Less Is More: Sparse and Cooperative Perturbation for Point Cloud Attacks

Keke Tang[1*], Tianyu Hao[1*], Xiaofei Wang[2*], Weilong Peng[1†], Denghui Zhang[1], Peican Zhu[3†], Zhihong Tian[1,4]

[1]Guangzhou University
[2]University of Science and Technology of China
[3]Northwestern Polytechnical University
[4]Guangdong Key Laboratory of Industrial Control System Security
tangbohutbh@gmail.com, howty666@gmail.com, wxf9545@mail.ustc.edu.cn
wlpeng@tju.edu.cn, denghui.zhang@gzhu.edu.cn, ericcan@nwpu.edu.cn, tianzhihong@gzhu.edu.cn

Abstract

Most adversarial attacks on point clouds perturb a large number of points, causing widespread geometric changes and limiting applicability in real-world scenarios. While recent works explore sparse attacks by modifying only a few points, such approaches often struggle to maintain effectiveness due to the limited influence of individual perturbations. In this paper, we propose SCP, a sparse and cooperative perturbation framework that selects and leverages a compact subset of points whose joint perturbations produce amplified adversarial effects. Specifically, SCP identifies the subset where the misclassification loss is locally convex with respect to their joint perturbations, determined by checking the positive-definiteness of the corresponding Hessian block. The selected subset is then optimized to generate high-impact adversarial examples with minimal modifications. Extensive experiments show that SCP achieves 100% attack success rates, surpassing state-of-the-art sparse attacks, and delivers superior imperceptibility to dense attacks with far fewer modifications.

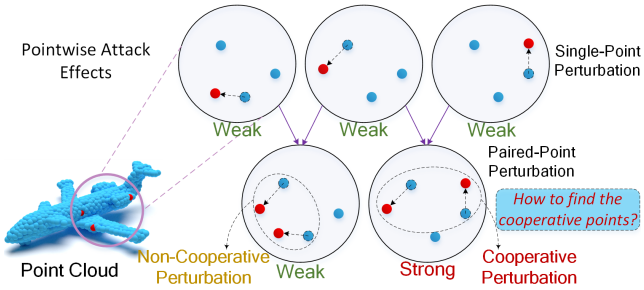

Figure 1: Illustration of point-wise perturbation effects in point cloud attacks. Single-point perturbations (top row) and non-cooperative combinations (bottom-left) often yield limited impact. In contrast, perturbing a cooperative subset (bottom-right) produces significantly stronger effects, motivating the need to identify such point groups.

Introduction

With the growing adoption of 3D sensors and the advancement of deep learning techniques, deep neural networks (DNNs) (LeCun, Bengio, and Hinton 2015) have become the dominant paradigm for point cloud perception (Guo et al. 2020; Ioannidou et al. 2017; Tang et al. 2025e), supporting applications such as autonomous driving and robotic manipulation. However, recent studies have shown that these models are highly vulnerable to adversarial attacks, where subtle perturbations to input point clouds can cause severe misclassifications (Xiang, Qi, and Li 2019). This fragility has raised growing concerns and drawn increasing attention to adversarial attacks, which have become a valuable tool for exposing model weaknesses and promoting the development of more robust 3D perception systems.

A large body of work has explored adversarial attacks on point clouds by applying point-wise perturbations across the entire input (Xiang, Qi, and Li 2019; Liu and Hu 2023; Huang et al. 2022; Tang et al. 2024a). While effective, such dense perturbations can significantly distort the object's overall geometry, limiting their practicality in real-world applications. To address this, recent methods have proposed sparse attacks that modify only a small subset of points using sparsity-constrained optimization (Kim et al. 2021) or semantic and geometric priors (Shi et al. 2022). Although these methods reduce the number of modified points, they often require stronger per-point changes and suffer from degraded attack performance under distortion constraints.

A common assumption underlying these sparse methods is that point-wise contributions to the attack are independent. However, point cloud networks involve local aggregation and hierarchical encoding (Qi et al. 2017a,b), which induce strong nonlinear interactions among points. As a result, perturbing certain points jointly can produce effects that are not captured when perturbing them individually. For example, two points within a semantically critical region may cooperatively activate or suppress key features, amplifying the adversarial effect, as illustrated in Fig. 1. We hypothesize that such cooperative subsets exist and that exploiting their joint

*These authors contributed equally.
†W. Peng and P. Zhu are joint corresponding authors.
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

influence enables stronger sparse attacks. This motivates us to move beyond isolated-point selection and instead identify compact subsets whose combined perturbation yields disproportionately strong adversarial impact.

In this paper, we propose SCP, a sparse and cooperative perturbation framework for point cloud attacks that perturbs a compact subset of points whose joint effect leads to strong adversarial behavior. Specifically, SCP first identifies a candidate set of influential points based on gradient information and then employs a greedy expansion strategy using Schur complement conditions to ensure the corresponding Hessian block is positive-definite, so that the constructed subset lies in a locally convex region of the loss landscape and thus exhibits cooperative influence that strengthens the adversarial effect. The resulting subset is then jointly optimized to induce misclassification with minimal distortion. Extensive experiments show that SCP achieves 100% attack success rates while modifying only a few points, significantly outperforming prior sparse attacks, and attains imperceptibility that is in most cases superior to dense attack methods.

Overall, our contribution is summarized as follows:

- We identify the overlooked role of cooperative interactions in sparse point cloud attacks and establish a Hessian-based criterion to characterize such synergy.
- We develop SCP, a framework that selects compact cooperative subsets via gradient screening and Schur complement-guided expansion for joint perturbation.
- We show by experiments that SCP achieves 100% attack success with minimal modifications, outperforming state-of-the-art sparse and dense attacks.

Related Work

Adversarial Attacks on 3D Point Clouds. Adversarial attacks on point clouds can be broadly categorized into addition-based (Xiang, Qi, and Li 2019), deletion-based (Zheng et al. 2019; Yang et al. 2019; Zhang et al. 2021), and perturbation-based methods (Liu and Hu 2023). Among these, perturbation-based attacks (Tang et al. 2022, 2023, 2024b,c, 2025a,b,c,d; Wang et al. 2025) that modify the coordinates of existing points are the focus of this work.

Early efforts (Xiang, Qi, and Li 2019; Liu, Yu, and Su 2019) extended classic 2D attacks like FGSM (Goodfellow, Shlens, and Szegedy 2015) and C&W (Carlini and Wagner 2017) to the 3D domain. Subsequent work focused on improving stealthiness by preserving geometric structure, e.g., via curvature (Wen et al. 2022), normal or tangent directions (Liu and Hu 2023; Huang et al. 2022), or manifold priors (Tang et al. 2024a). Generative approaches have also been explored through latent perturbation (Lee et al. 2020) and adversarial synthesis (Zhou et al. 2020). Despite their success, these methods typically perturb most or all points, causing global geometric changes that limit real-world deployability and motivate the need for sparser alternatives.

Sparse Adversarial Attacks on 3D Point Clouds. To improve stealthiness and enhance real-world applicability, sparse adversarial attacks restrict modifications to a limited subset of points. Existing methods can be broadly categorized into point removal/occlusion and point perturbation strategies. Point removal or occlusion techniques achieve adversarial effects by deleting a small number of points. Wicker and Kwiatkowska (2019) removed either randomly selected points or those contributing most to max-pool features, while Zheng et al. (2019) eliminated points with the highest or lowest saliency scores to compromise shape integrity. In contrast, perturbation-based methods manipulate a carefully selected set of points instead of removing them. Kim et al. (2021) jointly optimized point selection and displacement under l_0 constraints to minimize distortion, and Shi et al. (2022) incorporated shape priors to preserve semantic consistency during selective perturbation. Despite their effectiveness, these methods typically assume independent point-wise influence and do not account for cooperative interactions. This often leads to higher per-point distortion or degraded performance under strict sparsity constraints. In contrast, our approach explicitly models second-order interactions to identify compact, mutually reinforcing subsets, enabling more effective and efficient sparse attacks.

Deep 3D Point Cloud Classification. Modern 3D point cloud classifiers typically operate directly on point sets, beginning with PointNet (Qi et al. 2017a) and its hierarchical variant PointNet++ (Qi et al. 2017b). Later advancements include point-based convolutional networks (Wu, Qi, and Fuxin 2019), graph-based architectures (Wang et al. 2019), and transformer-based models (Zhao et al. 2021; Wu et al. 2022), which better capture local geometry and global context. We aim to attack these classifiers in a sparse manner.

Problem Formulation

Problem Statement of Sparse Adversarial Attacks

Preliminary on Adversarial Attacks. Given a point cloud $\mathcal{P} \in \mathbb{R}^{n\times 3}$ with its corresponding label $y \in \{1, \ldots, C\}$, where C is the number of semantic categories, the goal of perturbation-based adversarial attacks is to mislead a 3D deep classification model $\mathcal{F}$ into making incorrect predictions. This is achieved by crafting an adversarial point cloud $\mathcal{P}^{adv}$ through the application of imperceptible perturbations to the original points.

Formally, the adversarial point cloud is defined as:

$$\mathcal{P}^{adv} = \mathcal{P} + \mathbf{\Delta} = \mathcal{P} + \begin{bmatrix} \Delta P_1^\top & \cdots & \Delta P_n^\top \end{bmatrix}^\top, \quad (1)$$

where $\mathbf{\Delta} \in \mathbb{R}^{n\times 3}$ is the perturbation matrix, and each $\Delta P_i \in \mathbb{R}^{1\times 3}$ is a row vector perturbing point P_i.

The perturbation is typically obtained by solving the following optimization problem:

$$\min_{\mathbf{\Delta}} \; L_{mis}(\mathcal{F}, \mathcal{P} + \mathbf{\Delta}, y) + \beta_1 D(\mathcal{P}, \mathcal{P} + \mathbf{\Delta}), \quad (2)$$

where $L_{mis}(\cdot,\cdot,\cdot)$ is a misclassification loss (e.g., the negative cross-entropy), $D(\cdot,\cdot)$ measures the distortion to enforce imperceptibility, and β_1 balances the two objectives. While this formulation is typically used for untargeted attacks, it can also be readily adapted for targeted scenarios.

Sparse Adversarial Attacks. Sparse adversarial attacks aim to perturb only a small subset of points in the input point cloud, rather than modifying all points. Such selective perturbations are considered more stealthy and physically realizable, especially when large-scale modifications are impractical or easily detectable.

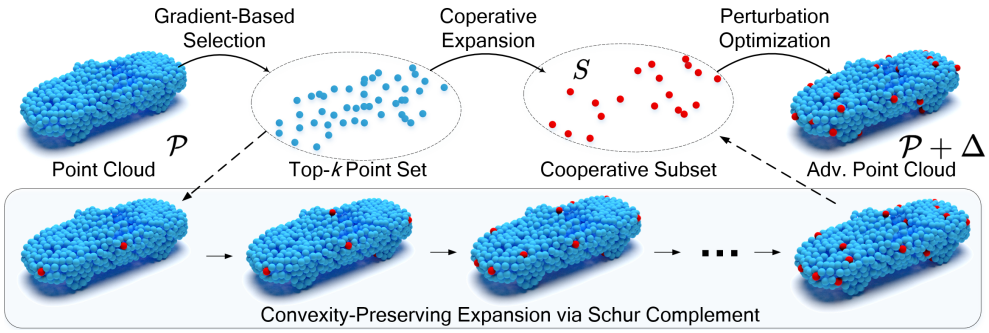

Figure 2: Overview of the SCP framework. Starting from a point cloud $\mathcal{P}$, we first apply gradient-based selection to extract a top-k point set. Then, convexity-preserving expansion via Schur complement yields a cooperative subset. Finally, perturbation optimization is performed on this subset to obtain the perturbation Δ.

Let $S = \{s_1, s_2, \ldots, s_t\}$ denote the indices of perturbed points, and let $\{\Delta P_{s_i}\}_{i=1}^{t}$ be their corresponding perturbations. The adversarial point cloud is then given by:

$$\mathcal{P}^{adv} = \mathcal{P} + \sum_{i=1}^{t} \left(\mathbf{e}_{s_i} \otimes \Delta P_{s_i}\right), \tag{3}$$

where $\mathbf{e}_{s_i} \in \mathbb{R}^n$ is a standard basis vector with a 1 in the s_i-th position and zeros elsewhere, and $\otimes$ denotes the Kronecker product. Each term yields a matrix in $\mathbb{R}^{n\times 3}$ whose s_i-th row equals ΔP_{s_i} and all other rows are zero.

Discussion. Sparse adversarial attacks typically modify only a small subset of points using sparsity-constrained optimization (Kim et al. 2021) or by incorporating semantic and geometric priors (Shi et al. 2022). While these approaches reduce the number of modified points, they generally assume independent perturbation effects and overlook interactions among the selected points. As a result, sparse attacks commonly require stronger per-point distortions to remain effective, which may degrade their imperceptibility or lead to reduced attack success under strict distortion budgets.

Cooperative Behavior in Sparse Perturbations

While sparse adversarial attacks aim to perturb only a limited number of points, the overall adversarial effectiveness depends not only on the strength of individual perturbations but also on how these perturbations interact. In particular, a small set of perturbations can produce a disproportionately large impact if they exhibit *cooperative behavior*, where their joint effect exceeds the sum of their individual effects. We formalize this notion below.

Definition of Cooperation via Jensen's Inequality. Let $\{\Delta P_{s_1}, \ldots, \Delta P_{s_t}\}$ be the perturbations applied to selected points $\{s_1, \ldots, s_t\}$. We say these perturbations are *cooperative* if the following strict Jensen-type inequality holds:

$$L_{mis}\left(\mathcal{F}, \mathcal{P} + \sum_{i=1}^{t} \alpha_i (\mathbf{e}_{s_i} \otimes \Delta P_{s_i}), y\right) > \sum_{i=1}^{t} \alpha_i \cdot L_{mis}\left(\mathcal{F}, \mathcal{P} + \mathbf{e}_{s_i} \otimes \Delta P_{s_i}, y\right), \tag{4}$$

where $\alpha_i > 0$ and $\sum_{i=1}^{t} \alpha_i = 1$. This inequality implies that the joint loss induced by the weighted combination of perturbations exceeds the weighted average of individual losses, indicating synergy among the selected perturbations.

A Sufficient Condition for Identifying Cooperation. To characterize when cooperative effects arise, we consider the local curvature of the misclassification loss. Define the concatenated perturbation vector as

$$\boldsymbol{\delta}_S = [\Delta P_{s_1} \quad \Delta P_{s_2} \quad \cdots \quad \Delta P_{s_t}]^\top \in \mathbb{R}^{3t\times 1}. \tag{5}$$

A sufficient condition for the strict inequality in Eq. (4) to hold is that the loss function L_{mis} is locally strictly convex with respect to $\boldsymbol{\delta}_S$ (Boyd and Vandenberghe 2004), which requires that the Hessian matrix

$$\mathbf{H}(\boldsymbol{\delta}_S) = \nabla^2_{\boldsymbol{\delta}_S} L_{mis}\left(\mathcal{F}, \mathcal{P} + \sum_{i=1}^{t} \alpha_i (\mathbf{e}_{s_i} \otimes \Delta P_{s_i}), y\right) \tag{6}$$

is positive definite, i.e.,

$$\mathbf{H}(\boldsymbol{\delta}_S) \succ 0. \tag{7}$$

This ensures that the perturbation combination lies in a region of positive curvature, resulting in a joint adversarial effect greater than the sum of individual effects.

This condition underpins our approach for identifying cooperative perturbation subsets and guides the selection of sparse yet effective adversarial points.

Method

In this section, we present SCP, a sparse and cooperative perturbation framework for point cloud attacks. The framework consists of two stages: we first identify a subset of points whose perturbations exhibit cooperative behavior, and then optimize perturbations over this subset to generate effective adversarial point clouds. An overview of the pipeline is illustrated in Fig. 2.

Selection of Cooperative Subset

Directly computing the full Hessian matrix of the misclassification loss with respect to all point-wise perturbations,

and subsequently identifying its largest positive definite submatrix, is computationally infeasible. To address this, SCP adopts a three-stage greedy selection strategy: (i) identifying influential points via gradient analysis, (ii) constructing a convex cooperative subset via Schur complement checks, and (iii) refining the subset with relaxed inclusion criteria.

Gradient-Based Selection of Influential Points. We first compute the gradient of the misclassification loss with respect to each point in the input point cloud:

$$g_i = \|\nabla_{P_i} L_{mis}(\mathcal{F}, \mathcal{P}, y)\|_2 . \tag{8}$$

Points with larger gradient magnitudes indicate greater local influence on the loss. We thus select the top-k points ranked by g_i as the initial candidate set for cooperative expansion.

Convexity-Preserving Expansion via Schur Complement. Given an initial cooperative set, we iteratively evaluate candidate points to determine whether their inclusion preserves the local positive definiteness of the associated Hessian block.

Let the current cooperative set correspond to a Hessian submatrix $\mathbf{A}$, a candidate point contribute a block $\mathbf{C}$, and the interaction terms be represented by a coupling block $\mathbf{B}$. The augmented Hessian has the structure:

$$\mathbf{H}^{'} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^{\top} & \mathbf{C} \end{bmatrix}. \tag{9}$$

By the Schur complement condition, the augmented matrix $\mathbf{H}^{'}$ is positive definite if and only if $\mathbf{A} \succ 0$ and

$$\mathbf{C} - \mathbf{B}^{\top}\mathbf{A}^{-1}\mathbf{B} \succ 0. \tag{10}$$

We define the *Schur Surplus* as the minimum eigenvalue of the residual block:

$$S_s = \lambda_{\min}\left(\mathbf{C} - \mathbf{B}^{\top}\mathbf{A}^{-1}\mathbf{B}\right). \tag{11}$$

If $S_s > 0$, the candidate point is deemed cooperative and added to the current subset.

Tolerance-Based Inclusion of Marginal Points. To improve flexibility and account for numerical imprecision, we relax the strict Schur condition by introducing a tolerance threshold $\epsilon > 0$. A candidate point is accepted if the corresponding Schur Surplus satisfies

$$S_s > -\epsilon. \tag{12}$$

This relaxation permits inclusion of marginal candidates that may still yield cooperative effects, even if the strict positive definiteness condition is not fully met.

Perturbation Optimization on Cooperative Subset

Given the selected cooperative subset $S = \{s_1, \ldots, s_t\}$, we optimize the associated perturbations $\{\Delta P_{s_i}\}$ by specializing the general objective in Eqn. 2 to this subset, which leads to the following formulation:

$$\min_{\{\Delta P_{s_i}\}} \; L_{mis}\left(\mathcal{F}, \mathcal{P} + \sum_{i=1}^{t} \left(\mathbf{e}_{s_i} \otimes \Delta P_{s_i}\right), y\right) + \beta_1 D, \tag{13}$$

where $\beta_1 D$ is the distortion penalty term defined in Eqn. 2.

The optimization is conducted following (Xiang, Qi, and Li 2019).

Experiments

Experimental Setup

Implementation. We implement SCP in PyTorch (Paszke et al. 2019). The cooperative subset selection starts from $k = 256$ candidate points, and the tolerance parameter is set to $\epsilon = 10^{-6}$. Perturbation optimization is performed using the Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$) with a learning rate of 0.01, running 10 binary search rounds and 500 gradient steps. All experiments are conducted on a workstation with dual 2.40 GHz CPUs, and eight NVIDIA RTX 3090 GPUs.

Datasets. We evaluate our attack on ModelNet40 (Wu et al. 2015) and ScanObjectNN (Uy et al. 2019). All point clouds are uniformly resampled to 1,024 points.

Victim Models. We evaluate our attack on four DNN classifiers: PointNet (Qi et al. 2017a), DGCNN (Wang et al. 2019), PTv1 (Zhao et al. 2021), and Mamba3D (Liang et al. 2024), using their original training settings.

Baseline Methods. For sparse attacks, we include *Adversarial Stick (AdvStick)* (Liu, Yu, and Su 2020), which inserts stick-like structures; *Random Occlusion (RandOcc)* and *Critical Occlusion (CritOcc)* (Wicker and Kwiatkowska 2019), which remove random points or those contributing most to max-pool features; *High Saliency (HighSal)* and *Low Saliency (LowSal)* (Zheng et al. 2019), which drop points with the highest or lowest saliency scores; and *Minimal Perturbation (MiniPert)* (Kim et al. 2021), which jointly optimizes point selection and displacement under l_0 regularization. For dense attacks, we select the gradient-based *PGD* and *IFGM* (Dong et al. 2020), the geometry-aware optimization method *GeoA*^{3} (Wen et al. 2022), the frequency-domain approach *AOF* (Liu, Zhang, and Zhu 2022), and the direction-guided *SI-Adv* (Huang et al. 2022).

Evaluation Setting and Metrics. For sparse attack methods, we follow the configurations reported in their original papers and evaluate performance using attack success rate (ASR), Chamfer distance (CD) (Fan, Su, and Guibas 2017), Hausdorff distance (HD) (Taha and Hanbury 2015), and the number of modified points (# Points) as an indicator of sparsity. For dense attacks and our SCP framework, we configure each approach to achieve its maximum ASR. Under this maximal adversarialness condition (Tang et al. 2024a), we further evaluate imperceptibility using additional metrics: l_2-norm (l_2), Curvature (Curv), Geometric Regularity (GR) (Wen et al. 2022), and Earth Mover's Distance (EMD) (Rubner, Tomasi, and Guibas 2000).

Comparison and Performance Analysis

Comparison with Sparse Attacks. We first compare SCP with representative sparse attack methods. As shown in Tab. 1, SCP consistently achieves the highest attack success rates across all datasets and classifiers, reaching 100% in all cases. In contrast, the success rates of other methods remain far below this level, often failing to surpass 80%. Furthermore, these baselines produce significantly larger CD and HD values, indicating lower imperceptibility. They also tend to modify more points, whereas SCP attains stronger attacks with sparser changes to the input point clouds.

Model	Metric	ModelNet40							ScanObjectNN						
		AdvStick	RandOcc	CritOcc	HighSal	LowSal	MiniPert	Ours	AdvStick	RandOcc	CritOcc	HighSal	LowSal	MiniPert	Ours
PointNet	ASR	83.70	53.23	21.74	60.47	57.80	89.38	**100**	87.50	60.19	43.83	67.12	64.99	91.72	**100**
	CD (10^{-4})	49.30	6.94	1.27	8.41	6.82	1.55	**0.74**	51.80	5.71	1.80	6.73	5.87	1.12	**0.93**
	HD (10^{-2})	14.90	**0.25**	1.03	**0.25**	**0.25**	1.88	0.32	16.7	0.25	2.57	**0.24**	0.25	1.15	0.42
	# Points	210	413	50	200	200	36	**34**	210	397	70	200	200	34	**32**
DGCNN	ASR	73.70	40.20	8.87	46.22	43.08	73.64	**100**	85.90	44.90	28.64	38.02	33.92	78.29	**100**
	CD (10^{-4})	17.54	7.68	1.31	10.47	11.49	5.10	**1.06**	10.45	7.03	2.38	7.83	6.33	2.67	**1.68**
	HD (10^{-2})	4.68	**0.24**	1.05	0.26	0.25	1.91	0.84	4.01	**0.23**	2.34	0.25	0.28	1.15	0.46
	# Points	500	689	86	200	200	138	**62**	500	712	93	200	200	110	**60**
PTv1	ASR	61.58	42.76	12.83	37.25	32.10	54.69	**100**	74.70	61.07	25.63	31.56	27.71	49.22	**100**
	CD (10^{-4})	16.79	6.37	1.36	13.24	11.25	6.78	**1.01**	8.10	6.51	2.57	6.02	5.37	3.70	**0.42**
	HD (10^{-2})	5.30	0.26	1.16	**0.25**	**0.25**	2.50	0.88	6.13	0.25	2.21	**0.24**	0.25	0.86	0.78
	# Points	500	427	114	200	200	207	**52**	500	488	125	200	200	206	**53**
Mamba3D	ASR	74.91	50.64	7.51	51.06	48.65	73.32	**100**	88.15	65.15	22.46	53.09	49.36	72.45	**100**
	CD (10^{-4})	17.57	9.29	1.41	9.52	10.76	2.27	**0.83**	8.35	8.12	2.70	8.79	8.22	1.11	**0.68**
	HD (10^{-2})	4.33	0.26	1.15	**0.25**	0.26	0.88	0.79	3.98	**0.24**	2.70	0.25	0.25	1.09	0.90
	# Points	500	482	97	200	200	74	**49**	500	473	106	200	200	52	**51**

Table 1: Comparison on ASR, imperceptibility (CD and HD), and the number of modified points (# Points) for different sparse attacks across four DNN classifiers on ModelNet40 and ScanObjectNN.

Model	Attack	ModelNet40								ScanObjectNN							
		ASR (%)	CD (10^{-4})	HD (10^{-2})	l_2	GR	Curv (10^{-2})	EMD (10^{-2})	# Points	ASR (%)	CD (10^{-4})	HD (10^{-2})	l_2	GR	Curv (10^{-2})	EMD (10^{-2})	# Points
PointNet	PGD	100	26.582	10.777	2.804	0.406	5.681	5.705	1003	100	20.12	7.314	2.722	0.307	7.641	4.993	1019
	IFGM	100	8.982	9.859	1.195	0.368	1.209	1.537	670	100	5.915	6.297	0.993	0.258	1.830	1.463	762
	GeoA^3	100	4.869	0.524	1.423	0.215	0.348	2.415	957	100	3.425	0.655	1.355	0.128	0.574	0.493	933
	AOF	100	18.063	1.204	1.896	0.161	3.719	3.575	966	100	5.842	0.484	0.924	0.159	3.455	1.814	675
	SI-Adv	100	2.855	2.259	0.779	0.183	0.276	0.783	998	100	1.120	1.122	**0.400**	0.116	2.154	0.493	972
	Ours	100	**0.741**	**0.315**	**0.521**	**0.160**	**0.148**	**0.199**	**34**	100	**0.931**	**0.416**	0.665	**0.091**	**0.133**	**0.167**	**32**
DGCNN	PGD	100	28.517	8.772	2.853	0.347	5.915	5.927	1024	100	19.29	1.030	2.766	0.145	7.963	5.091	1024
	IFGM	100	10.701	7.148	1.622	0.363	2.849	3.777	1024	100	4.821	0.842	0.891	0.120	3.349	2.256	1024
	GeoA^3	100	13.017	2.059	1.589	0.169	1.700	2.279	1024	100	9.328	0.886	1.508	0.213	4.337	2.241	1024
	AOF	100	25.229	1.249	2.351	0.161	4.325	4.959	1024	100	7.973	**0.375**	1.084	0.107	4.262	2.513	1024
	SI-Adv	100	8.911	1.811	**1.348**	**0.136**	1.317	2.908	1024	100	3.007	0.555	0.752	**0.096**	1.276	1.526	1024
	Ours	100	**1.059**	**0.841**	1.378	0.150	**0.385**	**0.719**	**62**	100	**1.675**	0.458	**0.691**	0.163	**0.383**	**0.384**	**60**
PTv1	PGD	100	24.521	1.724	2.584	0.181	5.548	5.589	1024	100	17.79	1.484	2.636	0.136	7.483	4.899	1024
	IFGM	100	7.544	5.563	1.082	0.520	0.900	1.704	1024	100	2.778	0.464	0.819	0.107	2.278	1.592	1024
	GeoA^3	100	7.815	3.296	0.973	0.199	1.618	2.357	1024	100	6.452	0.830	1.050	0.122	4.259	2.132	1024
	AOF	100	31.840	1.164	3.065	0.194	6.943	5.581	1024	100	11.96	**0.370**	1.029	0.106	4.220	2.432	1024
	SI-Adv	100	13.456	3.044	1.806	0.186	2.334	3.385	1024	100	6.833	1.186	1.221	0.115	2.253	2.090	1024
	Ours	100	**1.010**	**0.877**	**0.795**	**0.139**	**0.168**	**0.314**	**52**	100	**0.415**	0.784	**0.366**	**0.106**	**0.250**	**0.183**	**53**
Mamba3D	PGD	100	28.407	3.170	2.763	0.229	6.084	5.872	1024	100	21.27	1.371	2.604	0.159	8.108	5.162	1024
	IFGM	100	6.075	1.652	**0.854**	0.179	1.827	1.979	1024	100	4.902	1.521	0.962	0.146	3.211	1.935	1024
	GeoA^3	100	8.833	1.772	1.019	0.255	1.959	2.043	1024	100	5.748	0.899	1.083	0.177	3.588	2.014	1024
	AOF	100	18.073	**0.711**	1.619	**0.137**	3.413	4.033	1024	100	8.238	**0.262**	0.967	**0.103**	3.983	2.505	1024
	SI-Adv	100	5.685	1.722	0.965	0.152	0.617	1.940	1024	100	2.836	0.887	**0.868**	0.127	0.833	1.256	1024
	Ours	100	**0.831**	0.794	1.010	0.158	**0.514**	**0.581**	**49**	100	**0.682**	0.904	0.879	0.138	**0.294**	**0.271**	**51**

Table 2: Comparison on the perturbation sizes required by dense attack methods and our sparse SCP to reach their highest achievable ASR, evaluated across different DNN classifiers on ModelNet40 and ScanObjectNN.

Comparison with Dense Attacks. To assess the effectiveness of using fewer but cooperative perturbations, we compare SCP with dense baselines that modify nearly all points. Tab. 2 summarizes the perturbation sizes required to achieve the highest ASR on ModelNet40 and ScanObjectNN across various classifiers. Dense attacks including PGD, IFGM, GeoA^3, AOF, and SI-Adv reach 100% ASR while introducing high distortion, as shown by large CD, HD, and other imperceptibility metrics. In contrast, SCP also achieves 100% ASR while modifying fewer than 50 points on average, which is two orders of magnitude fewer than dense methods. It also produces lower distortion on most metrics such as CD, HD, GR, Curv, and EMD. These results show that even when dense methods modify almost the entire input, our cooperative framework offers better imperceptibility with significantly fewer changes.

Visualization. Fig. 3 shows adversarial point clouds generated by various attack methods against PointNet. Most

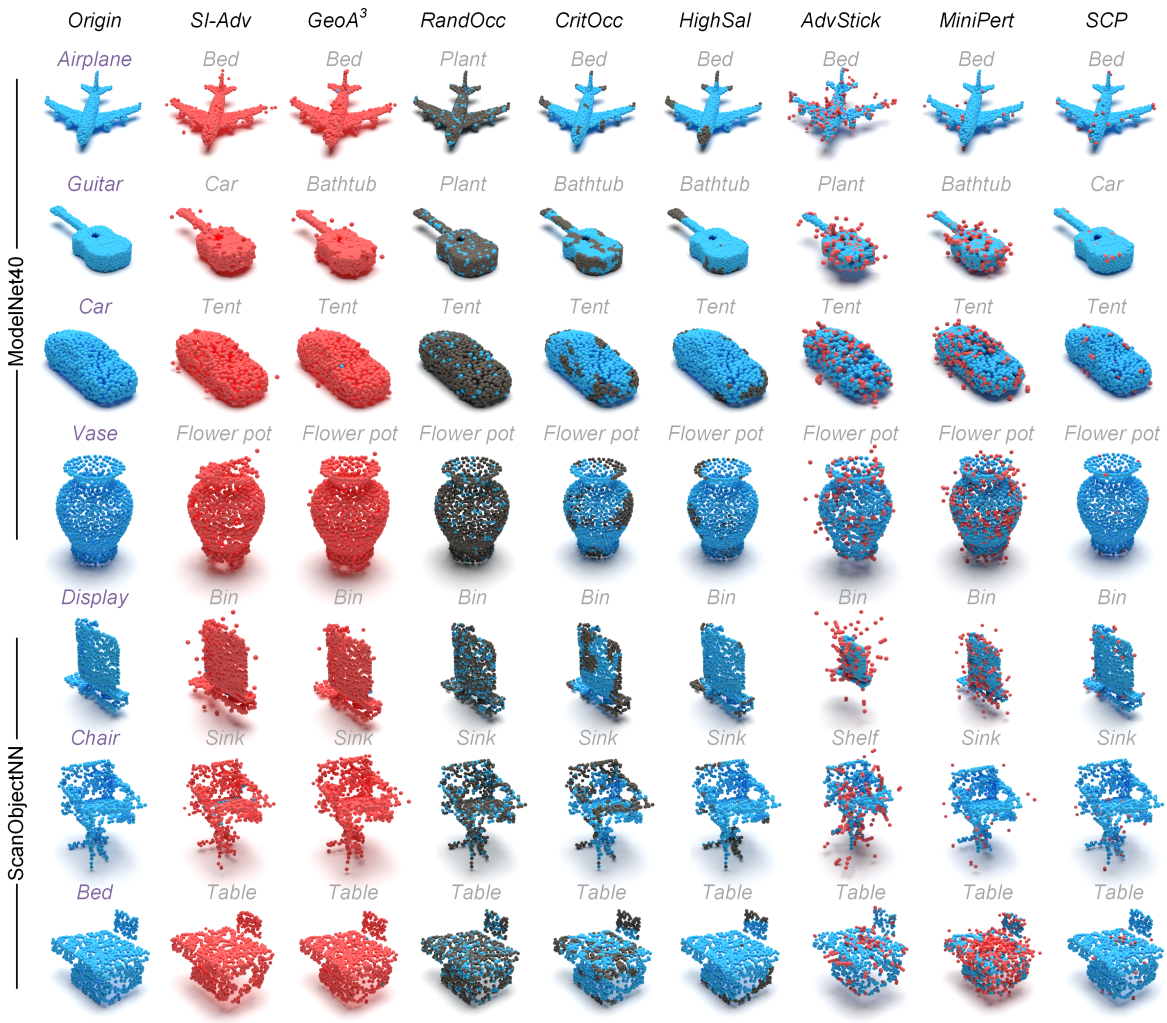

Figure 3: Visualization of original and adversarial point clouds generated by different attack methods targeting PointNet on ModelNet40 and ScanObjectNN. Blue points represent original points, red points denote perturbed points, and black points indicate removed points. Ground truth and predicted labels are shown above each point cloud in purple and gray, respectively.

sparse approaches remove points, creating visible holes, or perturb points with large displacements, both of which degrade geometric coherence. SCP, in contrast, perturbs only a small number of points with small magnitudes. Together with the quantitative results in Tab. 1, which show that SCP achieves the highest attack success rates, e.g., 100%, these observations confirm that SCP effectively balances attack strength and imperceptibility.

Ablation Studies and Other Analysis

Effect of Cooperative Subset Size. To assess the impact of cooperative subset size, we evaluate SCP under varying limits on the number of perturbed points, as shown in Tab. 3. The results reveal a clear trend: even with only two perturbed points, SCP attains ASR close to 90% on both datasets. Increasing the subset size to 5–10 points significantly boosts ASR, while distortion metrics such as CD, HD, l_2, and EMD remain low. When the subset size exceeds 20 points, SCP achieves near-perfect ASR and further reduces HD, indicating smaller maximum point displacements. At around 30 points, SCP reaches 100% ASR on both datasets with distortion still an order of magnitude lower than dense baselines. These results confirm that cooperative selection allows SCP to achieve strong attacks with very few sparse perturbations, validating the efficiency of the proposed approach.

Effect of Different Cooperative Subset Selection Strategies. To assess whether our greedy selection compromises performance, we compare SCP using full Hessian-based subset selection with SCP using the proposed strategy. The full Hessian variant identifies cooperative points by directly computing and analyzing the complete Hessian matrix. As shown in Tab. 4, the greedy approach achieves almost the same attack success, distortion metrics, and number of selected cooperative points as the full Hessian method, while reducing computation time by an order of magnitude, confirming that it preserves effectiveness at much lower cost.

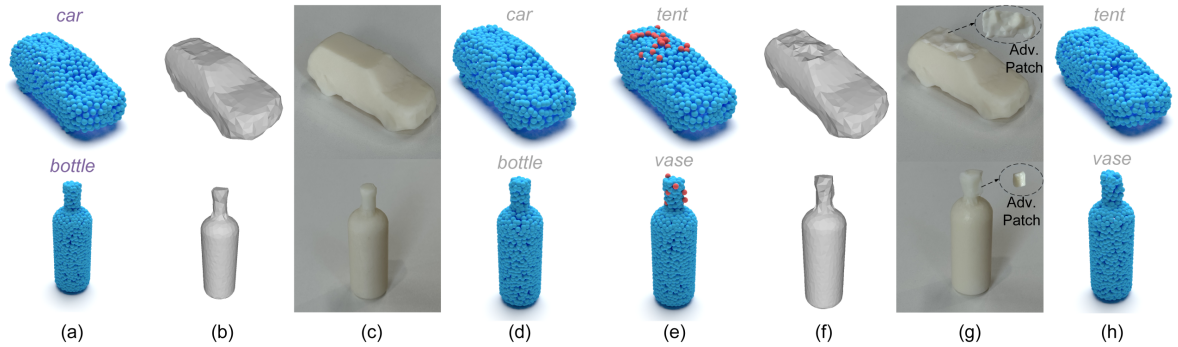

Figure 4: Physical attack results of SCP. (a) Original point clouds, (b) reconstructed meshes, (c) 3D-printed objects, (d) re-scanned point clouds of (c), (e) adversarial point clouds generated by SCP, (f) their reconstructed meshes, (g) physical adversarial objects obtained by attaching the 3D-printed patches to (c), and (h) re-scanned adversarial point clouds captured from the patched objects in (g). Purple text indicates ground-truth labels, while gray text indicates predictions by PointNet.

# Points	ModelNet40							ScanObjectNN						
	ASR (%)	CD (10^{-4})	HD (10^{-2})	l_2	GR	Curv (10^{-2})	EMD (10^{-2})	ASR (%)	CD (10^{-4})	HD (10^{-2})	l_2	GR	Curv (10^{-2})	EMD (10^{-2})
2	87.792	0.878	6.060	0.449	0.270	**0.040**	**0.062**	89.305	**0.852**	4.662	**0.211**	0.131	**0.031**	**0.031**
5	98.000	0.701	2.680	**0.440**	0.210	0.060	0.090	96.805	0.990	1.711	0.338	0.111	0.052	0.048
10	98.708	**0.607**	1.920	0.443	0.167	0.097	0.125	99.400	1.074	0.645	0.362	0.100	0.077	0.070
20	99.190	0.795	0.908	0.476	0.165	0.113	0.178	99.960	0.985	0.598	0.489	0.094	0.110	0.103
∼30	**100**	0.741	**0.315**	0.521	**0.160**	0.148	0.199	**100**	0.931	**0.416**	0.665	**0.091**	0.133	0.167

Table 3: Impact of cooperative subset size on SCP performance against PointNet on ModelNet40 and ScanObjectNN.

Model	Hessian	ASR (%)	CD (10^{-4})	HD (10^{-2})	l_2	GR	# Points	Time (s)
PointNet	Full	100	0.746	0.297	0.509	0.155	34	87.89
	Sparse	100	0.741	0.315	0.521	0.160	34	11.66
DGCNN	Full	100	1.044	0.840	1.370	1.500	63	197.96
	Sparse	100	1.059	0.841	1.378	0.150	62	24.27
PTv1	Full	100	1.012	0.881	0.796	0.138	52	349.15
	Sparse	100	1.010	0.877	0.795	0.139	52	43.47
Mamba3D	Full	100	0.824	0.788	1.005	0.157	49	285.25
	Sparse	100	0.831	0.794	1.010	0.158	49	37.09

Table 4: Comparison of SCP with full Hessian-based (Full) and greedy (Sparse) subset selection for attacking four DNN models on ModelNet40.

Physical Validation. We further evaluate SCP in a physical setting by transforming adversarial point clouds into real-world objects through 3D printing. To enable patch fabrication, the cooperative subset expansion is intentionally constrained to select spatially close points, forming a localized perturbation region. This region is 3D-printed as a detachable patch and affixed to the clean object. After re-scanning and re-sampling the patched object, the resulting point cloud is fed back into the classifier. As shown in Fig. 4, the clean samples are correctly classified, whereas the patched objects consistently cause misclassification, confirming that SCP successfully transfers to the physical domain.

Analyzing Cooperative/Counteractive Relationship. For each perturbed point, we calculate, in a pairwise manner, the number of other points satisfying Eqn. 4 with "$>$" (cooperative) or "$<$" (counteractive), evaluated on ModelNet40 using PointNet. The distributions in Fig. 5 show that cooperative counts are concentrated at small values but span a wide range, with a few points connected to dozens or even over a hundred others. In contrast, counteractive counts remain very low and drop sharply as the count increases, with only a negligible number of points showing higher values.

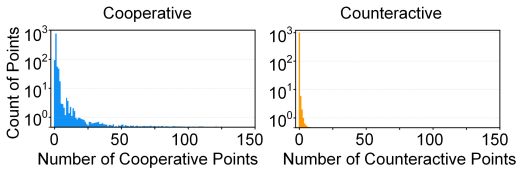

Figure 5: Frequency distributions of cooperative and counteractive relationships for PointNet on ModelNet40.

Conclusion

In this paper, we have presented SCP, a sparse and cooperative perturbation framework for adversarial attacks on 3D point clouds. SCP selects a compact yet cooperative subset of points, where restricted perturbations still amplify adversarial effects. Extensive experimental results show that SCP consistently achieves 100% attack success rates while maintaining high imperceptibility, outperforming both state-of-the-art sparse and dense methods. In future work, we plan to extend SCP to black-box and transferable attack settings.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (62472117, 62572400, U2436208, 62372129), the Guangdong Basic and Applied Basic Research Foundation (2025A1515010157, 2024A1515012064), the Science and Technology Projects in Guangzhou (2025A03J0137, 2024B0101010002), the CCF-NetEase ThunderFire Innovation Research Funding (CCF-Netease 202514), the Project of Guangdong Key Laboratory of Industrial Control System Security (2024B1212020010), the Key Laboratory Project of Computing Power Network and Information Security, Ministry of Education (2023ZD02), and the High-Quality Talent Training Program – Graduate Student Development Project of the Graduate School, Guangzhou University.

References

Boyd, S. P.; and Vandenberghe, L. 2004. *Convex optimization*. Cambridge university press.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *S&P*, 39–57.

Dong, X.; Chen, D.; Zhou, H.; Hua, G.; Zhang, W.; and Yu, N. 2020. Self-Robust 3D Point Recognition via Gather-Vector Guidance. In *CVPR*, 11513–11521.

Fan, H.; Su, H.; and Guibas, L. J. 2017. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, 605–613.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and harnessing adversarial examples. In *ICLR*.

Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; and Bennamoun, M. 2020. Deep learning for 3d point clouds: A survey. *IEEE TPAMI*, 43(12): 4338–4364.

Huang, Q.; Dong, X.; Chen, D.; Zhou, H.; Zhang, W.; and Yu, N. 2022. Shape-invariant 3D Adversarial Point Clouds. In *CVPR*, 15335–15344.

Ioannidou, A.; Chatzilari, E.; Nikolopoulos, S.; and Kompatsiaris, I. 2017. Deep learning advances in computer vision with 3d data: A survey. *ACM computing surveys (CSUR)*, 50(2): 1–38.

Kim, J.; Hua, B.-S.; Nguyen, T.; and Yeung, S.-K. 2021. Minimal adversarial examples for deep learning on 3d point clouds. In *ICCV*, 7797–7806.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *Nature*, 521(7553): 436–444.

Lee, K.; Chen, Z.; Yan, X.; Urtasun, R.; and Yumer, E. 2020. ShapeAdv: Generating Shape-Aware Adversarial 3D Point Clouds. *arXiv preprint arXiv:2005.11626*.

Liang, D.; Zhou, X.; Xu, W.; Zhu, X.; Zou, Z.; Ye, X.; Tan, X.; and Bai, X. 2024. Pointmamba: A simple state space model for point cloud analysis. In *NeurIPS*, volume 37, 32653–32677.

Liu, B.; Zhang, J.; and Zhu, J. 2022. Boosting 3D adversarial attacks with attacking on frequency. *IEEE Access*, 10: 50974–50984.

Liu, D.; and Hu, W. 2023. Imperceptible Transfer Attack and Defense on 3D Point Cloud Classification. *IEEE TPAMI*, 45(4): 4727–4746.

Liu, D.; Yu, R.; and Su, H. 2019. Extending adversarial attacks and defenses to deep 3d point cloud classifiers. In *ICIP*, 2279–2283.

Liu, D.; Yu, R.; and Su, H. 2020. Adversarial shape perturbations on 3d point clouds. In *ECCV Workshops*, 88–104.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Köpf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*, 8026–8037.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 652–660.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017b. PointNet++ deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 5105–5114.

Rubner, Y.; Tomasi, C.; and Guibas, L. J. 2000. The earth mover's distance as a metric for image retrieval. *IJCV*, 40: 99–121.

Shi, Z.; Chen, Z.; Xu, Z.; Yang, W.; Yu, Z.; and Huang, L. 2022. Shape prior guided attack: Sparser perturbations on 3d point clouds. In *AAAI*, volume 36, 8277–8285.

Taha, A. A.; and Hanbury, A. 2015. Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15: 1–28.

Tang, K.; Du, Z.; Peng, W.; Wang, X.; Liu, D.; Liu, L.; and Tian, Z. 2025a. Imperceptible 3d point cloud attacks on lattice-based barycentric coordinates. In *AAAI*, volume 39, 20814–20822.

Tang, K.; Du, Z.; Peng, W.; Wang, X.; Zhu, P.; Liu, L.; and Tian, Z. 2025b. Cage-Based Deformation for Transferable and Undefendable Point Cloud Attack. *arXiv preprint arXiv:2507.00690*.

Tang, K.; Gao, Y.; Peng, W.; Wang, X.; Fang, M.; and Zhu, P. 2025c. Transferable and Undefendable Point Cloud Attacks via Medial Axis Transform. *arXiv preprint arXiv:2507.18870*.

Tang, K.; He, X.; Peng, W.; Wu, J.; Shi, Y.; Liu, D.; Zhou, P.; Wang, W.; and Tian, Z. 2024a. Manifold Constraints for Imperceptible Adversarial Attacks on Point Clouds. In *AAAI*, volume 38, 5127–5135.

Tang, K.; Huang, L.; Peng, W.; Liu, D.; Wang, X.; Ma, Y.; Liu, L.; and Tian, Z. 2024b. FLAT: Flux-aware Imperceptible Adversarial Attacks on 3D Point Clouds. In *ECCV*, 198–215.

Tang, K.; Ke, W.; Peng, W.; Wang, X.; Du, Z.; Wu, Z.; Zhu, P.; and Tian, Z. 2025d. Imperceptible Adversarial Attacks on Point Clouds Guided by Point-to-Surface Field. In *ICASSP*, 1–5.

Tang, K.; Ma, Y.; Miao, D.; Song, P.; Gu, Z.; Tian, Z.; and Wang, W. 2025e. Decision Fusion Networks for Image Classification. *TNNLS*, 36(3): 3890–3903.

Tang, K.; Shi, Y.; Lou, T.; Peng, W.; He, X.; Zhu, P.; Gu, Z.; and Tian, Z. 2022. Rethinking perturbation directions for imperceptible adversarial attacks on point clouds. *IEEE Internet of Things Journal*, 10(6): 5158–5169.

Tang, K.; Wang, Z.; Peng, W.; Huang, L.; Wang, L.; Zhu, P.; Wang, W.; and Tian, Z. 2024c. SymAttack: Symmetry-aware Imperceptible Adversarial Attacks on 3D Point Clouds. In *MM*, 3131–3140.

Tang, K.; Wu, J.; Peng, W.; Shi, Y.; Song, P.; Gu, Z.; Tian, Z.; and Wang, W. 2023. Deep manifold attack on point clouds via parameter plane stretching. In *AAAI*, volume 37, 2420–2428.

Uy, M. A.; Pham, Q.-H.; Hua, B.-S.; Nguyen, T.; and Yeung, S.-K. 2019. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *ICCV*, 1588–1597.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic graph cnn for learning on point clouds. *ACM TOG (Proc. of SIGGRAPH)*, 38(5): 1–12.

Wang, Z.; Peng, W.; Wang, L.; Wu, Z.; Zhu, P.; and Tang, K. 2025. Eia: Edge-aware imperceptible adversarial attacks on 3d point clouds. In *MMM*, 348–361.

Wen, Y.; Lin, J.; Chen, K.; Chen, C. P.; and Jia, K. 2022. Geometry-Aware Generation of Adversarial Point Clouds. *IEEE TPAMI*, 44(6): 2984–2999.

Wicker, M.; and Kwiatkowska, M. 2019. Robustness of 3d deep learning in an adversarial setting. In *CVPR*, 11767–11775.

Wu, W.; Qi, Z.; and Fuxin, L. 2019. Pointconv: Deep convolutional networks on 3d point clouds. In *CVPR*, 9621–9630.

Wu, X.; Lao, Y.; Jiang, L.; Liu, X.; and Zhao, H. 2022. Point transformer v2: Grouped vector attention and partition-based pooling. In *NeurIPS*, volume 35, 33330–33342.

Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; and Xiao, J. 2015. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, 1912–1920.

Xiang, C.; Qi, C. R.; and Li, B. 2019. Generating 3D Adversarial Point Clouds. In *CVPR*, 9136–9144.

Yang, J.; Zhang, Q.; Fang, R.; Ni, B.; Liu, J.; and Tian, Q. 2019. Adversarial attack and defense on point sets. *arXiv preprint arXiv:1902.10899*.

Zhang, J.; Jiang, C.; Wang, X.; and Cai, M. 2021. Td-Net: Topology Destruction Network For Generating Adversarial Point Cloud. In *ICIP*, 3098–3102.

Zhao, H.; Jiang, L.; Jia, J.; Torr, P. H.; and Koltun, V. 2021. Point transformer. In *ICCV*, 16259–16268.

Zheng, T.; Chen, C.; Yuan, J.; Li, B.; and Ren, K. 2019. Pointcloud saliency maps. In *ICCV*, 1598–1606.

Zhou, H.; Chen, D.; Liao, J.; Chen, K.; Dong, X.; Liu, K.; Zhang, W.; Hua, G.; and Yu, N. 2020. Lg-gan: Label guided adversarial network for flexible targeted attack of point cloud based deep networks. In *CVPR*, 10356–10365.