Good Can Sometimes be Bad: A Unified Attack against 3D Point Cloud Classifier by a Flexible Isotropic Resampling

Supplementary Material

6. More Details of Related Works

Adversarial attack is firstly developed in 2D image domain. Xiang *et al.* extend such attack to 3D point cloud by introducing a 3D shape constraint [45]. In the following development, points adding [45], points dropping [57] and points perturbing [16, 35, 55, 56] have become the mainstream methods. In detail, Zhao *et al.* conduct adversarial perturbation using a simple rotation rather than points perturbation. Its main aim is to preserve the isometric feature of attacked 3D point cloud [55]. To limit the amplitude of the adversarial disturbance, Kim *et al.* aim to detect the minimal points subset rather than all the points [21]. Tang *et al.* constrain lattice-based barycentric coordinates during the attack process from a local parametric perspective to ensure the imperceptibility [35]. To enhance the imperceptibility of adversarial attacks, Zhang *et al.* propose the curvature-invariant method, which directly regularizes the back-propagated gradient during the generation of adversarial point clouds based on two assumptions [53].

Backdoor attack misleads 3D DNN by polluting the training dataset in the training stage. Its core is to design an effective backdoor trigger. Point addition attacks conduct attack by building mapping from a certain geometrical pattern to the target class. Specifically, PointPBA-I [23] regards a points set with a certain shape and location as the backdoor trigger. PCBA [47] optimizes the shape and location of a points set as the trigger for higher efficiency. PointNCBW [40] explore 3D point cloud backdoor attack under clean label scenario. It is for 3D point cloud dataset ownership verification. Besides, shape transformation attacks implant backdoor into point cloud by 3D transformation such as rotation, scaling, and affine. In particular, IRBA [10] processes point cloud by the designed weighted local transformation. PointPBA-O [23] and NRBdoor [8] propose to implant backdoor trigger by conducting clean rotation and noisy rotation, respectively. Instead of designing backdoor trigger manually, iBA proposes a novel solution utilizing a learnable auto-encoder [4].

7. More Details of Experiments

7.1. Experiments Setting

Datasets and victim 3D DNNs. We select ModelNet40 [44], ShapeNet16 [49] and ScanObjectNN [37] as the evaluating datasets. They are wildly used to measure the performance of newly designed 3D DNN. In particular, there are 12311 models for 40 categories in ModelNet40, where 9843 models are utilized for training and 2468 models for testing. The ShapeNet16 is with 16 categories which contains 12128 and 2874 objects for training and testing, respectively. The ScanObjectNN comprises 15 categories which contains 11471 and 2893 objects for training and testing, respectively. For easy of comparison, we uniformly sample 1024 points from the surface of each object and re-scale them into a unit cube.

Attack Setting of Our UAtt3D. For backdoor attack, we select the second label of each dataset as the target label y^t. The poison rate α is set to 0.05. DGCNN is used as the surrogate 3D DNN. $\eta_i = \pi/2, \gamma_i = \pi$ are set to the initial value of the optimization process. All the inference samples are used to test *ASR*. For adversarial attack, the balancing parameter t_s is set to 0.5 for each victim 3D DNN. The angle of sampling rays is selected as $\eta_i = \pi/2, \gamma_i = \pi$. The number of starting points k is set to 16.

Attack Setting of Baseline Backdoor Attacks. We regard most existing SOTA 3D attacks as the benchmarks. The selected backdoor attacks include PointBA [23], PCBA [47], IRBA [10], NRBdoor [8] and IBAPC. Their setting are as follows:

PointBA contains two basic triggers including points set addition(PointBA-I) and rotation(PointBA-O). For PointBA-I, we insert a sphere with radius $r = 0.05$ whose center is $(0.5, 0.5, 0.5)$. For PointBA-O, the point cloud is rotated around the z-axis by 10°.

For NRBdoor, its backdoor trigger is conducted by an affine matrix K which is:

$$K = \begin{bmatrix} 1 & 0.1 & 0.2 \\ 0 & 0.9 & 0 \\ 0.1 & 0 & 1 \end{bmatrix}$$

For IRBA, we set the number of anchors W to 16. In the multi-anchor transformation, the rotation angle α and the scaling factor s are set to 5° and 5, respectively. The bandwidth h in smooth aggregation is set to 0.5.

For PCBA, the source label is selected as 35 for ModelNet40 and 8 for ShapeNet16. The shape and position of backdoor trigger is optimized using its default setting.

For IBAPC, its backdoor graph frequency trigger is learned by the joint training. The balancing parameter is set to 0.5.

Attack Setting of Baseline Adversarial Attacks. The selected adversarial attacks include GeoA3 [41], AdvPC [15], SIA [17], HiT [25], SS-Attack [52] and Eidos [32]. Their setting are as follows:

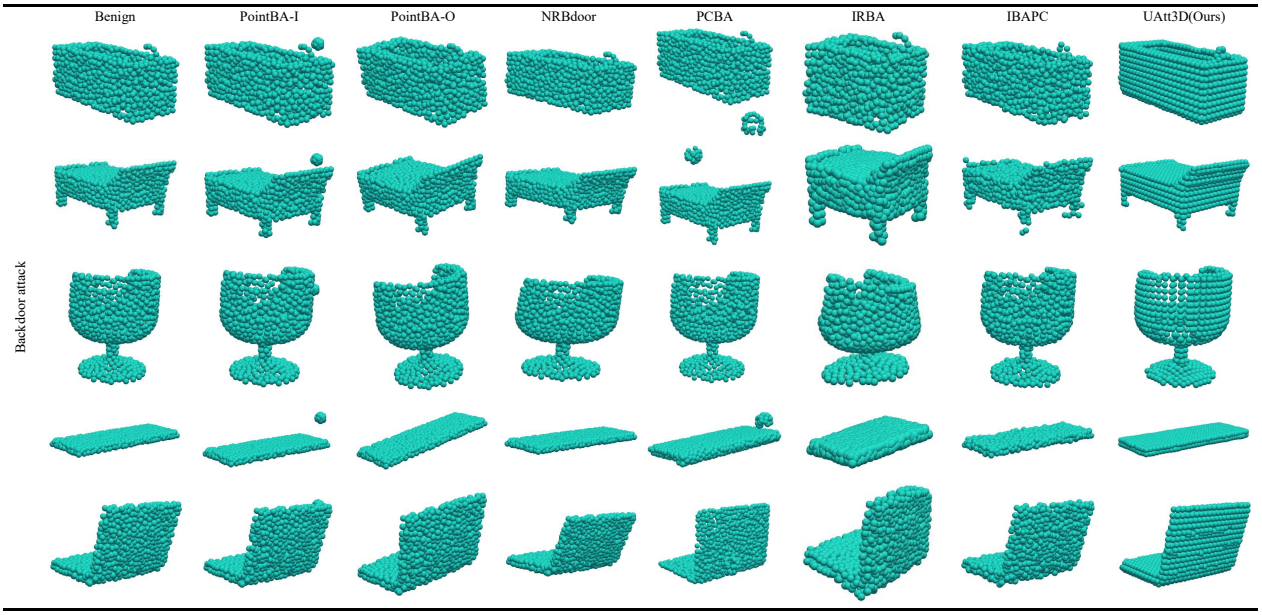

Figure 11. More instance of backdoored 3D point cloud.

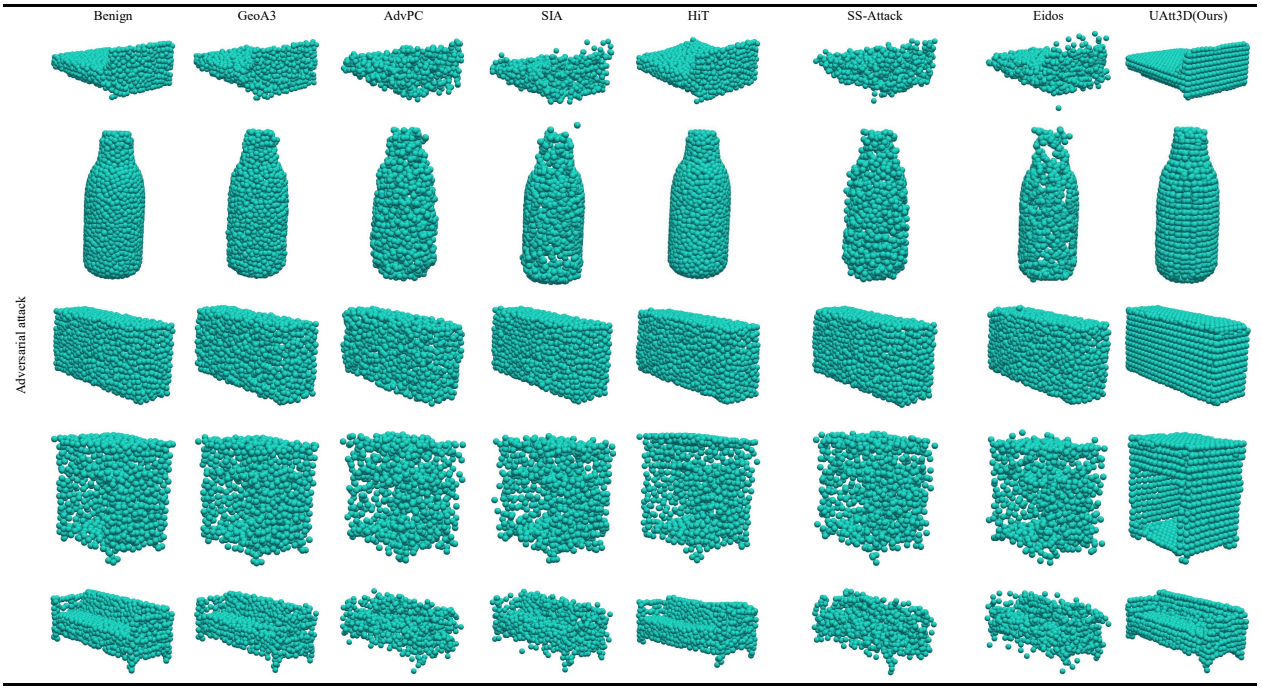

Figure 12. More instance of adversarial 3D point cloud.

For the GeoA3 attack, 10 binary search steps are employed, each with 500 iterations. Under all constraints, the chamfer distance loss weight and curvature loss weight are both set to 1. The attack step size λ is set to 0.02.

In HiT, 20 neighboring points are designated for the computation of uniformity loss, while 16 are assigned for the calculation of curvature loss. The parameter ϵ is configured as 0.55 to constrain the perturbation range. Chamfer distance loss weight is set to 0.0001 and kernel-weight as 1, max-sigm as 1.2 and min-sigm as 0.1.

For the AdvPC attack, perturbations are generated using a trained autoencoder on datasets such as ModelNet40. Two

binary search steps are performed, each with 200 iterations. The perturbation size ϵ is set to 0.2, and the attack step size λ is set to 0.03.

In the Eiods attack, a KNN graph with $k = 10$ is constructed, and 10 binary search runs are performed, each with a maximum of 100 iterations.

The SS-Attack is configured with a hyperparameter γ of 0.25. It employs 180 optimization iterations. The parameter ϵ is set to 0.2, and the attack step size λ is set to 0.03. In addition, AdvPC is used as its basic method.

For SIA, the perturbation ϵ is set to 0.16, and the attack step size λ is set to 0.02.

Evaluating Metrics. In the stealthiness experiment, two metrics are used to quantify the isotropy of 3D point cloud, named Kde Uniformity Variance(KUV) [6] and Cumulative Uniformity Deviation(CUD) [7].

For CUD, it reflects the isotropy I of the point cloud by calculating density deviations under different radii:

$$\rho_i^{(r_j)} = \text{count}\left(\{p_k \mid \|p_k - p_i\| \leq r_j\}\right) \tag{12}$$

$$\rho_{\text{mean}}^{(r_j)} = \frac{1}{n}\sum_{i=1}^{n} \rho_i^{(r_j)} \tag{13}$$

$$I = \frac{1}{m}\frac{1}{n}\sum_{j=1}^{m}\sum_{i=1}^{n}\left(\rho_i^{(r_j)} - \rho_{\text{mean}}^{(r_j)}\right)^2 \tag{14}$$

where $\rho_i^{(r_j)}$ denotes the number of points within radius r_j for each point p_i in the point cloud. p_k means the points contained in its sphere. The mean value of the local densities of all points is denoted as $\rho_{\text{mean}}^{(r_j)}$. n is the total number of points in the point cloud, and m is the number of different radii r_j $(j = 1, 2, \ldots, m)$.

For KUV, it evaluates the isotropy of a point cloud by calculating the degree of its density fluctuation:

$$K_h(\mathbf{u}) = \frac{1}{(2\pi h^2)^{3/2}} \exp\left(-\frac{\|\mathbf{u}\|^2}{2h^2}\right) \tag{15}$$

$$\hat{f}(\mathbf{p}_i) = \frac{1}{n}\sum_{j=1}^{n} K_h(\mathbf{p}_i - \mathbf{p}_j) \tag{16}$$

$$I = \frac{1}{n}\sum_{i=1}^{n} \hat{f}(\mathbf{p}_i) \tag{17}$$

where $K_h(\mathbf{u})$ is the gaussian kernel function, h is a bandwidth parameter and u is an input vector. $\hat{f}(\boldsymbol{p}_i)$ denotes the estimated probability density at the point $\boldsymbol{p}_i$. I is the average kernel density estimates of all sample points, which represent its isotropy.

7.2. Attack Experiments.

Attack Stealthiness. To fully prove the advantage of our proposed UAtt3D in guaranteeing attack stealthiness. More attacked 3D point clouds are shown in Fig 11 and Fig 12. Additional stealthiness evaluation via Chamfer Distance(CD) and Hausdorff Distance(HD) on PointNet++ (MN40 dataset), shown in Table 5, confirms our UAtt3D achieves SOTA geometric preservation by optimizing point distribution rather than damaging surface shape.

Table 5. Stealthiness measured by CD $\times$ 100 $\downarrow$ and HD $\times$ 100 $\downarrow$

	Backdoor attacks					Adversarial attacks				
	PCBA	PointBA-I	IRBA	NRBdoor	UAtt3D	Eidos	SS-Attack	HiT	AdvPC	UAtt3D
HD	10.64	11.16	8.67	4.47	**2.43**	7.23	5.78	21.03	6.11	**2.26**
CD	0.46	0.49	2.09	0.85	**0.40**	2.20	2.11	10.70	2.14	**0.28**

Human Inspection. Fig 14 exhibits ten questions of the human inspection experiment. Each question contains four point cloud snapshots. One snapshots is from point cloud attacked by our UAtt3D. The remaining three snapshots is from benign point cloud or point cloud attacked by PointPBA-I, IRBA, IBAPC (backdoor attacks), HiT, SiA and GeoA3 (adversarial attacks) whose stealthiness is leading in the existing attacks. Each participant is asked to choose one point cloud that he thinks is the best and has the highest quality. The inspection results are shown as Fig 13. For a specific attack, its ratio is calculated by dividing the number of times a attack is selected by the total number of questions that contained that attack. The fractional form of the ratio is shown in Table 6. Since each question does not include all attacks, the sum of their ratio isn't 1.

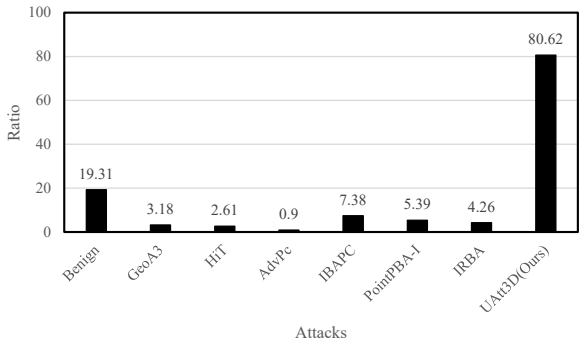

Figure 13. The ratio of attacked 3D point cloud selected as the one with the best quality.

Table 6. The ratio of each attacks selected by the participants. Taking GeoA3 attack as a example, a total of 880 questions included GeoA3, and 28 of them chose GeoA3 as the most high quality one.

Benign	GeoA3	HiT	AdvPc	IBAPC	PointBA-I	IRBA	UAtt3D(Ours)
170/880	28/880	23/880	8/880	78/1056	19/352	15/352	1419/1760

Attack Effectiveness on Defective Sample. To show the effectiveness of our UAtt3D on defective 3D point cloud, we further evaluate it on victim DGCNN whose results are shown in Table 7. More attacked practical 3D point

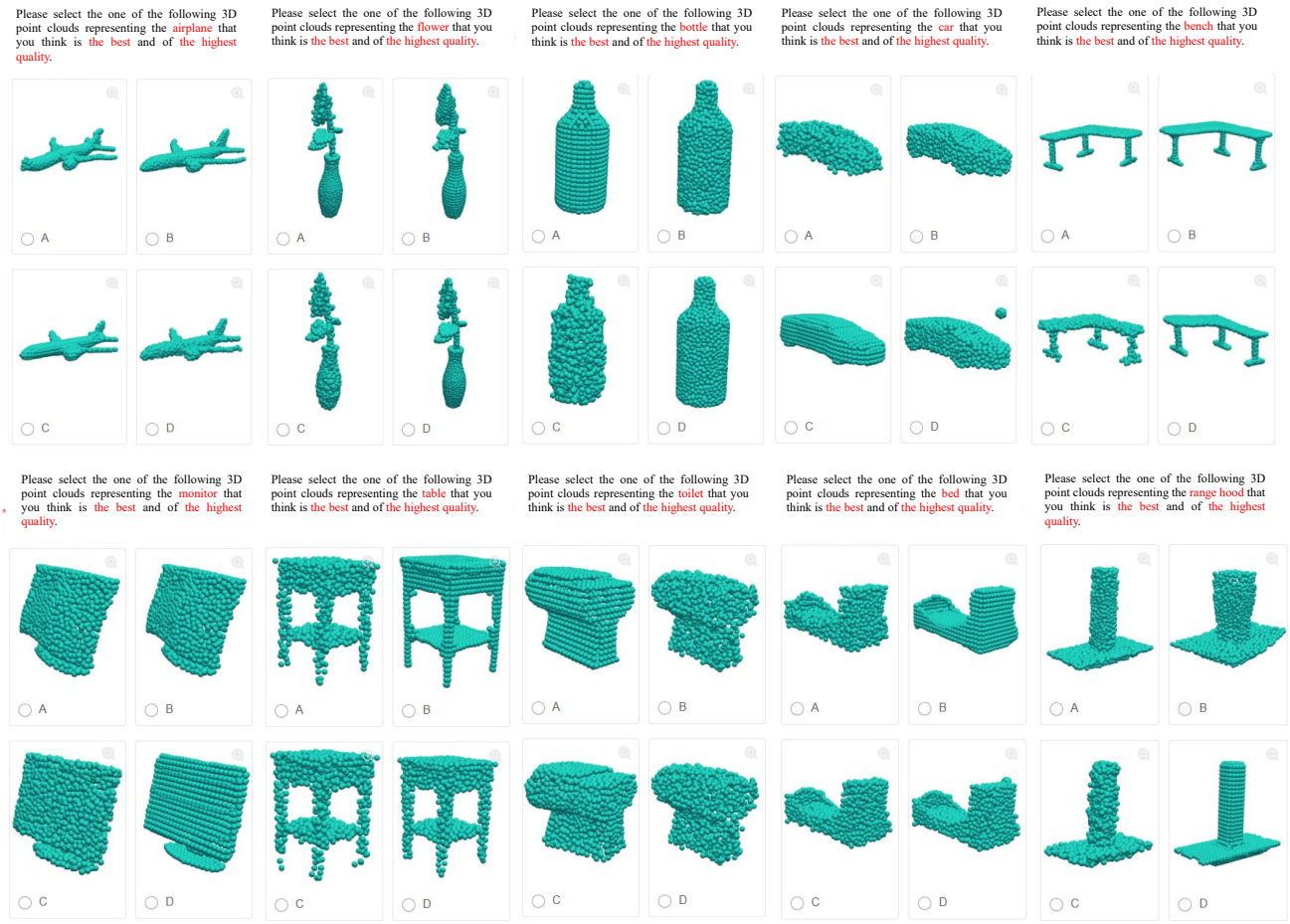

Figure 14. Ten questions of our human inspection experiment. Each question ask participant to select the best point cloud of the highest quality from the group.

Table 7. Performance on backdoor attack and adversarial attack of our UAtt3D with defective 3D point cloud and victim DGCNN.

Dataset ↓	Victim →	Backdoor attack				Adversarial attack		
	Attack ↓	*BAc*	*ASR*	CUD	KUV	*ASR*	CUD	KUV
SON	Benign	81.46	-	365.29	5.66	-	365.29	5.66
	UAtt3D	81.46	91.26	157.29	1.03	99.40	157.82	1.08
MN40_O	Benign	92.03	-	144.06	2.92	-	144.06	2.92
	UAtt3D	91.15	97.66	62.03	0.33	91.08	81.07	0.70
MN40_H	Benign	91.19	-	92.82	2.90	-	92.82	2.90
	UAtt3D	91.51	97.84	64.66	0.46	93.17	75.30	1.10
MN40_S	Benign	91.19	-	36.26	2.78	-	36.26	2.78
	UAtt3D	91.67	99.69	34.47	0.73	96.96	40.99	1.70
MN40_A	Benign	90.50	-	34.38	2.57	-	34.38	2.57
	UAtt3D	91.15	99.83	27.89	0.62	91.33	48.36	2.66

clouds from defective ModelNet40 and ScanObjectNN are shown in Fig 15 and Fig 16 respectively.

7.3. Defense Experiments.

Resistance to Resampling. Our UAtt3D works by arranging points in a point cloud isotropically. A natural question is whether it will fail when the attacked point loses strict isotropy. To answer this, we resample the attacked point cloud and observing the obtained *ASR*. The sampling methods include Farthest Point Sampling (FPS), Voxel Grid Downsampling(VGD)[27], Midpoint Interpolation Upsampling(MIU)[50], Random Sampling(RS)[22]. Results in Table 8 suggest that the decrease in *ASR* is small in most cases. Its because our UAtt3D does not strictly rely on a single point isotropy. It pushes the attacked point cloud away from its own feature space through isotropic resampling. It is difficult to push it back to its original space by simply breaking the isotropy.

Resistance to Adversarial Training. Adversarial training[1] is a traditional and effective defense method. *ASR* of our UAtt3D decreases 9.54% defended by adversarial training. It shows that it is able to resist such defense.

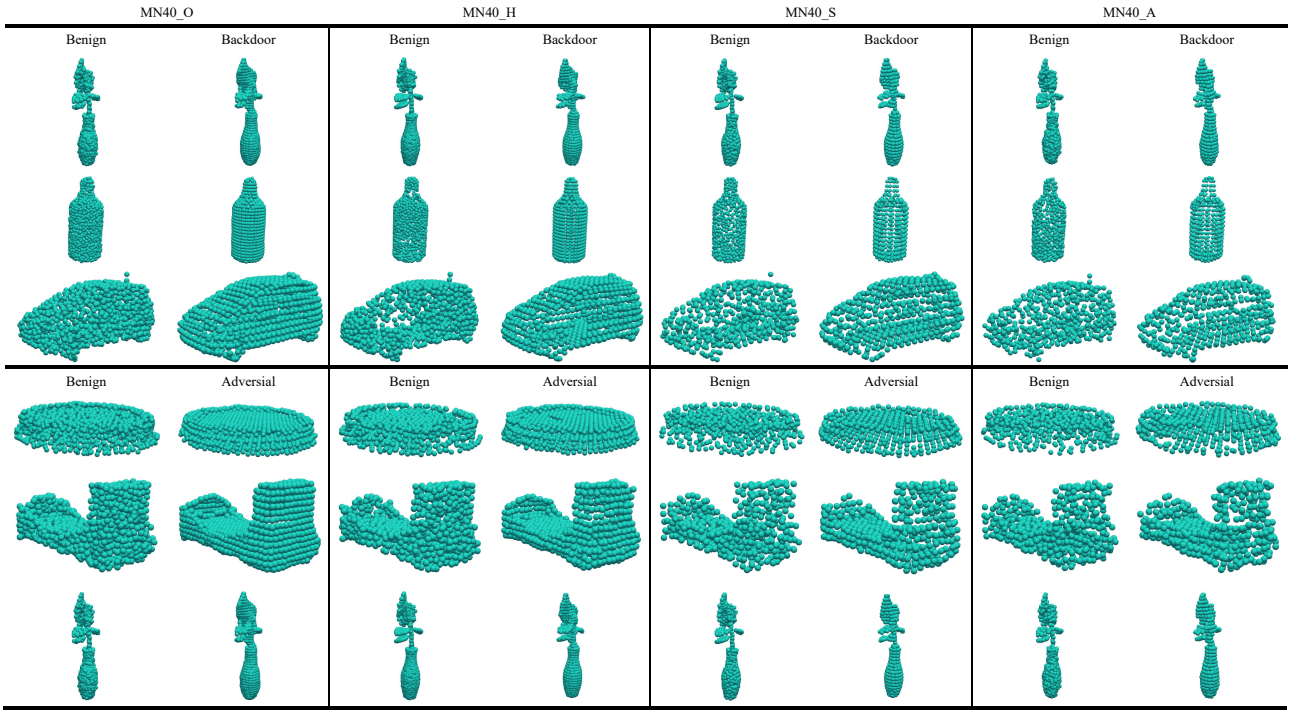

Figure 15. More backdoored point clouds and adversarial point clouds based on defective benign point clouds from MN40_O, MN40_H, MN40_S, and MN40_A datasets.

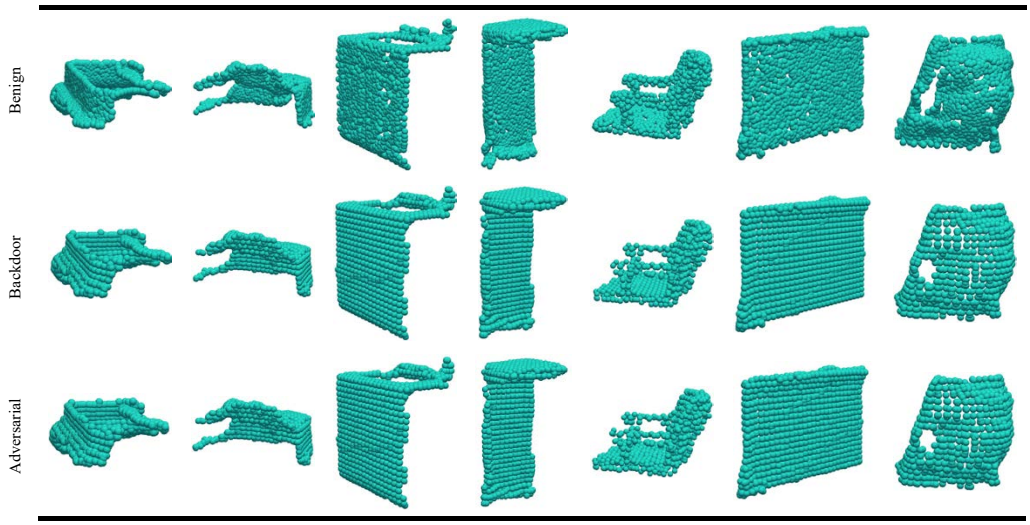

Figure 16. More backdoored point clouds and adversarial point clouds based on practical point clouds from SON dataset.

It is because our attack relies on an isotropic point distribution, which is hard to generated in the max-min process of adversarial training.

Resistance to SOR. Statistical Outlier Removal (SOR) [58] removes the malicious points according to the distribution of point distance. During the evaluation, we vary k of k-nearest neighbors from 2 to 65, and sets standard deviation σ to 1.0. Results comparing with PoinPBA-I shown in Fig 17 suggest that UAtt3D can resist SOR. Specifically, its *ASR* changes slightly according to the variation of k. In contrast, that of PointPBA-I and SIA decreases to 50.51% with DGCNN on MN40 when k = 65.

Resistance to Data Augmentation. Data augmentation is a popular way to increase the robustness of DNN. We

Table 8. Backdoor *ASR* of our UAtt3D resisted by resampling. It suggests that resampling the attacked 3D point cloud cannot disable our UAtt3D.

Resampling	FPS	MIU	RS	VGD
DGCNN	93.47	94.19	95.55	94.49
PointNet++	98.72	66.15	98.50	98.81

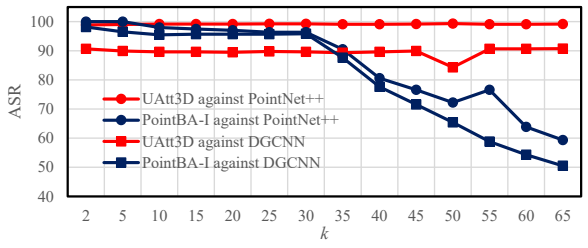

Figure 17. *ASR* of PointBA-I and our UAtt3D under the defense of SOR. UAtt3D achieves the better results.

Table 9. *ASR* of backdoor attack under data augmentation.

	None	Rotation	Jitter	Scaling	Shifting	All
UAtt3D	92.06	91.82	92.14	92.78	92.58	92.42
PointPBA-O	89.83	2.66	93.33	93.00	92.67	2.17

augment the backdoored data by random rotation, jitter, random scaling, and shift. The victim 3D DNN is DGCNN trained on MN40. Results in Table 9 illustrate that UAtt3D can evade data augmentation. Its *ASR* descent is minor when conducting all augmentation at once. In contrast, *ASR* descent of PointPBA-O is 87.66%.

Resistance to SSD. Spectral Signature Defense (SSD) [36] detoxify the victim 3D DNN by detecting and remove the backdoored point clouds in the training dataset. The obtained results illustrate that our UAtt3D still acquire 87.19% *ASR* trained on the defended dataset. Hence, the proposed method can resist SSD.

External Results against STRIP. Fig 18 exhibits the resistance performance against STRIP on victim PointNet++. It suggests that our UAtt3D achieves better results similar as on victim DGCNN.

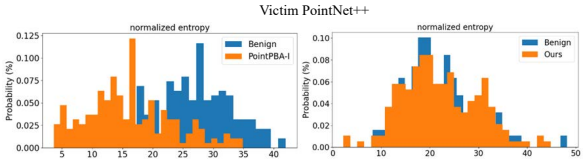

Figure 18. The defense result on STRIP with victim PointNet++. UAtt3D is the better one to resist STRIP.

7.4. Ablation Studies

Initial Sampling Angle (η_i, γ_i). In our attack experiment, we set the initial sampling angle $\eta_i = \pi/2, \gamma_i = \pi$ for convenience. To evaluate its influence, we randomly select five initial sampling angles for conducting backdoor attack and adversarial attack. Results in Fig 19 suggest that ours 3DAtt3D can usually achieve a promising attack effectiveness with the variation of initial sampling angle.

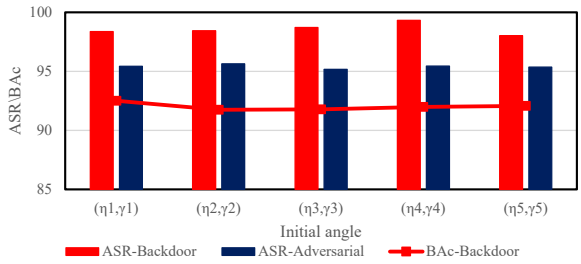

Figure 19. *ASR* of backdoor attack and adversarial attack with different initial sampling angle.

Necessity for the Designed FIR. To evaluate the necessity of the designed FIR for achieving the adversarial target and backdoor target. We compare the geometrical feature preservation of attacked 3D point cloud based on traditional resampling method, including density adaptive sampling [31] and median filter resampling [5]. Results in Fig 20 show that the designed FIR is the best one to preserve geometrical feature under similar *ASR*.

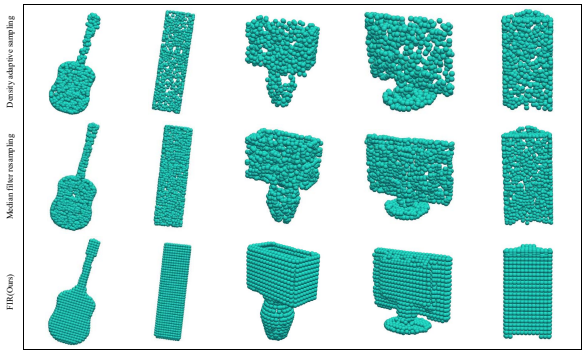

Figure 20. Necessity for the designed FIR. Our FIR achieves the best result in preserving the geometrical feature under similar *ASR*.

Necessity for the Designed Unified Attack. The proposed UAtt3D aims to solve the shortcoming that the backdoor attack and the adversarial attack cannot be transferred to each other. To show the existence of such shortcoming, we conduct backdoor attack using existing adversarial attack, and conduct adversarial attack using existing backdoor attack. Results in Table 10 prove that existing adversarial attack cannot work for backdoor attack, vice versa.

Table 10. Attack effectiveness of existing 3D backdoor attacks and adversarial attacks migrating to each other. For example 23.66% is *ASR* of PointBA-I (a backdoor attack) migrated to adversarial attacks.

Attacks	PointBA-I	PointBA-O	IRBA	IBAPC	HIT	IFGM
ASR	23.66	19.33	9.66	9.20	58.10	59.54

8. Ethical Statement

The main aim of this study is to enhance the understanding of security, specifically regarding 3D point cloud backdoor attacks. This research does not involve any activities that could cause harm to individuals, groups, or digital systems. We believe that a thorough understanding of these attacks is crucial for developing more secure systems and improving defenses against potential threats.