Imperceptible 3D Point Cloud Attacks on Lattice-based Barycentric Coordinates

Keke Tang[1*], Ziyong Du[1*], Weilong Peng[1†], Xiaofei Wang[2], Daizong Liu[3], Ligang Liu[2], Zhihong Tian[1]

[1]Guangzhou University
[2]University of Science and Technology of China
[3]Peking University
tangbohutbh@gmail.com, duxiaoshuaicst@gmail.com, wlpeng@tju.edu.cn,
wxf9545@mail.ustc.edu.cn, dzliu@hust.edu.cn, lgliu@ustc.edu.cn, tianzhihong@gzhu.edu.cn

Abstract

Imperceptible adversarial attacks on 3D point clouds rely on effective constraints. While manifold constraints have notable advantages over Euclidean ones, the global parameterization used in current methods often fails to fully preserve manifold properties. In this paper, we propose to constrain lattice-based barycentric coordinates during attacks from a local parametric perspective to ensure imperceptibility. Specifically, we utilize a permutohedral lattice to partition point clouds into multiple cells, and then extract barycentric coordinates for each point within these cells, forming a local parametric representation of the point clouds. By enforcing local parametric constraints that minimize the displacement of barycentric coordinates, we largely preserve the manifold properties, ultimately leading to improved imperceptibility. Extensive experiments validate that integrating these local parametric constraints into conventional adversarial attacks yields superior imperceptibility, outperforming state-of-the-art methods.

Introduction

With advances in deep learning (LeCun, Bengio, and Hinton 2015; Tang et al. 2022a) and depth-sensing techniques, deep neural networks (DNNs) have become the dominant method for 3D point cloud perception (Guo et al. 2020). However, DNN classifiers are vulnerable to adversarial attacks, where small perturbations to input point clouds can cause misclassifications (Xiang, Qi, and Li 2019), posing challenges for real-world applications, e.g., robotics (Jiang et al. 2021; You et al. 2017; Lin et al. 2022). Exploring such attacks is crucial for evaluating and enhancing the robustness of deep point cloud classifiers (Bai et al. 2021; Tang et al. 2024d).

As a crucial requirement for adversarial attacks, imperceptibility has been extensively studied. Early approaches employ metrics such as the l_2-norm, Chamfer distance, and Hausdorff distance to constrain perturbations on 3D point clouds. More recent research focus on reducing distortion from complex geometric perspectives, including limiting curvature changes (Wen et al. 2022) and guiding perturbation directions (Liu and Hu 2023; Huang et al. 2022). However, these methods often fail to fully capture the essence of 3D shapes as fundamentally 2-manifold surfaces (Spanier 1989), making it difficult to preserve manifold properties. To address this issue, Tang et al. (2024a) introduce bijective manifold mapping, which projects point clouds onto parameterized shapes like spheres, see Fig. 1. This method more effectively captures and constrains manifold properties, thereby significantly enhancing imperceptibility of adversarial attacks on 3D point clouds.

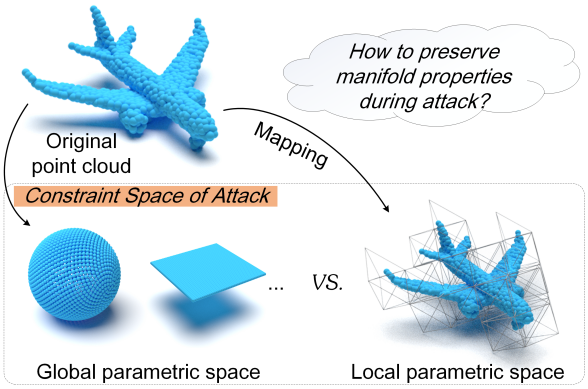

Figure 1: To preserve manifold properties during adversarial attacks, global parameterization maps the point cloud into a single parametric coordinate system, such as a sphere or plane, for applying constraints. In contrast, our lattice-based solution maps the point cloud into a local parametric space.

Essentially, projecting point clouds onto 2-manifold surfaces (Tang et al. 2024a) involves a method known as global parameterization. This approach uses a single parametric coordinate system to represent the entire object, aiming to provide a unified representation space for arbitrary shapes. However, it poses challenges, especially when dealing with complex shapes that have significant topological variations, such as differences in genus. Additionally, the necessity of fitting a global parameterization space can result in considerable geometric distortions. Consequently, the effectiveness of constraints derived from this space for managing changes in manifold properties during adversarial attacks is reduced.

To resolve the above issue, we devise a local parametric space where the manifold properties of 3D point clouds are

*These authors contributed equally.
†Corresponding author.

finely constrained during attacks. Specifically, we utilize a permutohedral lattice to partition point clouds into multiple cells, and further enhance the partition through additional subdivision and refinement, see Fig. 1. Therefore, each lattice can be viewed as a local space, and we extract barycentric coordinates within each cell to define local parametric constraints, limiting changes in these coordinates. These constraints, based on lattice-based barycentric coordinates (LBCs), can be integrated into existing adversarial attacks, e.g., C&W (Xiang, Qi, and Li 2019), to restrict changes in the manifold properties of 3D point clouds, thereby enhancing imperceptibility. We validate the effectiveness of our LBC-based local parametric constraints on various adversarial attack solutions for attacking common 3D DNN classifiers. Extensive experimental results show that the generated adversarial point clouds are significantly more imperceptible after applying the LBC-based constraints, outperforming those generated by state-of-the-art methods.

Overall, our contribution is summarized as follows:

- We are the first to explore imperceptible adversarial attacks on point clouds from a local parametric view.
- We develop novel constraints for adversarial attacks that restrict barycentric coordinates of point clouds partitioned using a permutohedral lattice.
- We demonstrate through experiments that adversarial attacks employing our LBC-based constraints achieve superior imperceptibility performance.

Related Work

Adversarial Attacks on 3D Point Clouds. Point cloud attacks are categorized into three types: addition-based, introducing independent points to induce errors (Xiang, Qi, and Li 2019); deletion-based, involving the removal of critical points to affect classification (Zheng et al. 2019; Yang et al. 2019); and perturbation-based, which involves altering existing points to facilitate attacks (Xiang, Qi, and Li 2019; Tang et al. 2022b, 2023c, 2024b,e,c; Wang et al. 2025). This paper specifically focuses on perturbation-based methods.

Xiang, Qi, and Li (2019) and Liu, Yu, and Su (2019) pioneered perturbation-based 3D adversarial attacks by extending C&W attack (Carlini and Wagner 2017) and FGSM (Goodfellow, Shlens, and Szegedy 2015). Zhao et al. (2020) introduced an isometric transformation attack using simple rotations instead of altering individual points. Kim et al. (2021) aimed to perturb only a minimal subset rather than all points. Generative solutions extend beyond point-coordinate perturbations, with approaches such as noise injection into latent features (Lee et al. 2020) or the use of generative adversarial networks (GANs) (Zhou et al. 2020). To facilitate imperceptibility, most adversarial attack solutions apply constraints to restrict the perturbation.

Constraints for Imperceptible Adversarial Attacks. The most commonly used constraints to ensure the imperceptibility of adversarial attacks on point clouds include restrictions on the l_2-norm, Chamfer distance, and Hausdorff distance between the original and adversarial point clouds (Xiang, Qi, and Li 2019). Beyond these standard constraints, GeoA^3 (Wen et al. 2022) maintains local curvatures after the attack. Liu and Hu (2023) constrained the perturbation direction of each point to its normal vector, and Huang et al. (2022) directed the perturbation along the tangent plane. These constraints were adaptively relaxed by Tang et al. (2023b) to be near the normal or tangential direction. The most related work is (Tang et al. 2024a), which also adopts a parametric perspective to restrict manifold property changes. Unlike their global parameterization, our local approach allows finer constraint control.

3D DNNs. Deep learning (Fang et al. 2024, 2023; Tang et al. 2024f) for point cloud classification has evolved from voxel-based methods (Maturana and Scherer 2015) to direct point processing (Qi et al. 2017a) and advanced architectures, including hierarchical models (Qi et al. 2017b), graph CNNs (Wang et al. 2019), and their combinations (Tang et al. 2023a). We aim to attack these DNNs imperceptibly.

Point Cloud Parameterization. 3D parameterization (Zhu et al. 2022) maps a point cloud onto a simpler domain, with applications in texturing, remeshing, and morphing. Global methods map the entire point set to a single domain (Barhak and Fischer 2001; Zhang et al. 2010), while local approaches divide the problem into smaller subproblems (Tewari, Gotsman, and Gortler 2006). In this paper, we devise a local parametric space to capture and constrain manifold properties of 3D point clouds, enabling imperceptible attacks.

Problem Formulation

Preliminary on Adversarial Attacks. Given an object point cloud $\mathcal{P} \in \mathbb{R}^{n\times 3}$ and its label $y \in \{1, ..., Z\}$, adversarial attack aims to mislead a 3D DNN classifier f by feeding it an adversarial point cloud $\mathcal{P}^{'}$, generated by applying a carefully designed perturbation σ to $\mathcal{P}$. Formally, the perturbation σ is obtained by solving the following optimization problem, typically via gradient descent:

$$\min_{\sigma} L_{mis}(f, \mathcal{P}+\sigma, y) + \lambda_1 D_e(\mathcal{P}, \mathcal{P}+\sigma), \qquad (1)$$

where $L_{mis}(\cdot,\cdot,\cdot)$ is the loss function promoting misclassification (e.g., the negation of cross-entropy loss), $D_e(\cdot,\cdot)$ represents constraints from the Euclidean perspective to ensure imperceptibility, and λ_1 is a weighting parameter. This study primarily focuses on untargeted attacks, and targeted attacks are also possible.

Common metrics for $D_e(\cdot,\cdot)$ include the l_2-norm, Chamfer distance, Hausdorff distance, and curvature deviation. Since point clouds are essentially manifolds, neglecting changes in manifold properties make the attack perceptible.

Adversarial Attacks with Parametric Constraints. To address changes in manifold properties, one viable solution is to impose parametric constraints:

$$\min_{\sigma} L_{mis}(f, \mathcal{P}+\sigma, y) + \lambda_1 D_e(\mathcal{P}, \mathcal{P}+\sigma) + \lambda_2 D_p^g(\mathcal{U}, \mathcal{U}^{'}), \qquad (2)$$

where $D_p^g(\mathcal{U}, \mathcal{U}^{'})$ is the constraint applied to the representations of $\mathcal{P}$ and $\mathcal{P}^{'}$ in the global parametric space, denoted as $\mathcal{U}$ and $\mathcal{U}^{'}$, respectively, and λ_2 is a weighting parameter.

Based on this formulation, Tang et al. (2024a) implement $\mathcal{U}$ as a fixed simple shape, such as a sphere or plane, and require that it be bijective with $\mathcal{P}$.

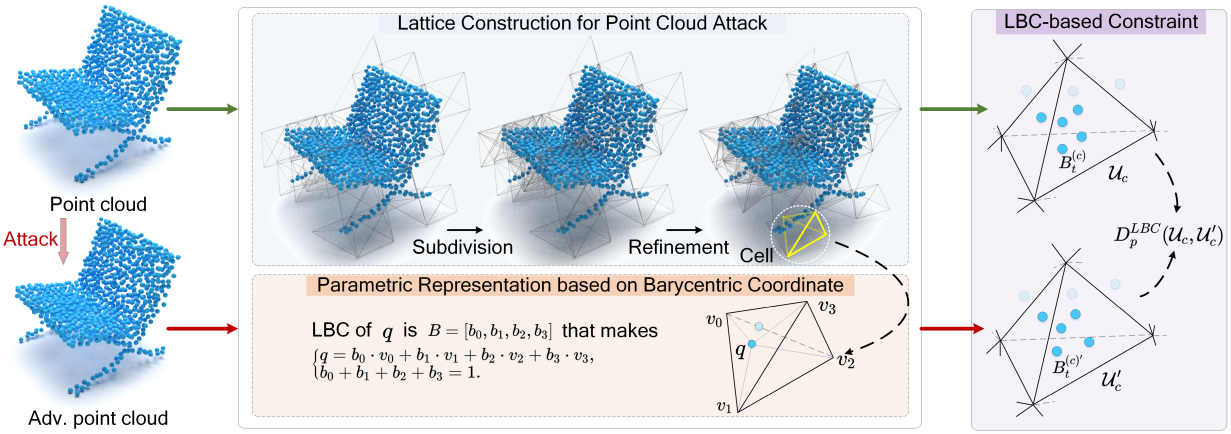

Figure 2: Illustration of our lattice-based barycentric coordinate (LBC)-constrained imperceptible adversarial attacks. Given an input point cloud, we construct a permutohedral lattice, followed by subdivision and refinement. The barycentric coordinates of points within each cell of the lattice are then used to apply the local parametric constraint during attacks.

Discussion. Using a fixed simple shape is a form of global parametrization solution. It is challenging, especially when dealing with complex shapes, and can easily introduce errors. This, in turn, reduces the effectiveness of constraints in preserving manifold properties during adversarial attacks.

Our Solution with Local Parametric Constraints. To fully utilize shape parametrization for preserving manifold properties, we propose imposing local parametric constraints for adversarial attacks, instead of using global ones:

$$\min_{\sigma} L_{mis}(f, \mathcal{P}+\sigma, y) + \lambda_1 D_e(\mathcal{P}, \mathcal{P}+\sigma) + \lambda_2 \sum_{i=1}^{I} D_p^l(\mathcal{U}_i, \mathcal{U}_i^{'}), \tag{3}$$

where $D_p^l(\mathcal{U}_i, \mathcal{U}_i^{'})$ represents the local parametric constraint applied to the i-th local patches of $\mathcal{P}$ and $\mathcal{P}^{'}$.

In this paper, we employ a permutohedral lattice to partition the point cloud into multiple cells, where barycentric coordinates (Zhang et al. 2014) are computed as the local parametric representation. By decomposing complex shapes into multiple simple patches, this approach better captures and constrains manifold properties during adversarial attacks, thereby achieving enhanced imperceptibility.

Method

In this section, we describe the construction of the permutohedral lattice to extract barycentric coordinates for the local parametric representation of point clouds. Next, we define local parametric constraints based on these coordinates and apply them to achieve imperceptible adversarial attacks on 3D point clouds. Please refer to Fig. 2 for an illustration.

Lattice-based Parametric Representation

Preliminary on Permutohedral Lattice. The d-dimensional permutohedral lattice A_d^* is formed by projecting the scaled regular grid $(d+1)\mathbb{Z}^{d+1}$ along the vector $\vec{1} = [1, \ldots, 1]$ onto the hyperplane $H_d : \vec{x} \cdot \vec{1} = 0$, which is the subspace of $\mathbb{R}^{d+1}$ where the coordinates sum to zero. The Delaunay cells of the permutohedral lattice are d-simplices, and these simplices tessellate H_d.

In particular, A_3^*, the three-dimensional permutohedral lattice, is widely used for processing point clouds (Baek and Adams 2009). Its ability to subdivide space into tetrahedra makes it suitable for efficiently representing and manipulating the spatial structure of point clouds.

Lattice Construction and Subdivision. We lift the 3D point clouds to 4D space by adding a linearly dependent basis, then partition it using a regular 4D grid with a resolution of r along each axis, following (Gu et al. 2019). Next, both the points and the grid are projected along $\vec{1} = [1, 1, 1, 1]$ onto a 3D space to form the permutohedral lattice with C cells, denoted as $\{V_c\}_{c=1:C}$, with the points within each lattice cell denoted as $\{P_t^{(c)}\}_{t=1:t_c}$.

Due to the high density or complex geometric characteristics in some cells, further subdivision is required to better capture manifold properties. Specifically, we design a subdivision criterion that considers curvature and density:

$$S(V_c) = \frac{1}{t_c} \sum_{t=1}^{t_c} S_{cur}(P_t^{(c)}) + \lambda_d S_{den}(P_t^{(c)}), \tag{4}$$

where $S_{cur}(P_t^{(c)})$ and $S_{den}(P_t^{(c)})$ represent the curvature and density of the local region centered at $P_t^{(c)}$, respectively, and λ_d is a weighting parameter. When $S(V_c)$ exceeds a threshold τ, we further subdivide the cell V_c.

Specifically, for each cell to be subdivided, we first compute the centroid of all points within it and then connect the centroid to the four vertices of the tetrahedron, thus dividing the original tetrahedron into four smaller tetrahedra. This process is repeated until the value of $S(\cdot)$ for each cell is below the threshold. Furthermore, we remove all empty cells. For simplicity, we continue to denote these cells as $\{V_c\}_{c=1:C}$ after subdivision and cleaning.

Lattice Refinement. Due to the uneven distribution of points within some cells, such as clustering near a corner,

we further refine the positions of the lattice vertices via:

$$\min_{\{V_c\}_{c=1:C}} \frac{1}{C} \sum_{c=1}^{C} \left(\sum_{t=1}^{t_c} D_{dis}(V_c, P_t^{(c)}) + \lambda_a D_{ang}(\theta_c) \right) \quad (5)$$

where $D_{dis}(V_c, P_t^{(c)})$ calculates the average distance from point $P_t^{(c)}$ to the four faces of the tetrahedral cell V_c, $D_{ang}(\theta_c)$ measures the cumulative variation in the four trihedral angles θ_c, and λ_a is a weighting parameter.

By optimizing the above equation, we aim to distribute the points more evenly within the cells while minimizing changes in cell angles.

Extracting Barycentric Coordinates. Based on the subdivided and refined lattice, we recompute the barycentric coordinates for each point within the cells, following (Adams, Baek, and Davis 2010). This results in the set $\{\mathcal{U}_c\}_{c=1:C}$, where $\mathcal{U}_c = \{B_t^{(c)}\}_{t=1:t_c}$. These lattice-based barycentric coordinates (LBCs) serve as the local parametric representations of the original point clouds.

LBC-constrained Imperceptible Attacks

LBC-based Constraints. Based on the extracted lattice-based barycentric coordinates (LBCs), we implement the local parametric constraint in Eqn. 3 for lattice V_c:

$$D_p^{l^*}(\mathcal{U}_c, \mathcal{U}_c^{'}) = \frac{1}{t_c} \sum_{t=1}^{t_c} ||B_t^{(c)} - B_t^{(c)^{'}}||_2. \quad (6)$$

By applying constraints across all lattice cells, we can better preserve the manifold properties of point clouds.

Usage for Imperceptible Adversarial Attacks. With the LBC-based constraints defined above, the final objective for imperceptible adversarial attacks can be formulated by incorporating $D_p^{l^*}$ from Eqn. 6 into Eqn. 3:

$$\min_{\sigma} L_{mis}(f, \mathcal{P}+\sigma, y) + \lambda_1 D_e(\mathcal{P}, \mathcal{P}+\sigma) + \lambda_2 \sum_{c=1}^{C} D_p^{l^*}(\mathcal{U}_c, \mathcal{U}_c^{'}). \quad (7)$$

Therefore, the imperceptible adversarial perturbation σ can be obtained by solving the above equation.

Note that, our LBC-based constraints can be applied to various adversarial attack methods, such as IFGM and C&W, to achieve enhanced imperceptibility.

Experimental Results

Experimental Setup

Implementation. We implement the LBC-constrained adversarial attack framework using PyTorch (Paszke et al. 2019). Following (Gu et al. 2019), we construct a permutohedral lattice with a resolution of $r = 10$ along each axis. For lattice subdivision, we compute each cell's density score and mean curvature, normalize them to [0,1], and calculate the subdivision criterion $S(\cdot)$ as their weighted sum with $\lambda_d = 0.5$. The subdivision threshold τ is set so that the top one-third of cells are subdivided. During lattice refinement, we optimize vertex positions over 2,000 iterations, setting $\lambda_a = 10.0$ to maintain cell angles. Experiments are conducted on a workstation with dual 2.40 GHz CPUs, 128 GB RAM, and eight NVIDIA RTX 3090 GPUs.

Datasets. We evaluate on ModelNet40 (Wu et al. 2015) and ShapeNet Part (Chang et al. 2015), sampling 1,024 points per cloud following (Xiang, Qi, and Li 2019).

Our Attack Solutions. We incorporate our LBC-based constraints into two classic attack methods: IFGM (Dong et al. 2020) and C&W (Xiang, Qi, and Li 2019), resulting in two distinct configurations: IFGM-LBC and C&W-LBC.

Baseline Attack Methods. We select eight baselines: gradient-based methods IFGM and PGD (Dong et al. 2020), direction-based methods SI-Adv (Huang et al. 2022) and ITA (Liu and Hu 2023), optimization-based methods GeoA^3 (Wen et al. 2022) and 3d-Adv (Xiang, Qi, and Li 2019), and manifold constraint configurations IFGM-MC and C&W-MC (Tang et al. 2024a).

Victim 3D DNN Classifiers. We use PointNet (Qi et al. 2017a), PointNet++ (Qi et al. 2017b), DGCNN (Wang et al. 2019), and PointConv (Wu, Qi, and Fuxin 2019) as victim models. These models are trained following the procedures outlined in their original papers.

Evaluation Setting and Metrics. To ensure fair comparisons, we configure each attack method to achieve its maximum attack success rate (ASR), defined as the percentage of adversarial point clouds that successfully deceive the victim model. Under this maximal adversarialness condition (Tang et al. 2024a), we evaluate the imperceptibility of the attacks. Specifically, we employ six widely recognized metrics: Chamfer distance (CD) (Fan, Su, and Guibas 2017), Hausdorff distance (HD) (Taha and Hanbury 2015), 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). Unless explicitly mentioned, all discussions regarding attack results refer to untargeted attacks.

Comparison and Performance Analysis

Imperceptibility from the Euclidean Perspective. We report the ASR and imperceptibility results of various adversarial attack methods tested on ModelNet40 and ShapeNet Part in Tab. 1. The results indicate that all adversarial attack solutions achieve a 100% ASR. However, the imperceptibility varies significantly across different methods. Methods specifically designed for imperceptibility, such as ITA (Liu and Hu 2023), require relatively smaller perturbation sizes. The two configurations of manifold constraints (MC) (Tang et al. 2024a) also perform well, showing significant improvement over the original IFGM and C&W methods, with some cases achieving the highest levels of imperceptibility. Notably, our approach consistently demonstrates superior performance, particularly with the IFGM-LBC configuration, which shows the best results in most cases. This validates the effectiveness and superiority of our method.

Imperceptibility from the Manifold Perspective. To provide a more comprehensive comparison, we also measure changes in the manifold properties of different attacks. Specifically, we back-map adversarial point clouds generated by different attack methods to the parameter sphere using a pretrained manifold mapping network, and then

Model	Attack	ModelNet40							ShapeNet Part						
		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})
PointNet	PGD	100	7.155	5.025	0.981	0.302	1.624	2.315	100	13.172	17.068	1.569	0.521	3.679	3.358
	IFGM	100	4.039	5.565	0.789	0.314	0.775	0.864	100	3.328	10.269	0.785	0.408	0.619	0.556
	GeoA^3	100	4.646	0.497	1.307	0.121	0.396	2.319	100	7.531	1.444	2.655	0.146	0.465	4.104
	3d-Adv	100	6.115	4.372	0.863	0.250	1.215	1.410	100	15.659	5.495	1.787	0.279	4.006	3.693
	SI-Adv	100	2.768	2.595	0.731	0.220	0.271	0.725	100	3.435	3.692	0.881	0.233	0.441	0.825
	ITA	100	2.747	0.414	0.534	0.122	0.555	1.214	100	5.872	1.917	1.002	0.181	1.016	2.035
	C&W-MC	100	3.198	0.263	0.597	0.120	1.292	1.129	100	5.840	**0.654**	2.060	0.138	3.489	2.992
	IFGM-MC	100	0.749	**0.208**	0.264	0.119	0.478	0.472	100	1.257	0.940	0.432	0.147	1.009	0.517
	C&W-LBC	100	0.895	0.246	0.277	**0.113**	0.218	0.462	100	1.403	0.924	0.473	**0.137**	0.477	0.457
	IFGM-LBC	100	**0.703**	0.356	**0.245**	0.122	**0.185**	**0.283**	100	**1.218**	1.183	**0.406**	0.151	**0.381**	**0.359**
PointNet++	PGD	100	5.182	0.636	0.753	0.125	1.508	2.146	100	10.090	3.257	1.342	0.215	3.969	3.328
	IFGM	100	3.558	1.162	0.640	0.146	1.149	1.454	100	4.532	3.608	0.548	0.220	1.824	1.584
	GeoA^3	100	6.579	0.461	1.615	0.114	0.762	2.919	100	7.701	0.847	2.875	0.105	1.375	4.176
	3d-Adv	100	8.915	3.564	1.535	0.141	1.288	2.784	100	9.564	3.778	2.014	0.197	3.021	3.590
	SI-Adv	100	9.399	2.377	1.422	0.185	1.061	2.684	100	9.266	3.233	1.535	0.203	1.146	2.811
	ITA	100	6.792	0.708	0.998	0.121	3.533	2.272	100	5.202	0.802	0.999	0.110	3.423	2.152
	C&W-MC	100	3.418	0.154	0.586	0.136	1.300	1.605	100	7.070	0.404	1.054	0.108	3.773	2.745
	IFGM-MC	100	**0.232**	0.075	0.159	0.135	0.419	0.353	100	**0.986**	0.470	0.362	0.124	1.264	0.741
	C&W-LBC	100	2.131	0.198	0.449	0.105	0.547	1.155	100	2.150	0.476	0.582	0.105	1.102	1.173
	IFGM-LBC	100	0.339	**0.068**	**0.157**	**0.099**	**0.111**	**0.346**	100	1.057	**0.403**	**0.358**	**0.101**	**0.576**	**0.693**
DGCNN	PGD	100	19.968	5.098	1.933	0.267	4.924	4.785	100	63.556	27.557	5.224	0.511	7.275	9.233
	IFGM	100	15.791	12.391	1.622	0.363	2.849	3.777	100	19.623	26.040	2.069	0.504	4.954	4.387
	GeoA^3	100	7.566	0.546	1.585	0.119	0.741	3.083	100	27.612	3.748	5.798	0.199	1.695	7.502
	3d-Adv	100	10.345	3.807	3.589	0.227	5.997	6.685	100	21.553	8.531	2.258	0.282	5.119	4.628
	SI-Adv	100	7.146	1.691	1.087	0.143	0.666	2.495	100	11.685	3.019	1.772	0.160	2.054	3.646
	ITA	100	3.249	0.524	0.552	0.114	0.971	1.359	100	27.633	4.597	2.492	0.244	3.847	4.696
	C&W-MC	100	8.117	0.314	0.980	0.119	2.630	2.724	100	18.103	1.678	2.104	0.153	6.323	4.802
	IFGM-MC	100	1.425	**0.262**	0.370	0.115	0.687	0.907	100	3.509	1.217	0.713	0.137	2.249	1.492
	C&W-LBC	100	2.846	0.336	0.569	0.113	0.643	1.425	100	6.491	**1.208**	1.197	**0.136**	2.161	2.387
	IFGM-LBC	100	**1.332**	0.292	**0.318**	**0.111**	**0.264**	**0.765**	100	**3.410**	1.754	**0.685**	0.144	**1.150**	**1.446**
PointConv	PGD	100	14.551	2.216	1.442	0.184	3.491	3.862	100	42.202	9.949	3.784	0.252	6.866	7.277
	IFGM	100	7.959	2.608	1.015	0.184	1.741	2.427	100	16.139	8.776	1.812	0.231	3.526	3.807
	GeoA^3	100	6.809	0.644	2.169	0.119	1.119	3.556	100	9.383	1.222	4.224	0.120	**1.190**	5.391
	3d-Adv	100	11.213	1.763	1.179	0.163	3.279	2.807	100	21.034	3.687	2.277	0.193	4.912	4.548
	SI-Adv	100	6.060	1.784	0.977	0.144	0.576	2.081	100	11.281	3.500	1.741	0.165	1.949	3.514
	ITA	100	5.539	0.480	0.833	0.111	1.904	1.971	100	9.082	1.452	1.375	0.146	3.645	2.925
	C&W-MC	100	5.947	0.294	0.806	**0.106**	1.710	2.393	100	9.543	1.302	2.229	**0.114**	6.126	5.468
	IFGM-MC	100	1.533	0.288	0.355	0.114	0.726	0.909	100	4.751	1.262	0.809	0.125	2.458	1.795
	C&W-LBC	100	2.312	**0.248**	0.483	0.107	0.567	1.232	100	5.070	**1.101**	0.992	0.125	1.800	1.885
	IFGM-LBC	100	**1.366**	0.344	**0.316**	**0.106**	**0.367**	**0.731**	100	**3.918**	1.946	**0.736**	0.131	1.347	**1.484**

Table 1: Comparison on the perturbation sizes required by different methods to reach their highest achievable ASR. The evaluation is conducted across different DNN classifiers on ModelNet40 and ShapeNet Part.

quantitatively analyze the manifold-aware distortion following (Tang et al. 2024a). Additionally, we partition the point clouds using a lattice structure and calculate the displacement of barycentric coordinates in each lattice cell using Eqn. 6. The results in Tab. 2 reveals that our methods consistently outperform state-of-the-art techniques under both metrics, including the two variants of MC, i.e., IFGM-MC and C&W-MC, by a considerable margin, validating the effectiveness of local parametric constraints. Notably, IFGM-LBC achieves the best performance in most cases.

Visualization. We visualize adversarial point clouds generated by various attack methods aimed at fooling PointNet, as shown in Fig. 3. The visualizations reveal that adversarial point clouds produced by most baseline methods exhibit noticeable outliers. The manifold constraints (MC) (Tang et al. 2024a), designed to address manifold-aware distortions, result in fewer outliers. However, due to the challenges in achieving accurate global parameterization, some cases still display outliers. By imposing constraints in the local parametric space, both IFGM-LBC and C&W-LBC show a significant reduction in outliers, with IFGM-LBC demonstrating the best performance, validating its effectiveness.

Ablation Studies and Other Analysis

Importance of Lattice Subdivision and Refinement. To validate the importance of lattice subdivision and refinement in enhancing the imperceptibility of adversarial attacks, we visualize the results of point cloud partitioning before and after these operations. As shown in Fig. 4, subdivision refines coarse cells into finer partitions, especially in complex

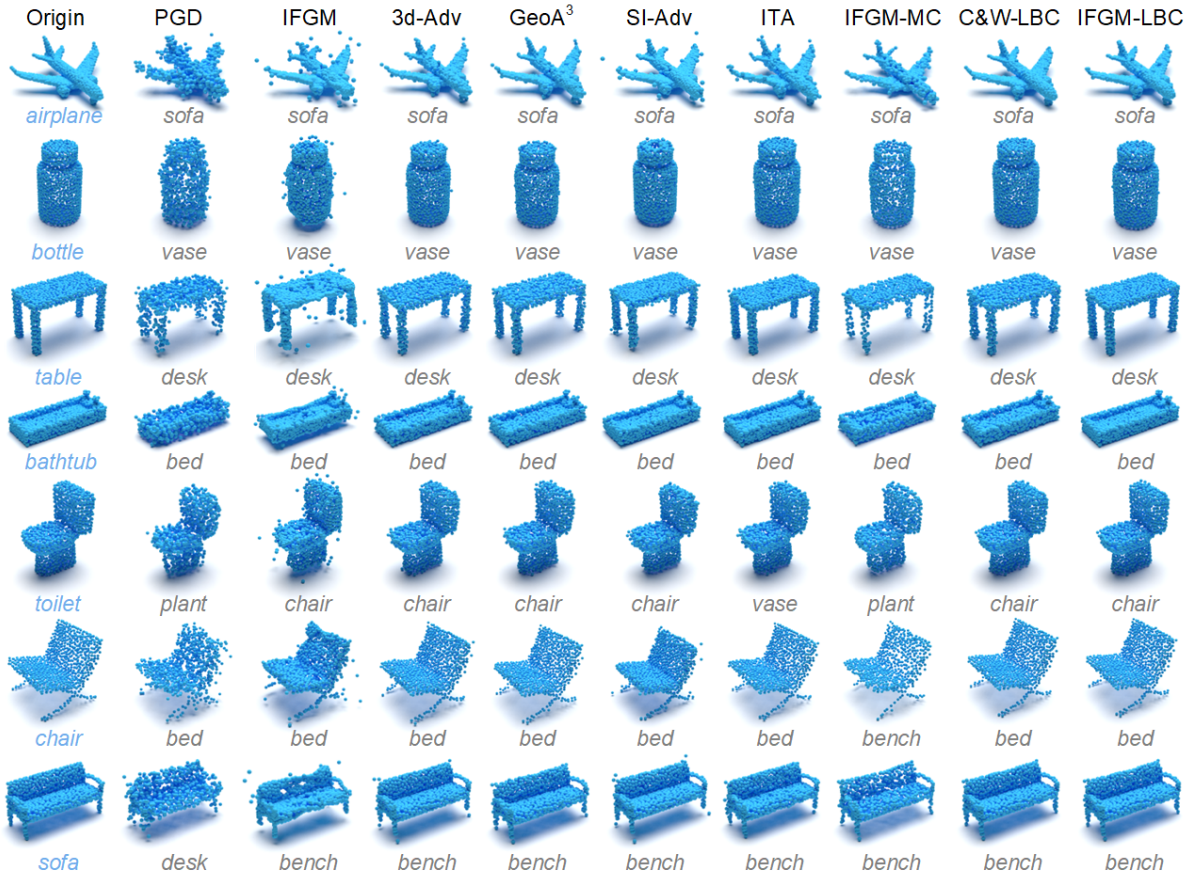

Figure 3: Visualizations of original and adversarial point clouds generated to fool PointNet on ModelNet40 by various adversarial attack methods. The ground truth and predicted labels are marked in blue and gray below the images.

Attack	Sphere									Lattice								
	PointNet			DGCNN			PointConv			PointNet			DGCNN			PointConv		
	l_2	CD (10^{-4})	HD (10^{-2})	l_2	CD (10^{-4})	HD (10^{-2})	l_2	CD (10^{-4})	HD (10^{-2})	l_2	CD (10^{-4})	HD (10^{-2})	l_2	CD (10^{-4})	HD (10^{-2})	l_2	CD (10^{-4})	HD (10^{-2})
PGD	2.094	31.622	1.867	2.149	32.392	2.038	2.150	32.527	1.986	7.381	19.277	1.815	7.405	19.267	1.804	7.431	19.397	1.806
IFGM	1.704	21.752	1.945	1.802	24.321	1.920	1.804	24.381	1.874	2.972	6.411	1.397	3.709	15.358	1.407	3.950	16.163	1.633
GEO-A^3	1.280	13.103	1.514	1.499	18.347	1.550	1.339	15.242	1.467	1.639	8.364	0.743	3.275	14.770	1.064	2.722	14.307	1.010
3d-Adv	1.086	9.690	1.394	1.518	18.767	1.647	1.348	14.810	1.490	0.894	3.730	0.656	2.732	13.278	0.921	1.723	11.366	0.657
SI-Adv	1.216	12.423	1.583	1.741	23.111	1.899	1.441	17.093	1.556	1.707	2.807	0.889	4.359	13.630	1.261	1.430	16.854	1.554
ITA	1.254	12.480	1.485	1.407	16.158	1.469	1.411	16.426	1.455	1.405	6.190	0.810	2.193	10.801	0.974	2.169	11.853	0.945
C&W-MC	1.318	15.089	1.425	1.701	22.438	1.768	1.635	20.990	1.690	1.933	7.072	1.008	2.934	16.119	1.000	2.468	16.935	0.910
IFGM-MC	1.177	10.500	1.521	1.587	19.387	1.809	1.572	19.159	1.670	0.839	2.719	0.722	1.163	6.328	0.676	1.162	6.729	0.619
C&W-LBC	**1.020**	9.171	**1.280**	1.245	13.514	**1.377**	1.253	13.381	1.426	0.866	3.383	**0.600**	1.768	10.039	0.747	1.617	10.077	0.687
IFGM-LBC	1.023	**8.658**	1.346	**1.134**	**10.708**	1.394	**1.108**	**10.347**	**1.367**	**0.748**	**1.686**	0.601	**0.999**	**5.051**	**0.530**	**0.983**	**4.767**	**0.534**

Table 2: Comparison on the manifold property changes required by different methods to reach their highest achievable ASR. The evaluation is conducted on ModelNet40, measured using manifold-aware distortion with a sphere as the parameter shape (Sphere) and the displacemen of barycentric coordinates under our permutohedral lattice-based partition (Lattice).

regions like object junctions, while lattice refinement further aligns the cell edges with the point cloud boundaries. The results in Tab. 3 show that omitting these processes leads to larger distortion in both C&W and IFGM methods, reducing imperceptibility. This analysis highlights the critical role of both subdivision and refinement in enhancing the imperceptibility of adversarial point clouds.

Permutohedral Lattice vs. Regular Volume for Partition. We also explore the impact of different partitioning solutions on the imperceptibility of adversarial attacks, specifically the permutohedral lattice and regular volume. To ensure fairness, we use the same partitioning resolution for

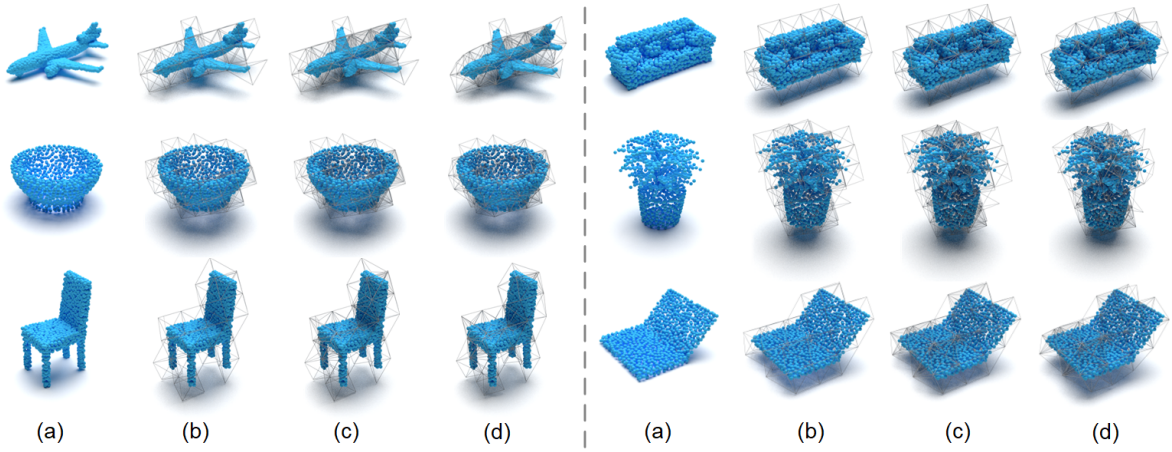

Figure 4: Visualizations of (a) original point clouds, (b) initially constructed permutohedral lattice, (c) lattice after subdivision, and (d) lattice after refinement, used for partitioning point clouds.

S	R	C&W-LBC l_2	C&W-LBC CD (10^{-4})	C&W-LBC Curv (10^{-2})	C&W-LBC EMD (10^{-2})	IFGM-LBC l_2	IFGM-LBC CD (10^{-4})	IFGM-LBC Curv (10^{-2})	IFGM-LBC EMD (10^{-2})
✓		0.285	0.920	0.219	0.473	0.251	0.711	0.186	0.284
	✓	0.288	0.938	0.237	0.499	0.257	0.739	0.195	0.301
✓	✓	**0.277**	**0.895**	**0.218**	**0.462**	**0.245**	**0.703**	**0.185**	**0.283**

Table 3: Comparison of the perturbation sizes introduced by different variants of our LBC-constrained attacks, with and without lattice subdivision (S) and lattice refinement (R), when attacking PointNet on ModelNet40.

		Partition	ASR (%)	CD (10^{-4})	HD (10^{-2})	l_2	GR	Curv (10^{-2})	EMD (10^{-2})
ModelNet40	C&W	-	100	1.529	0.322	0.612	0.120	0.397	0.880
		Volume	100	1.373	0.315	0.426	0.118	0.350	0.730
		Lattice	100	**1.007**	**0.243**	**0.302**	**0.112**	**0.253**	**0.527**
	IFGM	-	100	4.039	5.565	0.789	0.314	0.775	0.864
		Volume	100	1.098	**0.290**	0.304	**0.116**	0.262	0.419
		Lattice	100	**0.871**	0.294	**0.283**	0.117	**0.231**	**0.365**
ShapeNet Part	C&W	-	100	2.754	0.975	1.480	0.130	1.137	1.924
		Volume	100	2.579	0.946	0.872	**0.121**	0.935	1.264
		Lattice	100	**1.726**	**0.853**	**0.538**	0.129	**0.608**	**0.605**
	IFGM	-	100	3.328	10.269	0.785	0.408	0.619	0.556
		Volume	100	1.452	2.141	0.449	0.151	0.458	0.438
		Lattice	100	**1.349**	**1.180**	**0.430**	**0.149**	**0.423**	**0.403**

Table 4: Comparison of the ASR and perturbation sizes introduced by baseline methods and two variants of local parametric constrained attacks with different partitioning solutions, including regular volume and permutohedral lattice, when attacking PointNet. Note: both configurations are evaluated without subdivision and refinement.

both methods and do not apply subdivision or refinement. For the regular volume, we employ a normalized local coordinate system within each cube. The results in Tab. 4 show that using local parametric constraints with both partitioning methods enhances the imperceptibility of attacks. However,

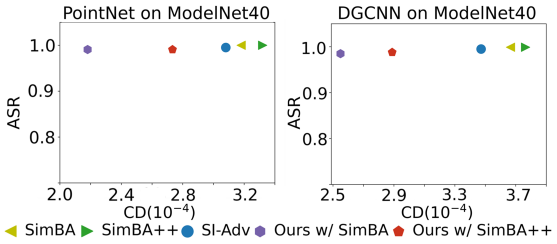

Figure 5: The trade-off between ASR and imperceptibility, measured by CD, of different black-box methods in attacking PointNet and DGCNN on ModelNet40.

the improvement is more significant with the permutohedral lattice partition, validating the superiority of lattice-based local parametric constraints.

Analysis on Black-box Setting. We also investigate the applicability of our method in the black-box setting. Although our approach is not specifically designed for black-box scenarios, it can be seamlessly integrated into existing black-box attack frameworks. Specifically, we incorporate our method into SimBA (Guo et al. 2019) and SimBA++ (Yang et al. 2020). The results drawn in Fig. 5 demonstrate that both SimBA and SimBA++ combined with our method achieves nearly 100% ASR in black-box settings, with significantly lower distortion. This validates the generalization capability of our solution in black-box scenarios.

Conclusion

This paper has introduced novel local parametric constraints for achieving imperceptible adversarial attacks on point clouds. The rationale involves preserving manifold properties by constraining lattice-based barycentric coordinates partitioned with a permutohedral lattice. Comprehensive experiments validate the effectiveness of our LBC-constrained attacks in achieving high imperceptibility. In future work, we plan to enhance the black-box capability of point cloud attacks while maintaining this level of imperceptibility.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (62472117, 62025207, U2436208), the Guangdong Basic and Applied Basic Research Foundation (2024A1515012064), the Science and Technology Projects in Guangzhou (2025A03J0137), the Academician Binxing Fang's Specialized Class, and the Huangpu Research School of Guangzhou University.

References

Adams, A.; Baek, J.; and Davis, M. A. 2010. Fast high-dimensional filtering using the permutohedral lattice. In *Computer Graphics Forum (Proc. of Eurographics)*, volume 29, 753–762.

Baek, J.; and Adams, A. 2009. Some useful properties of the permutohedral lattice for Gaussian filtering. Technical report, Stanford University.

Bai, T.; Luo, J.; Zhao, J.; Wen, B.; and Wang, Q. 2021. Recent Advances in Adversarial Training for Adversarial Robustness. In *IJCAI*, 4312–4321.

Barhak, J.; and Fischer, A. 2001. Parameterization and reconstruction from 3D scattered points based on neural network and PDE techniques. *IEEE TVCG*, 7(1): 1–16.

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

Chang, A. X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*.

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.

Fang, X.; Easwaran, A.; Genest, B.; and Suganthan, P. N. 2024. Your Data Is Not Perfect: Towards Cross-Domain Out-of-Distribution Detection in Class-Imbalanced Data. *ESWA*.

Fang, X.; Liu, D.; Zhou, P.; and Nan, G. 2023. You can ground earlier than see: An effective and efficient pipeline for temporal sentence grounding in compressed videos. In *CVPR*.

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

Gu, X.; Wang, Y.; Wu, C.; Lee, Y. J.; and Wang, P. 2019. Hplflownet: Hierarchical permutohedral lattice flownet for scene flow estimation on large-scale point clouds. In *CVPR*, 3254–3263.

Guo, C.; Gardner, J.; You, Y.; Wilson, A. G.; and Weinberger, K. 2019. Simple black-box adversarial attacks. In *ICML*, 2484–2493.

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.

Jiang, H.; Wang, Z.; Jin, Y.; Chen, X.; Li, P.; Gan, Y.; Lin, S.; and Chen, X. 2021. Hierarchical control of soft manipulators towards unstructured interactions. *IJRR*, 40(1): 411–434.

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*.

Lin, N.; Li, Y.; Tang, K.; Zhu, Y.; Zhang, X.; Wang, R.; Ji, J.; Chen, X.; and Zhang, X. 2022. Manipulation planning from demonstration via goal-conditioned prior action primitive decomposition and alignment. *IEEE Robotics and Automation Letters*, 7(2): 1387–1394.

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.

Maturana, D.; and Scherer, S. 2015. Voxnet: A 3d convolutional neural network for real-time object recognition. In *IROS*, 922–928.

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.

Spanier, E. H. 1989. *Algebraic topology*. Springer Science & Business Media.

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.; Chen, Y.; Peng, W.; Zhang, Y.; Fang, M.; Wang, Z.; and Song, P. 2023a. Reppvconv: attentively fusing reparameterized voxel features for efficient 3d point cloud perception. *The Visual Computer*, 39(11): 5577–5588.

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. 2024c. Imperceptible Adversarial Attacks on Point Clouds Guided by Point-to-Surface Field. arXiv:2412.19015.

Tang, K.; Lou, T.; Peng, W.; Chen, N.; Shi, Y.; and Wang, W. 2024d. Effective Single-Step Adversarial Training With Energy-Based Models. *TETCI*, 8(5): 3396–3407.

Tang, K.; Ma, Y.; Miao, D.; Song, P.; Gu, Z.; Tian, Z.; and Wang, W. 2022a. Decision Fusion Networks for Image Classification. *TNNLS*, 1–14.

Tang, K.; Shi, Y.; Lou, T.; Peng, W.; He, X.; Zhu, P.; Gu, Z.; and Tian, Z. 2023b. Rethinking Perturbation Directions for Imperceptible Adversarial Attacks on Point Clouds. *IEEE Internet of Things Journal*, 10(6): 5158–5169.

Tang, K.; Shi, Y.; Wu, J.; Peng, W.; Khan, A.; Zhu, P.; and Gu, Z. 2022b. NormalAttack: Curvature-aware shape deformation along normals for imperceptible point cloud attack. *Security and Communication Networks*, 2022.

Tang, K.; Wang, Z.; Peng, W.; Huang, L.; Wang, L.; Zhu, P.; Wang, W.; and Tian, Z. 2024e. 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. 2023c. Deep Manifold Attack on Point Clouds via Parameter Plane Stretching. In *AAAI*, volume 37, 2420–2428.

Tang, K.; Zhao, W.; Peng, W.; Fang, X.; Cui, X.; Zhu, P.; and Tian, Z. 2024f. Reparameterization Head for Efficient Multi-Input Networks. In *ICASSP*, 6190–6194.

Tewari, G.; Gotsman, C.; and Gortler, S. J. 2006. Meshing genus-1 point clouds using discrete one-forms. *Computers & Graphics*, 30(6): 917–926.

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*.

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.

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

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.; Jiang, Y.; Huang, X.; Ni, B.; and Zhao, C. 2020. Learning black-box attackers with transferable priors and query feedback. In *NeurIPS*, volume 33, 12288–12299.

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*.

You, X.; Zhang, Y.; Chen, X.; Liu, X.; Wang, Z.; Jiang, H.; and Chen, X. 2017. Model-free control for soft manipulators based on reinforcement learning. In *IROS*, 2909–2915.

Zhang, J.; Deng, B.; Liu, Z.; Patanè, G.; Bouaziz, S.; Hormann, K.; and Liu, L. 2014. Local Barycentric Coordinates. *ACM TOG (Proc. of SIGGRAPH Asia)*, 33: 188:1–188:12.

Zhang, L.; Liu, L.; Gotsman, C.; and Huang, H. 2010. Mesh reconstruction by meshless denoising and parameterization. *Computers & Graphics*, 34(3): 198–208.

Zhao, Y.; Wu, Y.; Chen, C.; and Lim, A. 2020. On isometry robustness of deep 3d point cloud models under adversarial attacks. In *CVPR*, 1201–1210.

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.

Zhu, Z.; Iglesias, A.; You, L.; and Zhang, J. J. 2022. A review of 3D point clouds parameterization methods. In *International Conference on Computational Science*, 690–703.