Supplementary Material for "3DHacker: Spectrum-based Decision Boundary Generation for Hard-label 3D Point Cloud Attack"

Yunbo Tao[1*] Daizong Liu[2*] Pan Zhou[1] Yulai Xie[1†] Wei Du[1] Wei Hu[2†]
[1]Hubei Key Laboratory of Distributed System Security,
Hubei Engineering Research Center on Big Data Security,
School of Cyber Science and Engineering, Huazhong University of Science and Technology
[2]Wangxuan Institute of Computer Technology, Peking University
{tyb666, panzhou, ylxie, weidu666}@hust.edu.cn dzliu@stu.pku.edu.cn forhuwei@pku.edu.cn

In the supplementary material, we first implement our attack method on more victim models for attack performance comparison, then we provide corresponding defense comparison to validate the robustness of our attack. After that, we provide more visualization results on the adversarial examples generated by different 3D attackers on different victim models. Finally, we provide more details of our proposed spectrum iterative walking strategy.

1. Attack Performance on More Victim Models

To investigate the effectiveness and generalization-ability of our attack, we perform our 3DHackker on more victim models, *i.e.*, PAConv [6], SimpleView [1], and CurveNet [5]. For comparison, we select the SOTA attack method SI-Adv [2] in both white- and black-box settings. *Note that, our 3DHacker is the first 3D adversarial attack in more challenging hard-label black-box setting, which is much harder to achieve success since it has no information of model details (white-box) and output logits (black-box).* As shown in Table 1, our 3DHacker achieves smaller perturbation sizes than the black-box SI-Adv^b model and achieves very competitive results with the white-box SI-Adv^w model. Overall, our 3DHacker achieves the lowest perturbation D_h in all three victim models, demonstrating the effectiveness of our 3DHacker.

2. Defense on More Victim Models

To evaluate the robustness of our 3DHacker compared to SI-Adv^b [2], we also conduct the defense methods Statistical Outlier Removal (SOR) [10] and Simple Random Sampling (SRS) [7]) on corresponding adversarial examples generated on PAConv [6], SimpleView [1], and CurveNet [5]. As shown in Table 2, (1) As for the defense method SOR, our 3DHacker can achieve a higher attack success rate than SI-Adv^b on all three victim models. (2) As for the SRS defense, our 3DHacker still achieves a better attack performance than SI-Adv^b as we generate the adversarial sample with high similarity to the original one in both geometric topology and local point distributions. (3) Our adversarial samples achieve the lowest perturbations with a much higher attack success rate when attacking the model protected by defenses. Overall, compared to the previous best attack method SI-Adv, our 3DHacker is much more robust to existing defense strategies.

3. More Qualitative Results

To further demonstrate the effectiveness of our method on other point clouds of different object categories, we expand the visualization experiment that provides visualization on adversarial samples generated by our 3DHacker, SI-Adv^w[4] (white box attack) and SI-Adv^b[2] (black box attack) as shown in Figure 1, Figure 2 and Figure 3. It shows that previous white- and black-box attackers easily lead to outlier problems and uneven distributions. Moreover, they require more knowledge of the model details (parameters or output logits) during the generation process of adversarial samples. Compared to them, our hard-label setting only accesses the output label of the model and is harder to achieve successful attack. Even though, as shown in the figures, our 3DHackker can alleviate the outlier point problems and produce more imperceptible adversarial samples.

4. More Details of Spectrum Walking

As mentioned in Section 3.4 of the main paper, in addition to the general coordinate walking, we design a spectrum-wise walking strategy in the boundary-cloud optimization stage to jump out the local optimum for producing a better optimized adversarial point cloud. Here, we first

*Equal contributions. †Corresponding authors.

Table 1. Comparative results on the perturbation sizes of different methods for adversarial point clouds. ***Our setting is harder to attack.***

Setting	Attack	Model Details		PAConv [6]			SimpleView [1]			CurveNet [5]		
		Para.	Logits	D_h	D_c	D_{norm}	D_h	D_c	D_{norm}	D_h	D_c	D_{norm}
White-Box	SI-Adv^w [2]	✓	✓	0.0097	0.0004	0.6920	0.0256	0.0014	2.1522	0.0199	0.0006	0.9803
Black-Box	SI-Adv^b [2]	×	✓	0.0449	0.0004	1.3386	0.0469	0.0010	1.8754	0.0453	0.0004	1.4336
Hard-Label Black-Box	Ours	×	×	0.0046	0.0014	0.9444	0.0136	0.0029	1.6150	0.0125	0.0022	1.2332

Table 2. Resistance of the black-box attacks on defended point cloud models.

Defense	Attack	PAConv [6]			SimpleView [1]			CurveNet [5]		
		ASR(%)	D_h	D_{norm}	ASR(%)	D_h	D_{norm}	ASR(%)	D_h	D_{norm}
SOR [10]	SI-Adv^b [2]	94.4	0.0359	1.9640	95.2	0.0375	3.1333	88.8	0.0351	2.5402
	Ours	95.5	0.0028	0.6744	93.6	0.0083	1.0873	89.2	0.0095	1.1752
Drop(30%)	SI-Adv^b [2]	73.6	0.0402	1.1979	56.8	0.0411	1.2577	71.2	0.0400	1.4630
	Ours	95.2	0.0061	0.8290	91.2	0.0092	0.9638	82.5	0.0157	0.8598
Drop(50%)	SI-Adv^b [2]	84.8	0.0390	0.8537	68.8	0.0368	0.9119	79.2	0.0392	1.1759
	Ours	93.8	0.0136	0.7261	97.6	0.0066	0.7570	83.4	0.0186	0.7558

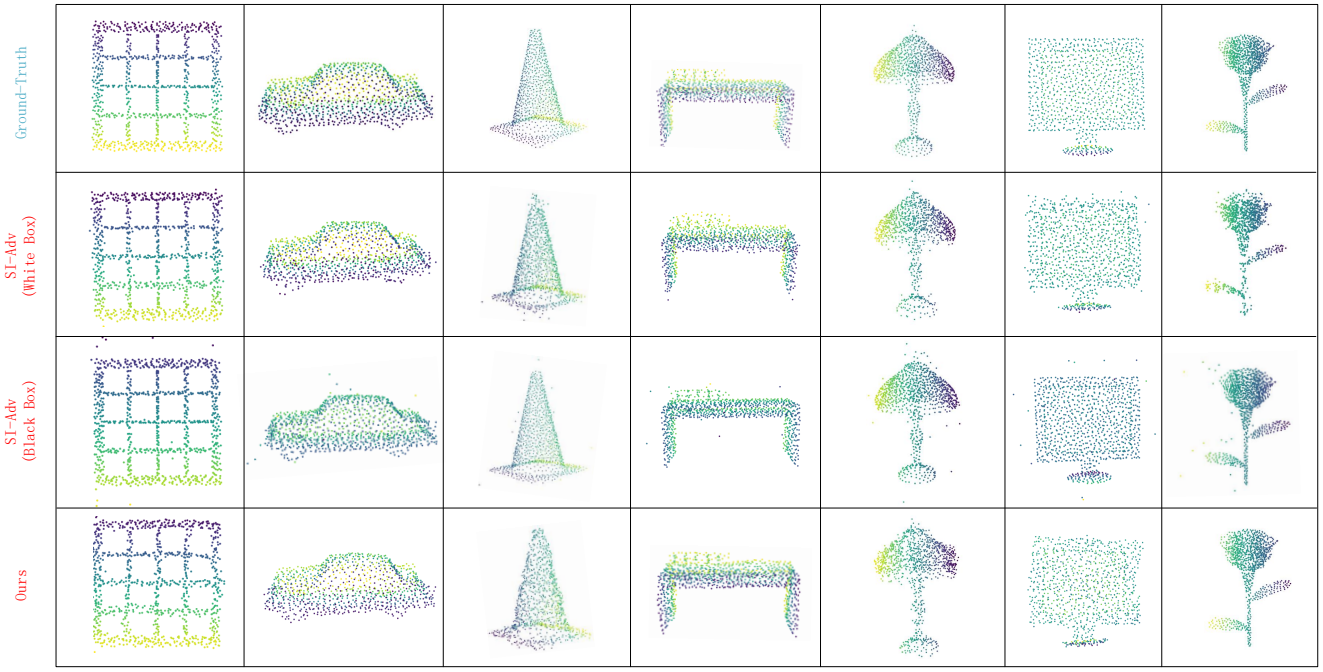

Figure 1. Visualization results of adversarial samples generated by different attack methods on PAConv model.

provide more details of this local optimum problem, and then explain why our spectrum walking strategy work.

Details of local optimum. By only utilizing the coordinate walking to move the boundary cloud along the decision boundary, the optimization process may stop earlier and stuck into a local concave of the decision boundary. For example, if we design an optimization process with 200 iterations for each boundary cloud, some optimized point clouds may keep constant at the beginning. This is because the adversarial point cloud has a chance to fall into a 'trap' due to the concave-convex of the decision boundary, where a further small walking step in arbitrary direction is likely to change the classification result of the victim model to the ground-truth label of benign cloud, thus it is hard to estimate a gradient direction of coordinate walking while keeping adversarial in the next iteration. We call this phenomenon as the local optimum problem, and the boundary cloud falling into the 'trap' may possess low quality. To this end, in addition to the data domain, we need to explore additional knowledge in other latent spaces to adjust the point cloud geometry without losing its antagonism.

Why spectrum walking work? To overcome such local

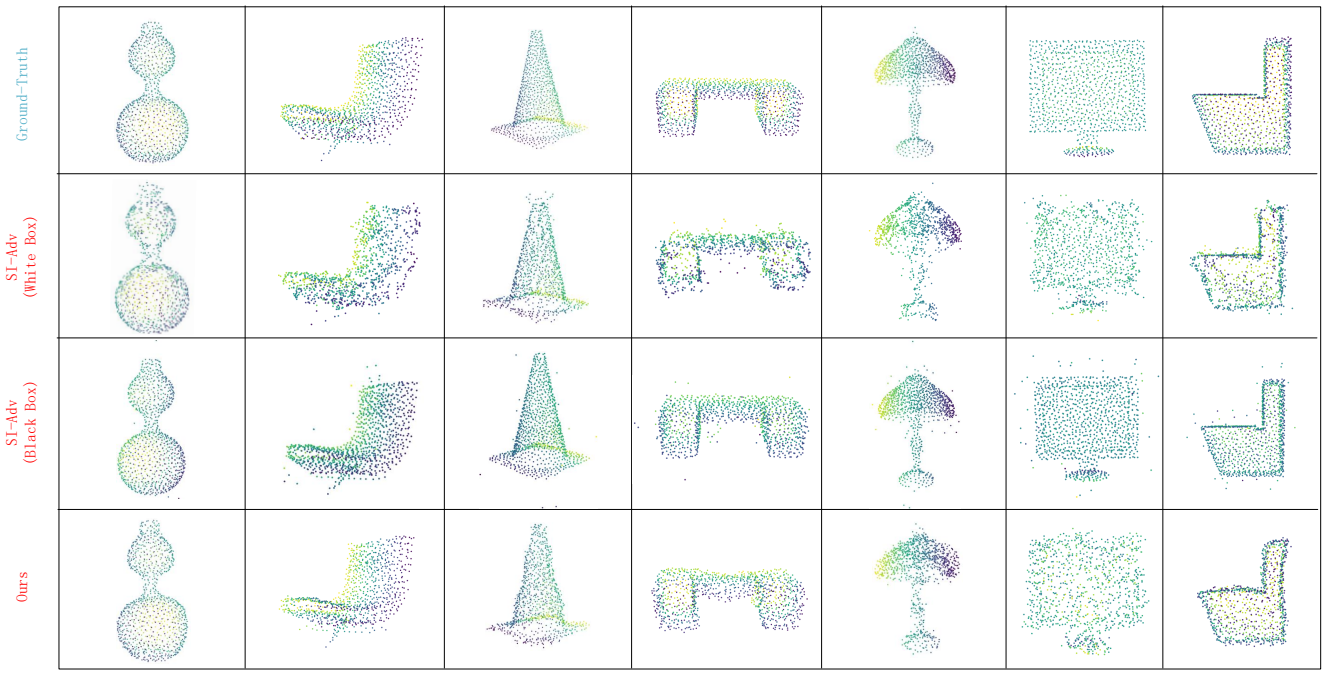

Figure 2. Visualization results of adversarial samples generated by different attack methods on SimpleView model.

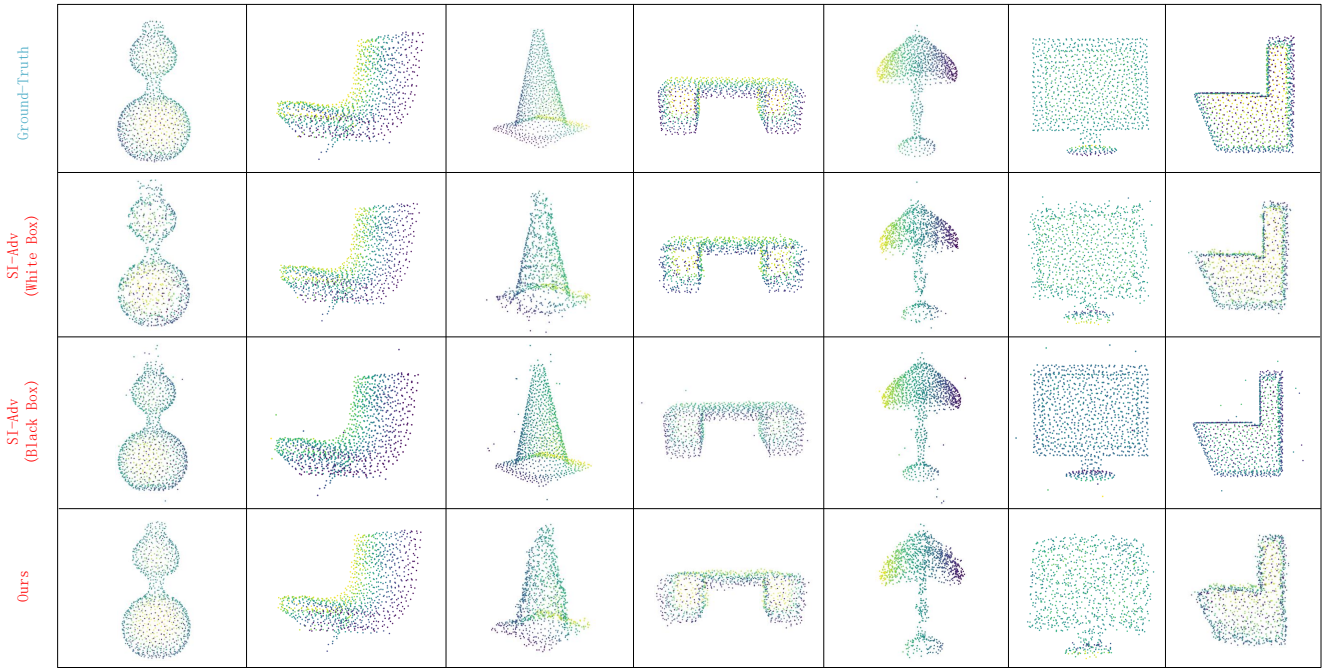

Figure 3. Visualization results of adversarial samples generated by different attack methods on DGCNN model.

optimum, the adversarial point cloud needs to walk a long step when falling into a 'trap'. A general intuition is to increase the coordinate walking size, however, directly utilizing a large coordinate walking step will produce outliers that are hard to be eliminated in the following iterations, since the outliers contribute more to the adversarial performance than ordinary points. Therefore, we design a spectrum walking strategy in the spectral domain instead of the simple data domain, which not only can preserve high-quality geometric shape of the point cloud, but also has the potential to keep its latent adversarial characteristics during the spectrum walking optimization. Moreover, unlike

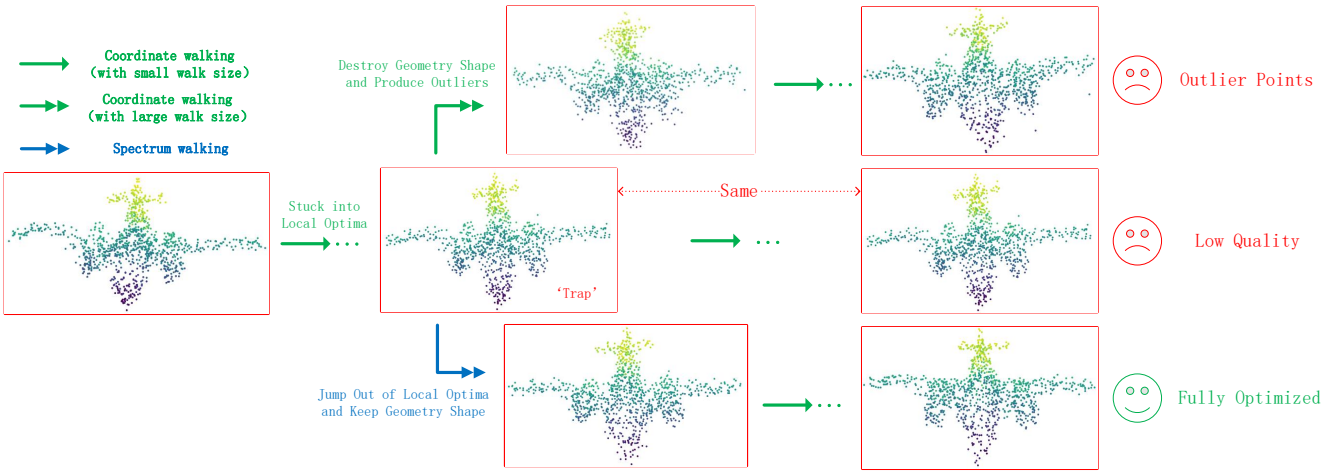

Figure 4. Visualization on the different combinations of coordinate and spectrum walking strategies for optimizing adversarial point cloud.

the coordinate-wise strategy that adds point-wise offsets for walking, walking in the spectral domain is to search trivial offsets of the spectrum frequency and will not lead to the data-domain problems of changing classification results and destroying the shape. Therefore, spectrum walking is effective enough to help to jump out of the local optimum and avoid the outlier problems. However, only utilizing the spectrum walking is not decision-boundary awareness, validated in Table 4 of the main paper. Overall, by jointly utilizing coordinate and spectrum walking strategies, we can take advantage of both of them, and optimize the best adversarial point cloud along the decision boundary. Figure 4 also illustrates the effectiveness of the joint coordinate-spectrum walking strategy.

5. Other experimental results

Running time. We conduct running time experiments to evaluate the attack efficiency of our 3DHacker. As shown in Table 3, our running time is competitive to the black-box model since our optimization steps can be efficiently achieved. The white-box model is most time-consuming since it needs complicated backpropagation through the victim model.

Method	PointNet	DGCNN	CurveNet	PAConv
SI-ADV^w	1.32s	3.87s	21.53s	2.18s
SI-ADV^b	0.58s	1.25s	8.77s	0.31s
Ours	1.16s	2.18s	10.60s	1.09s

Table 3. **Average time** for each adversarial point cloud generation.

Comparison on hard-label settings. Since existing 3D attacks rely on either model parameters or output logits, they can not be adapted to hard-label setting. Therefore, we re-implement two 2D hard-label settings into 3D domain for comparison. In Table 4, our method performs much better.

Experiments on ShapeNetPart dataset for other victim models. We conduct additional experiments on novel victim point cloud classification models [9, 8] and achieve remarkable performance similar to the results performed in main body. Our 3DHacker produces a higher Chamfer distance D_c because we modify all the points leading to a large sum of displacements. However, it performs better in D_h since we conduct relatively average perturbations to point cloud which does not count on a few outliers to confuse the victim models, leading to imperceptible and having the potential to bypass the outlier detection defense.

Method	PointNet			DGCNN		
	D_h	D_c	D_{norm}	D_h	D_c	D_{norm}
Chen *et al.* 2020	0.1284	0.0695	1.1784	0.1291	0.0493	0.9827
Li *et al.* 2021	0.0814	0.0445	1.0863	0.0892	0.0505	1.1338
Ours	**0.0136**	**0.0017**	**0.8561**	**0.0129**	**0.0026**	**0.9030**

Table 4. Comparison on the same **hard-label setting**.

Method	PointTransformer [B]			Point-BERT [C]		
	D_h	D_c	D_{norm}	D_h	D_c	D_{norm}
SI-ADV^w	0.0325	**0.0021**	1.2536	0.0161	**0.0012**	1.5381
SI-ADV^b	0.0453	0.0038	1.5702	0.0511	0.0015	1.9875
Ours	**0.0273**	0.0028	**1.0126**	**0.0157**	0.0031	**1.2848**

Table 5. Comparison on the **ShapeNetPart dataset**.

Method	PointTransformer [B]			Point-BERT [C]		
	D_h	D_c	D_{norm}	D_h	D_c	D_{norm}
SI-ADV^w	0.0491	0.0052	1.0151	0.0385	0.0028	**1.2403**
SI-ADV^b	0.0543	0.0039	**0.8312**	0.0672	**0.0027**	1.4317
Ours	**0.0243**	**0.0035**	0.8635	**0.0294**	0.0047	1.2618

Table 6. Comparison on the **ScanObjectNN dataset**.

Model	Method	ASR (%)	D_h	D_c	D_{norm}
Pointnet	SI-ADV^b	82.1	0.0458	**0.0012**	2.7804
	ours	**84.5**	**0.0146**	0.0018	**1.3519**
DGCNN	SI-ADV^b	65.3	0.0421	**0.0016**	1.5804
	ours	**71.8**	**0.0213**	0.0031	**1.4652**

Table 7. Experiment of **more defense** on Modelnet40.

Experiment of defense method. We conduct an experiment on a novel defense method: Lattice Point Classifier (LPC) [3]. Our 3DHacker achieves a better attack than SI-Adv^b [2] as we generate the adversarial sample with high similarity to the original one in both geometric topology and local point distributions.

References

[1] Ankit Goyal, Hei Law, Bowei Liu, Alejandro Newell, and Jia Deng. Revisiting point cloud shape classification with a simple and effective baseline. In *International Conference on Machine Learning*, pages 3809–3820. PMLR, 2021.

[2] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Hang Zhou, Weiming Zhang, and Nenghai Yu. Shape-invariant 3d adversarial point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15335–15344, 2022.

[3] Kaidong Li, Ziming Zhang, Cuncong Zhong, and Guanghui Wang. Robust structured declarative classifiers for 3d point clouds: Defending adversarial attacks with implicit gradients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15294–15304, 2022.

[4] Yuxin Wen, Jiehong Lin, Ke Chen, CL Philip Chen, and Kui Jia. Geometry-aware generation of adversarial point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[5] Tiange Xiang, Chaoyi Zhang, Yang Song, Jianhui Yu, and Weidong Cai. Walk in the cloud: Learning curves for point clouds shape analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 915–924, 2021.

[6] Mutian Xu, Runyu Ding, Hengshuang Zhao, and Xiaojuan Qi. Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3173–3182, 2021.

[7] 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.

[8] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19313–19322, 2022.

[9] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021.

[10] Hang Zhou, Kejiang Chen, Weiming Zhang, Han Fang, Wenbo Zhou, and Nenghai Yu. Dup-net: Denoiser and upsampler network for 3d adversarial point clouds defense. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1961–1970, 2019.