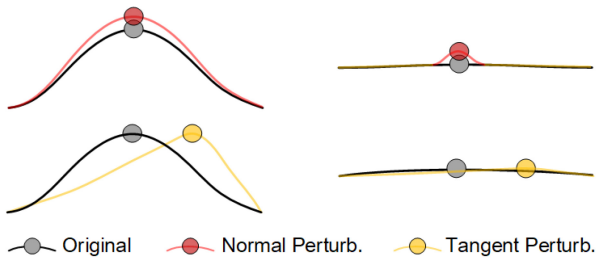


Fig. 6. Comparison on the imperceptibility of applying perturbation at large and small curvature region along ND and TD.

large curvature. Differently, TA obtains its imperceptibility by implementing the attacks via rotating the object shapes. Therefore, the rotation is more difficult to be aware at regions with small curvature. Please refer to Fig. 6 for demonstration.

V. Attacks Combining Normal and Tangent Directional Perturbation

In this section, we introduce the hybrid NTA framework that combines normal and tangent directional perturbation to leverage both of their benefits. We will first describe the directional controlling module and then the whole framework. Please refer to Fig. 7 for demonstration.

Directional Controlling Module: Suppose $\hat{\mathcal{P}}$ is the adversarial point cloud generated from $\mathcal{P}$, and $\hat{p}_i$ denote the corresponding point of p_i in $\hat{\mathcal{P}}$, $\overrightarrow{p_i\hat{p}_i}$ denote the vector from p_i to $\hat{p}_i$, the projected perturbation size in the ND can be calculated as follows:

$$D_{\text{tangent}}(p_i, \hat{p}_i) = \sqrt{||\overrightarrow{p_i\hat{p}_i}||^2 - \left(\frac{\overrightarrow{p_i\hat{p}_i} \cdot n_i}{||n_i||}\right)^2}. \tag{8}$$

Similarly, the projected perturbation size in the TD can be calculated via

$$D_{\text{normal}}(p_i, \hat{p}_i) = \left|\frac{\overrightarrow{p_i\hat{p}_i} \cdot n_i}{||n_i||}\right|. \tag{9}$$

Therefore, by enforcing the value of D_{tangent} or D_{normal} to be small, the perturbation is concentrated along the normal or TDs.

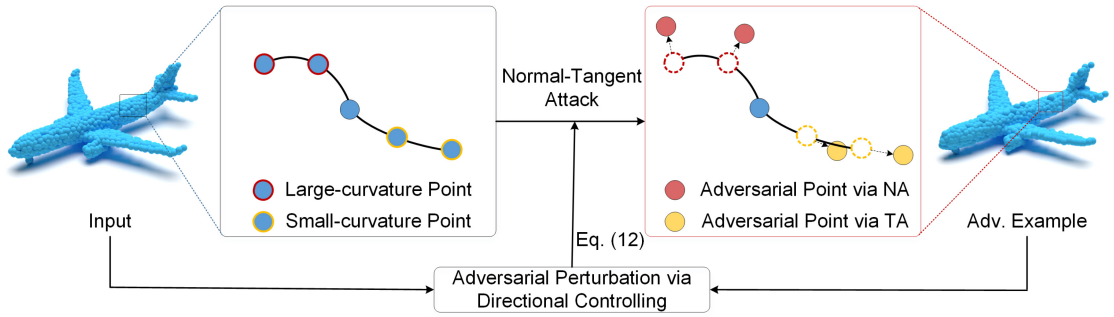


Fig. 7. Demonstration of the NTA framework.

To better exploit the benefits of attacking along ND and TD, we choose the perturbation direction according to the curvature adaptively via

$$D_{\text{dir}}\left(\mathcal{P}, \hat{\mathcal{P}}\right) = \sum_{i=1}^{n} w_l(i) D_{\text{tangent}}\left(p_i, \hat{p}_i\right) + \sum_{i=1}^{n} w_s(i) D_{\text{normal}}\left(p_i, \hat{p}_i\right). \quad (10)$$

In this way, perturbation is guided to be concentrated along the ND at regions with large curvature, and along the TD, otherwise.

NTA Framework: Given the clean point cloud $\mathcal{P}$, our NTA framework first randomly initializes the perturbation to form the adversarial point cloud $\mathcal{P}_1^{\text{adv}}$, and then optimize it iteratively, i.e., $\mathcal{P}_N^{\text{adv}}$.

Specifically, the objective loss function of our NTA framework is defined as follows:

$$\mathcal{J}\left(F, \mathcal{P}_N^{\text{adv}}\right) = -L_{\text{class}}\left(F\left(\mathcal{P}_N^{\text{adv}}\right)\right) + \lambda_1 D_{\text{dir}}\left(\mathcal{P}, \hat{\mathcal{P}}\right) + \lambda_2 D_h\left(\mathcal{P}, P_N^{\text{adv}}\right) + \lambda_3 D_c\left(\mathcal{P}, \mathcal{P}_N^{\text{adv}}\right) \quad (11)$$

where L_{class} is the cross-entropy loss for category classification, D_c is the CD loss, D_h is the HD loss, and λ_1, λ_2, and λ_3 are weighting parameters. By applying gradient descent iteratively, the adversarial point clouds can be refined via

$$\mathcal{P}_{N+1}^{\text{adv}} = \mathcal{P}_N^{\text{adv}} - \mathbf{w} \cdot \epsilon \cdot \text{sign}\left(\nabla_{\mathcal{P}} \mathcal{J}\left(F, \mathcal{P}_N^{\text{adv}}\right)\right) \quad (12)$$

where ϵ is the step size and $\mathbf{w}$ is a vector with element $\mathbf{w}[i] = \max(w_s(i), w_l(i))$. Note that, we additionally adopt $\mathbf{w}$ to suppress the perturbation at regions whose curvatures are not large or small enough.

VI. Experimental Results

A. Implementation

We implement the NTA framework and reproduce all the DNN models with PyTorch, and report the results on a workstation with an Intel Xeon Gold 5218R CPU@2.10 GHz and 64 GB of memory using a single RTX 2080Ti GPU. For the objective loss function of our NTA framework, i.e., (11), we set the weighting parameters with $\lambda_1 = 1.0$, $\lambda_2 = 0.1$, and $\lambda_3 = 1.0$. For the curvature at each point, we estimate it using a 12-nearest neighborhood, and adopt a temperature scaling parameter $t = 20$ to normalize it. For the perturbation step size ϵ, we set it as 0.05.

B. Experimental Setup

Datasets: We adopt two publicly available point cloud data sets, i.e., ModelNet40 [41] and ShapeNet Part [42], to evaluate the performance of adversarial attacks. The ModelNet40 data set contains 12 311 CAD models in 40 most common object classes, with 9843 models for training and another 2468 for testing. The ShapeNet Part data set contains 16 681 point clouds selected from 16 144 categories of the ShapeNetCore, with 14 007 point clouds for training and 2874 for testing. For both data sets, we uniformly sample 1024 points on each shape following [20].

DNN Classifiers: We choose three representative 3-D point cloud classification models with different architectures to evaluate the performance of adversarial attacks, i.e., MLP-based PointNet++ [6], graph-based DGCNN [7], and kernel-based PointConv [21]. All these DNN classifiers are trained on the training data following their original papers.

Baseline Attack Methods: We choose five state-of-the-art adversarial attack methods on point clouds as baselines, i.e., I-FGSM [43], 3D-ADV [34], GeoA^3 [15], ITA*, and SI-ADV [17]. Note that, ITA* indicates the reimplemented solution of ITA [16] without adding the adversarial transformation model, that is, originally designed for black-box attacks.

Defense Methods: We choose three strong adversarial defense methods to evaluate the robustness of adversarial attacks, i.e., statistical outlier removal (SOR), simple random sampling (SRS), denoiser and upsampler network (DUP-Net) [44]. SRS randomly drops 100 points from the input point clouds, and SOR trims irregular points that violate the mean and standard deviation of the nearest neighbor distances. For DUP-Net, it further applies point cloud upsampling to remap irregular points onto the surface.

Evaluation Metrics: We evaluate the effectiveness of adversarial attacks using the ASR, i.e., the rate of generated adversarial samples that successfully fool the classifiers. Besides, we also evaluate the imperceptibility of adversarial attacks by measuring the perturbation between the original point clouds and their corresponding adversarial examples with three metrics: 1) CD; 2) HD; and 3) l_2-norm distance.

TABLE I
COMPARISON ON THE PERTURBATION SIZES REQUIRED BY DIFFERENT ADVERSARIAL ATTACKS METHODS TO ACHIEVE 100% ASRS

Dataset	Model	Metric	I-FGSM	3D-ADV	GeoA^3	ITA*	SI-ADV	NTA
ModelNet40	PointNet++	CD	0.0004	**0.0003**	0.0064	0.0004	0.0005	**0.0003**
		HD	0.0063	0.0381	0.0357	0.0054	0.0182	**0.0047**
		l_2-norm	3.8798	0.6719	**0.3248**	0.4772	0.6507	0.8991
	DGCNN	CD	0.0007	**0.0005**	0.0176	0.0010	0.0006	**0.0005**
		HD	0.0088	0.0475	0.0402	0.0106	0.0117	**0.0078**
		l_2-norm	0.9650	**0.3326**	0.4933	1.1601	0.9049	0.8356
	PointConv	CD	0.0007	0.0011	**0.0005**	0.0008	0.0007	**0.0005**
		HD	0.0089	0.0077	**0.0037**	0.0095	0.0223	0.0076
		l_2-norm	0.9231	1.1230	2.3029	1.0034	1.0630	**0.7647**
ShapeNet Part	PointNet++	CD	0.0003	0.0004	**0.0001**	**0.0001**	**0.0001**	**0.0001**
		HD	0.0024	0.0018	0.0015	**0.0014**	**0.0014**	**0.0014**
		l_2-norm	0.7549	1.0608	0.5445	0.4567	0.4553	**0.3997**
	DGCNN	CD	0.0002	0.0006	0.0004	0.0002	0.0002	**0.0001**
		HD	0.0027	0.0023	0.0024	0.0021	0.0023	**0.0016**
		l_2-norm	0.7168	1.5966	0.7213	0.5933	0.6133	**0.4315**
	PointConv	CD	0.0004	0.0010	**0.0002**	0.0003	0.0003	**0.0002**
		HD	0.0049	0.0039	0.0041	0.0035	0.0041	**0.0027**
		l_2-norm	0.9806	2.0175	0.7021	0.7663	0.7696	**0.5843**

TABLE II
COMPARISON ON THE AVERAGE TIME (SECOND) REQUIRED BY DIFFERENT METHODS TO ATTACK THE DNN CLASSIFIERS TRAINED ON MODELNET40

Model	I-FGSM	3D-ADV	GeoA^3	ITA*	SI-ADV	NTA
PointNet++	**4.51**	4.56	147.11	4.92	6.62	10.09
DGCNN	**0.44**	0.90	15.52	0.84	1.08	1.06
PointConv	**6.06**	6.47	166.76	6.40	9.56	12.81

TABLE III
ASRS (%) OF DIFFERENT METHODS WITH AND WITHOUT APPLYING DEFENSE METHODS FOR ATTACKING POINTNET++

Dataset	Method	I-FGSM	3D-ADV	GeoA^3	ITA*	SI-ADV	NTA
MN	No Defense	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
	SRS	61.53	22.53	67.61	73.74	61.75	**75.40**
	SOR	74.50	17.19	62.47	84.46	73.40	**87.03**
	DUP-Net	22.20	5.44	22.63	22.17	22.52	**27.97**
SN	No Defense	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
	SRS	33.86	32.09	24.72	37.49	37.69	**41.02**
	SOR	56.49	65.63	24.62	83.49	84.37	**88.68**
	DUP-Net	50.57	48.18	**83.49**	78.41	74.87	81.11

C. Performance Comparison

Quantitative Results on Imperceptibility: For fair comparisons, we report three imperceptibility metrics of NTA and the state-of-the-art adversarial attack solutions when they reach the highest ASRs they can achieve. The results reported in Table I show that NTA requires the smallest CD and HD distances, and a medium l_2-norm distance to achieve 100% ASR on attacking PointNet++, DGCNN, and PointConv, outperforming the state-of-the-art methods significantly. Therefore, we conclude that our proposed NTA framework is imperceptible.

Efficiency: We evaluate the efficiency of different adversarial attack methods by measuring the average time they required to attack the DNN classifiers trained on ModelNet40. The results reported in Table II show that I-FGSM is the most efficient, and our method is comparable with most attack solutions. In particular, although GeoA^3 reveals its imperceptibility on some cases as shown in Table I, it requires significantly more time to attack DNN classifiers.

Visualization: To better demonstrate the superiority of our NTA framework in imperceptibility, we visualize the generated adversarial point clouds by different adversarial attack solutions when they reach 100% ASRs in Fig. 8. It could be seen that the adversarial point clouds generated by NTA contain fewer visible outliers, further validating the effectiveness of NTA in conducting imperceptible adversarial attacks.

Attack Performance against Defense: To evaluate the attack performance of our NTA framework against defense, we compare it with I-FGSM, 3D-ADV, GeoA^3, ITA*, and SI-ADV on attacking PointNet++ with applying three strong adversarial defense methods, i.e., SRS, SOR, and DUP-Net. The results reported in Table III show that the ASRs of all attack methods including our NTA framework drop after applying any one of the three defense strategies. In particular, I-FGSM, GeoA^3, ITA*, SI-ADV, and our NTA can still obtain more than 60% success rates after applying SRS and SOR, validating their effectiveness in handling traditional geometric defense methods. However, the performance of all the adversarial attack solutions deteriorate significantly when applying the strong DUP-Net that exploit DNN techniques. It is worth mentioning that our NTA framework achieves the strongest robustness to adversarial defense both on the ModelNet40 data set and on the ShapeNet Part data set.

To deeply understand why our NTA framework has the largest robustness to adversarial defense compared with other adversarial attack methods, we visualize the generated

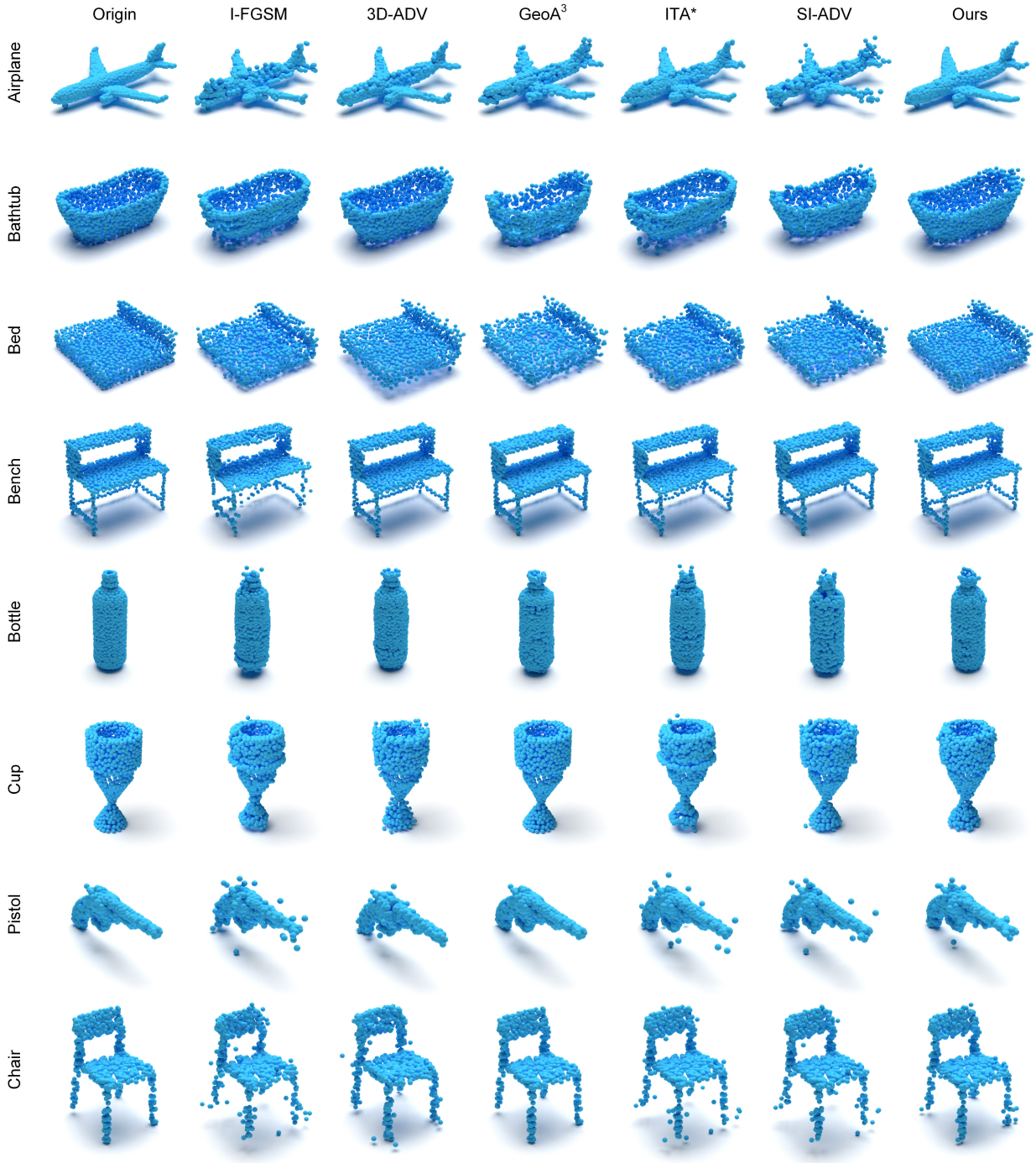


Fig. 8. Visualization of original and adversarial point clouds generated by different adversarial attack methods for attacking PointNet++.

adversarial point clouds before and after applying these three defense methods in Fig. 9. It could be seen that most outliers intentionally generated by adversarial attack methods for fooling DNN models are filtered out, and thus lead to the performance drops on attacking success rates. Differently, our NTA framework achieves imperceptible adversarial attacks, i.e., does not bring clearly visible outliers, and thus is robust to outlier removal.

Transferability: We further investigate the transferability of the NTA framework compared with the state-of-the-art adversarial attack methods, e.g., I-FGSM, 3D-ADV, GeoA^3, ITA*, and SI-ADV. Specifically, we generate adversarial point clouds originally to attack one DNN classifier, and then use them to attack another classifers, e.g., PointNet++, PointConv, and DGCNN. The results reported in Table IV show that our NTA framework has the strongest transferability, especially, from DGCNN to other models.

D. Other Analysis

The Tradeoff Between Attack And Imperceptibility: To further demonstrate the superiority of NTA to NA and TA, we

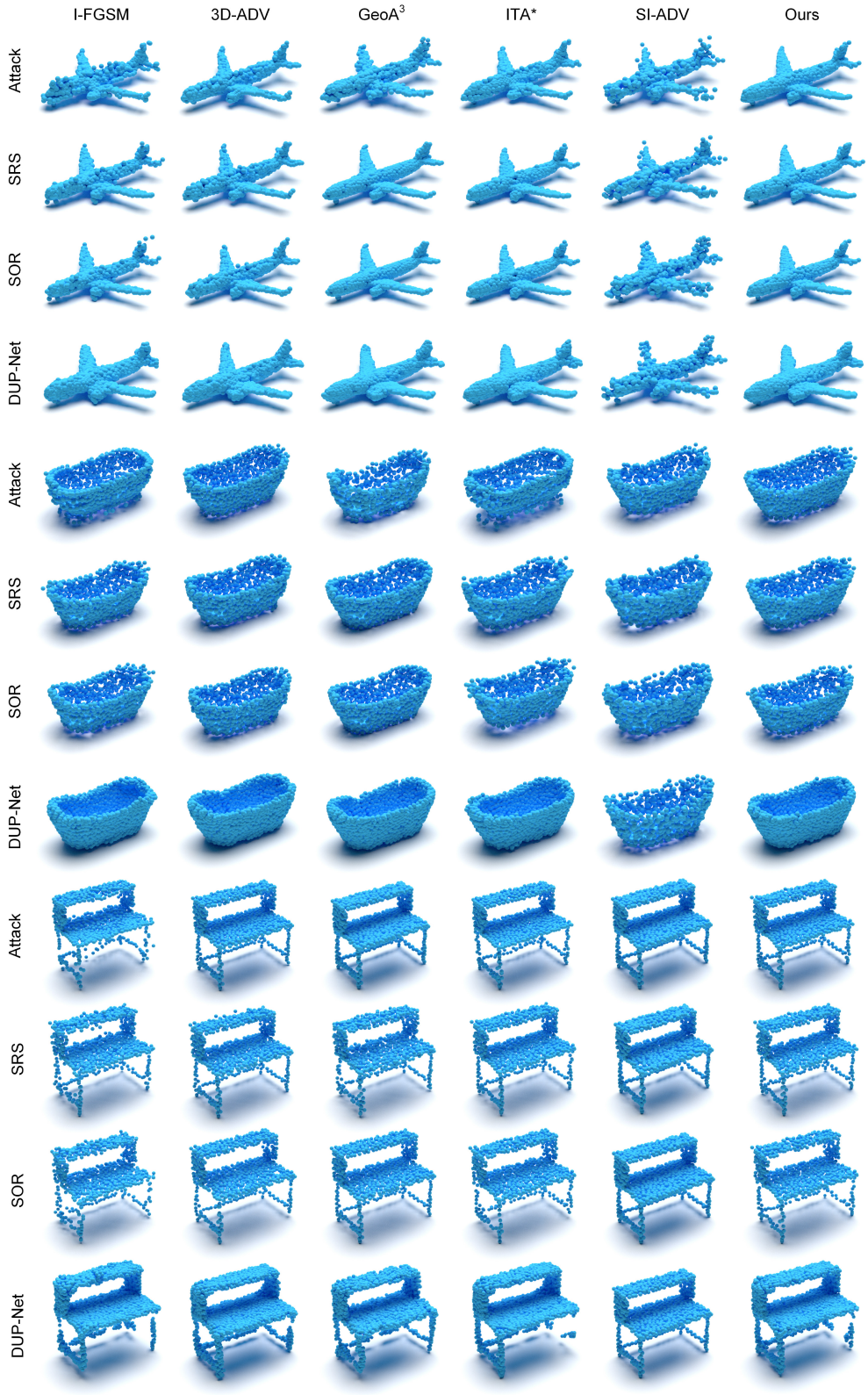


Fig. 9. Visualization of generated adversarial point clouds for attacking PointNet++ by different adversarial attack methods after applying different adversarial defense solutions.

report the ASR and imperceptibility of all these three directional attack solutions at different iterations. The results drawn in Fig. 10 show that NTA reveals much better tradeoff between ASR and imperceptibility, validating its superiority.

The Effectiveness in Simulating NA and TA: To validate that applying a projection restriction on the normal and TDs can simulate ND and TD, we count the angles between the perturbation direction and normal. The results reported in Fig. 11

TABLE IV
COMPARISON ON THE TRANSFERABILITY OF DIFFERENT ADVERSARIAL ATTACK METHODS MEASURED BY ASRS (%)

Dataset	Source	Target	Attack Method					
			I-FGSM	3D-ADV	GeoA^3	ITA*	SI-ADV	NTA
ModelNet40	PointNet++	DGCNN	**24.40**	20.44	18.35	21.42	18.02	21.31
		PointConv	22.40	19.56	22.42	21.86	12.31	**22.86**
	DGCNN	PointNet++	32.53	32.75	21.65	33.18	11.86	**40.87**
		PointConv	32.42	33.96	30.55	30.98	11.09	**39.01**
	PointConv	PointNet++	38.02	22.86	17.14	34.72	34.50	**38.13**
		DGCNN	33.41	**54.73**	23.19	31.75	31.64	36.37
ShapeNet Part	PointNet++	DGCNN	36.55	21.08	17.45	**44.13**	42.68	40.81
		PointConv	11.83	**22.85**	7.99	11.83	10.70	12.36
	DGCNN	PointNet++	30.52	30.02	24.92	38.21	35.82	**38.52**
		PointConv	13.18	17.24	16.87	15.16	14.53	**19.31**
	PointConv	PointNet++	50.25	54.73	25.03	62.40	61.26	**66.67**
		DGCNN	60.53	36.45	17.45	74.24	72.27	**75.08**

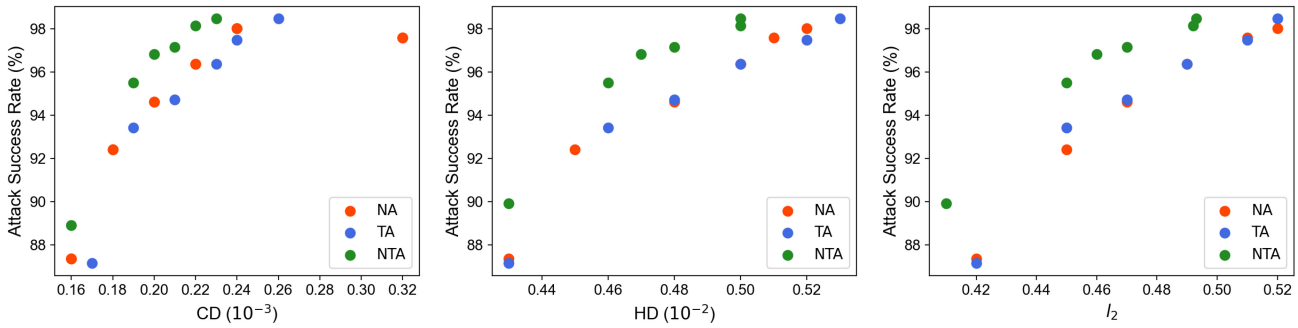


Fig. 10. Tradeoff between ASR and imperceptibility of NA, TA, and our NTA for attacking PointNet++.

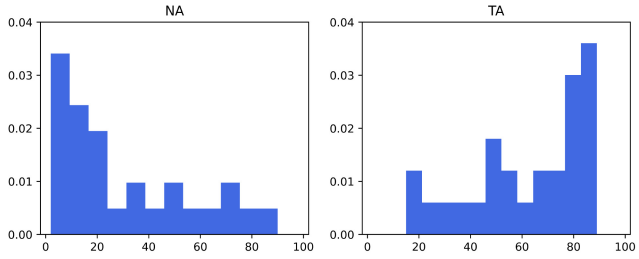


Fig. 11. Histogram of the angles between the perturbation direction of NA/TA and normal.

show that the angles of the perturbation direction of ND and normal are mostly smaller than 20 degrees, while that between TD and normal are mostly larger than 70 degrees, validating the effectiveness of our simulation.

VII. CONCLUSION

In this article, we have extensively investigated the effectiveness of adversarial attacks on point clouds with directional perturbation, and their applicable scenarios. We have further devised an NTA framework that hybrids the above two directional perturbations based on the curvature of the local region around the points to be attacked adaptively. Extensive experiments validated the effectiveness of NTA and its superiority. We hope our work can inspire more research on utilizing geometric properties to improve adversarial attacks on point clouds.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.
[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.
[4] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, 2017, pp. 4700–4708.
[5] K. Tang et al., "Decision fusion networks for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Aug. 11, 2022, doi: 10.1109/TNNLS.2022.3196129.
[6] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. NeurIPS*, vol. 30, 2017, pp. 5099–5108.
[7] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, 2019.
[8] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3D point clouds: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 12, pp. 4338–4364, Dec. 2021.
[9] C. Szegedy et al., "Intriguing properties of neural networks," in *Proc. ICLR*, 2014, pp. 1–9.
[10] J. Levinson et al., "Towards fully autonomous driving: Systems and algorithms," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2011, pp. 163–168.
[11] N. Lin et al., "Manipulation planning from demonstration via goal-conditioned prior action primitive decomposition and alignment," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1387–1394, Apr. 2022.
[12] J. Yang, Q. Zhang, R. Fang, B. Ni, J. Liu, and Q. Tian, "Adversarial attack and defense on point sets," 2019, *arXiv:1902.10899*.
[13] T. Zheng, C. Chen, J. Yuan, B. Li, and K. Ren, "PointCloud saliency maps," in *Proc. ICCV*, 2019, pp. 1598–1606.
[14] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. ICLR*, 2015, pp. 1–17.

[15] Y. Wen, J. Lin, K. Chen, C. L. P. Chen, and K. Jia, "Geometry-aware generation of adversarial point clouds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 2984–2999, Jun. 2022.
[16] D. Liu and W. Hu, "Imperceptible transfer attack and defense on 3D point cloud classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jul. 25, 2022, doi: 10.1109/TPAMI.2022.3193449.
[17] Q. Huang, X. Dong, D. Chen, H. Zhou, W. Zhang, and N. Yu, "Shape-invariant 3D adversarial point clouds," in *Proc. CVPR*, 2022, pp. 15314–15323.
[18] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. ICCV*, 2015, pp. 945–953.
[19] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. CVPR*, 2018, pp. 4490–4499.
[20] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. CVPR*, 2017, pp. 652–660.
[21] W. Wu, Z. Qi, and L. Fuxin, "PointConv: Deep convolutional networks on 3D point clouds," in *Proc. CVPR*, 2019, pp. 9621–9630.
[22] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on X-transformed points," in *Proc. NeurIPS*, vol. 31, 2018, pp. 820–830.
[23] Z. Liu, H. Tang, Y. Lin, and S. Han, "Point-voxel CNN for efficient 3D deep learning," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 965–975.
[24] Y. Chen, W. Peng, K. Tang, A. Khan, G. Wei, and M. Fang, "PyraPVConv: Efficient 3D point cloud perception with pyramid voxel convolution and sharable attention," *Comput. Intell. Neurosci.*, vol. 2022, May 2022, Art. no. 2286818.
[25] K. Tang et al., "RepPVConv: Attentively fusing reparameterized voxel features for efficient 3D point cloud perception," *Vis. Comput.*, to be published. [Online]. Available: https://doi.org/10.1007/s00371-022-02682-0
[26] A. Ioannidou, E. Chatzilari, S. Nikolopoulos, and I. Kompatsiaris, "Deep learning advances in computer vision with 3D data: A survey," *ACM Comput. Surv.*, vol. 50, no. 2, pp. 1–38, 2018.
[27] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: Going beyond euclidean data," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, Jul. 2017.
[28] Y. Dong et al., "Boosting adversarial attacks with momentum," in *Proc. CVPR*, 2018, pp. 9185–9193.
[29] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Security Privacy*, 2017, pp. 39–57.
[30] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. CVPR*, 2016, pp. 2574–2582.
[31] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346–360, 2020.
[32] A. Chakraborty, M. Alam, V. Dey, A. Chattopadhyay, and D. Mukhopadhyay, "A survey on adversarial attacks and defences," *CAAI Trans. Intell. Technol.*, vol. 6, no. 1, pp. 25–45, 2021.
[33] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial examples: Attacks and defenses for deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2805–2824, Sep. 2019.
[34] C. Xiang, C. R. Qi, and B. Li, "Generating 3D adversarial point clouds," in *Proc. CVPR*, 2019, pp. 9136–9144.
[35] M. Wicker and M. Kwiatkowska, "Robustness of 3D deep learning in an adversarial setting," in *Proc. CVPR*, 2019, pp. 11767–11775.
[36] D. Liu, R. Yu, and H. Su, "Extending adversarial attacks and defenses to deep 3D point cloud classifiers," in *Proc. ICIP*, 2019, pp. 2279–2283.
[37] K. Lee, Z. Chen, X. Yan, R. Urtasun, and E. Yumer, "ShapeAdv: Generating shape-aware adversarial 3D point clouds," 2020, *arXiv:2005.11626*.
[38] J. Kim, B.-S. Hua, T. Nguyen, and S.-K. Yeung, "Minimal adversarial examples for deep learning on 3D point clouds," in *Proc. ICCV*, 2021, pp. 7797–7806.
[39] T. Tsai, K. Yang, T.-Y. Ho, and Y. Jin, "Robust adversarial objects against deep learning models," in *Proc. AAAI*, vol. 34, 2020, pp. 954–962.
[40] K. Tang et al., "NormalAttack: Curvature-aware shape deformation along normals for imperceptible point cloud attack," *Security Commun. Netw.*, vol. 2022, Aug. 2022, Art. no. 1186633.
[41] Z. Wu et al., "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. CVPR*, 2015, pp. 1912–1920.
[42] A. X. Chang et al., "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*.
[43] X. Dong, D. Chen, H. Zhou, G. Hua, W. Zhang, and N. Yu, "Self-robust 3D point recognition via gather-vector guidance," in *Proc. CVPR*, 2020, pp. 11513–11521.
[44] H. Zhou, K. Chen, W. Zhang, H. Fang, W. Zhou, and N. Yu, "DUP-Net: Denoiser and upsampler network for 3D adversarial point clouds defense," in *Proc. ICCV*, 2019, pp. 1961–1970.

Keke Tang (Member, IEEE) received the B.Eng. degree from Jilin University, Changchun, China, in 2012, and the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 2017.

He is currently an Associate Professor with Guangzhou University, Guangzhou, China. Prior to joining Guangzhou University in 2019, he was a Postdoctoral Fellow with The University of Hong Kong, Hong Kong. His research interests fall into the areas of robotics, computer vision, computer graphics, and cyberspace security. His personal website is https://tangbohu.github.io/.

Yawen Shi received the B.S. degree in software engineering from Zhongyuan University of Technology, Zhengzhou, China, in 2020. She is currently pursuing the M.S. degree in computer technology with Guangzhou University, Guangzhou, China.

Her research focuses on computer vision and and AI security.

Tianrui Lou received the B.S. degree in software engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2020. He is currently pursuing the M.S. degree in computer technology with Guangzhou University, Guangzhou, China.

His research focuses on AI security and especially adversarial security of deep neural networks.

Weilong Peng received the Ph.D. degree in computer application technology from Tianjin University, Tianjin, China, in 2017.

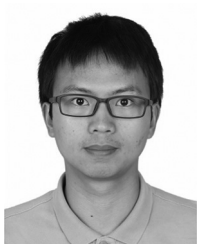
He is currently a Lecturer with Guangzhou University, Guangzhou, China. His research interests lie in artificial intelligence and computer vision.

Xu He received the B.S. degree in software engineering from Liaoning Technical University, Fuxin, China, in 2020. He is currently pursuing the master's degree majoring in cyberspace security with Guangzhou University, Guangzhou, China.

His current research interests mainly include computer vision, AI security, and 3-D point cloud attacks and defense.

Peican Zhu (Member, IEEE) received the Ph.D. degree from the University of Alberta, Edmonton, AB, Canada, in 2015.

He is currently an Associate Professor with the School of Artificial Intelligence, Optics and Electronics (iOPEN), NWPU, Xi'an, China. His research interests include data-driven complex systems modeling, complex social networks analysis, artificial intelligence, and system security.



Zhaoquan Gu (Member, IEEE) received the bachelor's and Ph.D. degrees in computer science from Tsinghua University, Beijing, China, in 2011 and 2015, respectively.

He is currently a Professor with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China, and a Researcher with the Department of New Networks, Peng Cheng Laboratory, Shenzhen. His research interests include wireless networks, distributed computing, big data analysis, and artificial intelligence security.

Zhihong Tian (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science and technology from Harbin Institute of Technology, Harbin, China, in 2001, 2003, and 2006, respectively.

He is currently a Professor and the Dean of the Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, Guangdong Province, China. He is also a part-time Professor with Carlton University, Ottawa, ON, Canada. Previously, he served in different academic and administrative positions with the Harbin Institute of Technology. He has authored over 200 journal and conference papers. His research has been supported in part by the National Natural Science Foundation of China, National Key research and Development Plan of China, National High-tech Research and Development Program of China (863 Program). His research interests include computer networks and cyberspace security.

Dr. Tian is honored as the Pearl River Scholar in Guangdong Province. He served as a member and the Chair, and the general chair of a number of international conferences. He is a Distinguished Member of the China Computer Federation.