Generating Adversarial Point Clouds Using Diffusion Model

Ruiyang Zhao,Bingbing Zhu Chuxuan Tong, Xiaoyi Zhou, Xi Zheng

ABSTRACT

Adversarial attack methods for 3D point cloud classification reveal the vulnerabilities of point cloud recognition models. This vulnerability could lead to safety risks in critical applications that use deep learning models, such as autonomous vehicles. To uncover the deficiencies of these models, researchers can evaluate their security through adversarial attacks. However, most existing adversarial attack methods are based on white-box attacks. While these methods achieve high attack success rates and imperceptibility, their applicability in real-world scenarios is limited. Black-box attacks, which are more meaningful in real-world scenarios, often yield poor results. This paper proposes a novel black-box adversarial example generation method that utilizes a diffusion model to improve the attack success rate and imperceptibility in the black-box setting, without relying on the internal information of the point cloud classification model to generate adversarial samples. We use a 3D diffusion model to use the compressed features of the point cloud as prior knowledge to guide the reverse diffusion process to add adversarial points to clean examples. Subsequently, its reverse process is employed to transform the distribution of other categories into adversarial points, which are then added to the point cloud. Furthermore, density-aware Chamfer distance is incorporated to constrain the noise added during back-propagation, further improving the imperceptibility of adversarial examples. Experimental results demonstrate that the proposed method exhibits high attack performance against various point cloud recognition models and defense methods, significantly enhancing the effectiveness of black-box attacks. In the black-box scenario, the attack success rate can reach about 90%. The code for this work is available at: https://github.com/AdvPC/Generating-Adversarial-Point-Clouds-Using-Diffusion-Model.

1 INTRODUCTION

Deep neural networks (DNNs) have achieved remarkable success in various computer vision tasks, particularly in processing and analyzing 2D [45, 55] and 3D data [31, 43, 90]. However, studies show that DNNs are vulnerable to adversarial examples (AEs): carefully crafted perturbations added to clean samples that can mislead models without being noticeable to humans [9, 18, 64, 96]. Specifically, for 2D data, such as 2D images, sight color modifications on clean samples can fool recognition models and even pose significant threats to real-world applications [15]. Similar adversaries can also be used on 3D point clouds. Recent research shifted focus to the recognition of 3D point clouds due to their increasing importance in multiple fields [14]. For instance, autonomous driving systems rely on LiDAR sensors to perceive and map the environment, because point clouds offer more precise geometric and structural information than 2D images [79, 101]. However, point cloud models are also found to be susceptible to AEs [2, 12, 35, 61, 62, 69, 95]. Existing methods dominantly use white-box adversaries to maximize attack success rates and imperceptibility, whereas black-box settings remain underexplored on point cloud. In this work, we propose a novel approach for generating point cloud AEs with diffusion models, aiming to enhance attack success rate and stealthiness under black-box settings.

Point clouds are sets of unordered points that describe the shape of objects [74], and recognition models make predictions based on the representation of point cloud [17]. Shape latent is the compression feature of 3D point cloud. Therefore, generated adversarial point clouds are expected to keep shape changes perceptually negligible to humans, but capable of misleading recognition models. Several works extended existing gradient-based and optimization-based methods to point clouds [34, 35, 82], including Fast Gradient Sign Method (FGSM) [34], Projected Gradient Descent (PGD) [35], and Carlini and Wagner (C&W) attack [82]. In these methods, the generated perturbations are adjustable hyperparameters typically measured by Hausdorff distance [94] and Chamfer distance [48]. However, they face a trade-off problem between imperceptibility and attack success rate. Higher attack success rates require more perturbations, while strong perturbations reduce imperceptibility. Another line of work aimed to minimize perturbations by adding, dropping, or shifting existing points. The modifications are based on manually designed rules either in greedy ways [87] or using optimization strategies [42]. These pioneering explorations on point clouds are predominantly in white-box settings, which are less applicable to real-world scenarios compared to black-box settings. Existing black-box attacks are primarily limited to query-based methods but have yet to achieve comparable attack success rates and stealthiness [26, 33, 47, 78]. Given the shortcomings of current adversarial examples generation schemes in black-box scenarios, we rethink the black-box adversarial examples generation method from the perspective of generative models.

In this work, we propose a novel black-box adversarial examples generation method, which uses the reverse diffusion process to add adversarial noise to the clean point cloud to craft adversarial examples. We regard AE generation as a reverse diffusion process, where the distribution of other classes is transformed into an adversarial distribution in the reverse-diffusion process. In addition, to enable the reverse-diffusion process to generate point clouds with obvious shape meanings, we use shape potential as prior knowledge. In the sample generation setting, the adversarial noise comes from the prior knowledge, and we use the normalizing flow [7] to parameterize the prior knowledge and drive the model to obtain strong expressive power. To further improve the

Conference'17, July 2017, Washington, DC, USA
© 2025 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

interference-free execution of adversarial examples, we introduce the density-aware chamfer distance [80] to bind the noise added during the back-propagation process. To evaluate the effectiveness of the black-box adversarial examples generation method using diffusion models, we evaluate our black-box adversarial examples generation method on common 3D point cloud recognition models, including PointNet2 [52], Curvenet [46], PointConv [81] and compare it with optimization-based methods [82] and generation-based methods [6]. The results demonstrate that our method successfully generates AEs capable of simultaneously fooling different point cloud recognition models. The contribution of our work can be summarised as follows:

- We propose a novel guided black-box adversarial method for generating adversarial point clouds. In our approach, point clouds from other classes are encoded into latent representations, which are conditions to guide the recovery process during reverse diffusion. The reverse diffusion process can automatically learn the significant features required to mislead the target model effectively under guidance.
- We leveraged a novel loss function to generate adversarial point clouds with minimal perturbations.
- The experimental data show that we can achieve over 90% attack success rate when facing different point cloud recognition models even with defenses.

We will introduce this paper from the preliminaries, method, experiment results and conclusion.

2 RELATED WORK

We introduced a diffusion model to create adversarial samples and verified it in the commonly used 3D point cloud recognition model. To confirm the robustness of adversarial examples, we used several common defense methods for testing. Details are introduced in Sections 2.1-2.3.

2.1 Point Cloud Recognition

A point cloud is a sparse collection of points sampled by sensors to capture surface details [21]. Each point in the cloud specifies its position in the 3D coordinate system. Due to the unique characteristics of point clouds, models must learn representations from unordered inputs and extract information over both local and global geometric features. PointNet [51] is a benchmark model that directly takes raw point clouds as inputs and ensures permutation invariance through a symmetric function. PointNet++ [52] incorporates a PointNet-based hierarchical structure to learn the neighborhood information of each point, further improving the recognition of local geometric information. Due to its simplicity and effectiveness, subsequent works have used PointNet++ as a backbone, extending it with attention mechanisms [13, 88, 97] or adaptive sampling strategies [32] for better local feature extraction and inference efficiency. Dynamic Graph CNN (DGCNN) [76] builds on these ideas by dynamically updating the graph structure of the point cloud to capture local geometric relationships more effectively, enabling robust and flexible feature learning. CurveNet [46] further enhances this by representing the local geometry of points with learned curve segments, which improves the model's ability to capture fine-grained geometric details. PointConv [81] introduces convolution operations specifically designed for point clouds, enabling efficient and scalable learning of both local and global features. In this work, we leverage these advanced models to evaluate the performance and robustness of our proposed method for generating adversarial point clouds.

2.2 3D Adversarial Attacks and Defences

Unlike image AEs directly adding perturbations, 3D AEs have to modify the shape of point clouds. However, large movements in shifting positions or changing the number of points bring unignorable perturbations to the perceptual quality. Therefore, current works focus on the trade-off between attack success rates and point movement. A few pioneering works extend gradient-based methods to 3D data that were originally developed for images. Liu *et al.* [34] and Yang *et al.* [87] applied the Fast Gradient Sign Method (FGSM) by constraining the movement of each point within a small range in L^2 norm. Further works [29, 35, 42, 77, 82, 95] improved Projected Gradient Descent (PGD) and Carlini and Wagner (C&W) attacks by using either distance constraints or optimization functions to limit the point movement during AE generation. This enhances the smoothness of the adversarial point cloud, making the difference between it and the clean point cloud invisible to human eyes while achieving better attack performance. However, all of these methods require access to model-specific parameter details. Naderi *et al.* [78] proposed a "model-free" approach that does not require knowledge of the target model. Tang *et al.* [66] use generative models for adversarial example generation, enabling the creation of high-quality and metastable adversarial examples without the need for model-specific information. The advantage of these black-box attacks lies in their ability to generate effective adversarial examples without prior knowledge of the target model's architecture or parameters, enhancing their applicability and versatility in various scenarios. The method we proposed is based on a generative diffusion model. Unlike previous generative approaches, we incorporate adversarial noise from the latent space perspective. This enhances the robustness, transferability, and quality of adversarial examples by mitigating outlier generation.

2.3 Diffusion Models

In recent years, diffusion models have garnered significant attention in both academia and industry, demonstrating remarkable results across various applications. The diffusion process considered in this paper is closely related to probabilistic diffusion models, as referenced in [25, 40, 49, 59]. Probabilistic diffusion models are a class of latent variable models that transform noise sampled from a Gaussian distribution into a data distribution using Markov chains. These models operate by iteratively adding and removing noise, effectively learning the underlying data distribution through a series of reversible transformations. While much of the existing work has focused on image-based diffusion models, our approach extends these concepts to the realm of 3D point clouds. In contrast to traditional image diffusion models, our diffusion model is specifically designed to handle 3D point cloud data by conditioning on latent variables to introduce noise into the 3D point cloud and subsequently rethinking point cloud black-box attacks from a latent variable perspective. By leveraging the inherent structure and properties of 3D point clouds, our method aims to enhance the robustness and accuracy of point cloud processing tasks. The application of diffusion models to 3D data presents unique challenges and opportunities, as the high-dimensional and unstructured nature of point clouds necessitates novel techniques for effective noise addition and removal. Our approach incorporates advanced probabilistic methods to manage these complexities, ensuring that the diffusion process preserves the geometric integrity and fine-grained details of the 3D structures. By conditioning on latent variables, our model introduces controlled perturbations to the point clouds, allowing for a nuanced examination of black-box attacks from a latent variable perspective. This enables us to develop more sophisticated and resilient defense mechanisms against such attacks, enhancing the overall robustness and accuracy of point cloud processing tasks. Our work demonstrates the potential of diffusion models to handle high-dimensional, unstructured data, paving the way for future research into leveraging these models for complex data structures and applications.

3 PRELIMINARIES

3.1 Point Cloud

Let $\{X, y\}$ be a point cloud and its corresponding label, where $X = \{x_i\}_{i=1}^{n}, x_i \in \mathcal{R}^3$ refers to n points involved to represent a meaningful shape of a point cloud. Each single point x_i has three dimensions to describe its location in the space. A 3D classifier F learns spatial features of point clouds, which is tasked with correctly predicting the label of each: $F(X) = y$. In this work, we aim to mislead the classifier F to output wrong predictions.

3.2 Diffusion Probabilistic Model

A standard probabilistic diffusion model on point clouds consists of two key processes [25, 40]: 1) a forward diffusion process that incrementally adds noise to the data, transforming it into pure noise, and 2) a reverse diffusion process that iteratively denoises the data to reconstruct the original input (e.g., an image or point cloud) [25, 40]. Assuming that each point x_i in a point cloud is sampled independently from the same underlying distribution (e.g., a point distribution), we model the diffusion and reverse processes for individual points x_i.

The forward process gradually corrupts original data x_i over a fixed number of steps T [25, 40]:

$$q(x_i^{(1:T)}|x_i^{(0)}) = \prod_{t=1}^{T} q(x_i^{(t)}|x_i^{(t-1)}), \tag{1}$$

where $q(x_i^{(t)}|x_i^{(t-1)})$ refers to the noise-adding process at each step t conditioned only on the previous step $t-1$ [25, 40]. Specifically, Gaussian noise is added iteratively as follows:

$$q\left(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}\right) = \mathcal{N}\left(\boldsymbol{x}^{(t)}; \sqrt{1-\beta_t}\boldsymbol{x}^{(t-1)}, \beta_t \boldsymbol{I}\right), t = 1, \ldots, T, \tag{2}$$

where $\beta_t \in (0, 1)$ are hyperparameters controlling the noise level at each step. Larger values of β bring more noise. As t increases, the point cloud x becomes increasingly noisy, eventually approaching pure Gaussian noise at step T.

The reverse process aims to recover meaningful point cloud structures from the noisy data generated by the forward process [40]. These meaningful structures are encoded into a latent representation z which serves as a condition for reconstructing the original sample from step T to 0. The reverse process at each step is defined as

$$p_{\theta}\left(\boldsymbol{x}^{(0:T)} \mid z\right) = p\left(\boldsymbol{x}^{(T)}\right) \prod_{t=1}^{T} p_{\theta}\left(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)}, z\right), \tag{3}$$

$$p_{\theta}\left(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)}, z\right) = \mathcal{N}\left(\boldsymbol{x}^{(t-1)}; \boldsymbol{\mu_{\theta}}\left(\boldsymbol{x}^{(t)}, t, z\right), \beta_t \boldsymbol{I}\right), \tag{4}$$

where μ_θ is an estimated mean through a neural network based on $x^{(t)}$ and the latent representation z of the desired point cloud. Unlike prior diffusion-based methods for point cloud generation [40], which rely on clean point clouds as guidance, our approach aims to generate adversarial examples. To achieve this, we condition the reverse process on latent representations z associated with adversarial classes rather than clean samples. This adaptation allows the reverse diffusion process to generate adversarial point clouds with minimal perceptual noises.

3.3 Problem Analysis

The adversarial perturbation is added by the reverse diffusion process of the diffusion model. Since each reverse diffusion generation of the diffusion model can generate samples similar to the previous step, based on this feature, we assume that the reverse diffusion process [40] can generate an adversarial point cloud close to the previous step's sample point cloud. After iterating this process, we generate the adversarial examples we need. In order to limit the added perturbation δ from being too large, resulting in unexpected deformation of the generated adversarial examples, we use the density-attracted chamfer distance (DCD) to ensure the consistency of the point cloud, which enhances the robustness to local details and is computationally efficient. To further improve the concealment of point clouds, we introduce Mean Square Error(MSE). At the same time, this approach can prevent the original shape of the point cloud from being destroyed due to the generation diversity of the generation model itself. Subsequently, in order to improve the effectiveness of the attack, we need a query update module to improve the adversarial nature of the noise points and ensure that they effectively mislead the recognition model. Based on this idea, we propose an adversarial examples generation scheme based on the diffusion model, and the specific approach will be explained in detail in the next section.

4 METHOD

This section outlines three steps for generating adversarial point clouds, as presented in Figure 2: 1) calculate the latent representations of guidance point clouds, 2) generate adversarial point clouds through a reverse diffusion process, 3) suppress the diversity of the diffusion model. We begin by introducing the threat model considered in this work.

4.1 Threat Model

This work focuses on generating adversarial point clouds under black-box settings to simulate real-world scenarios. The goal is to

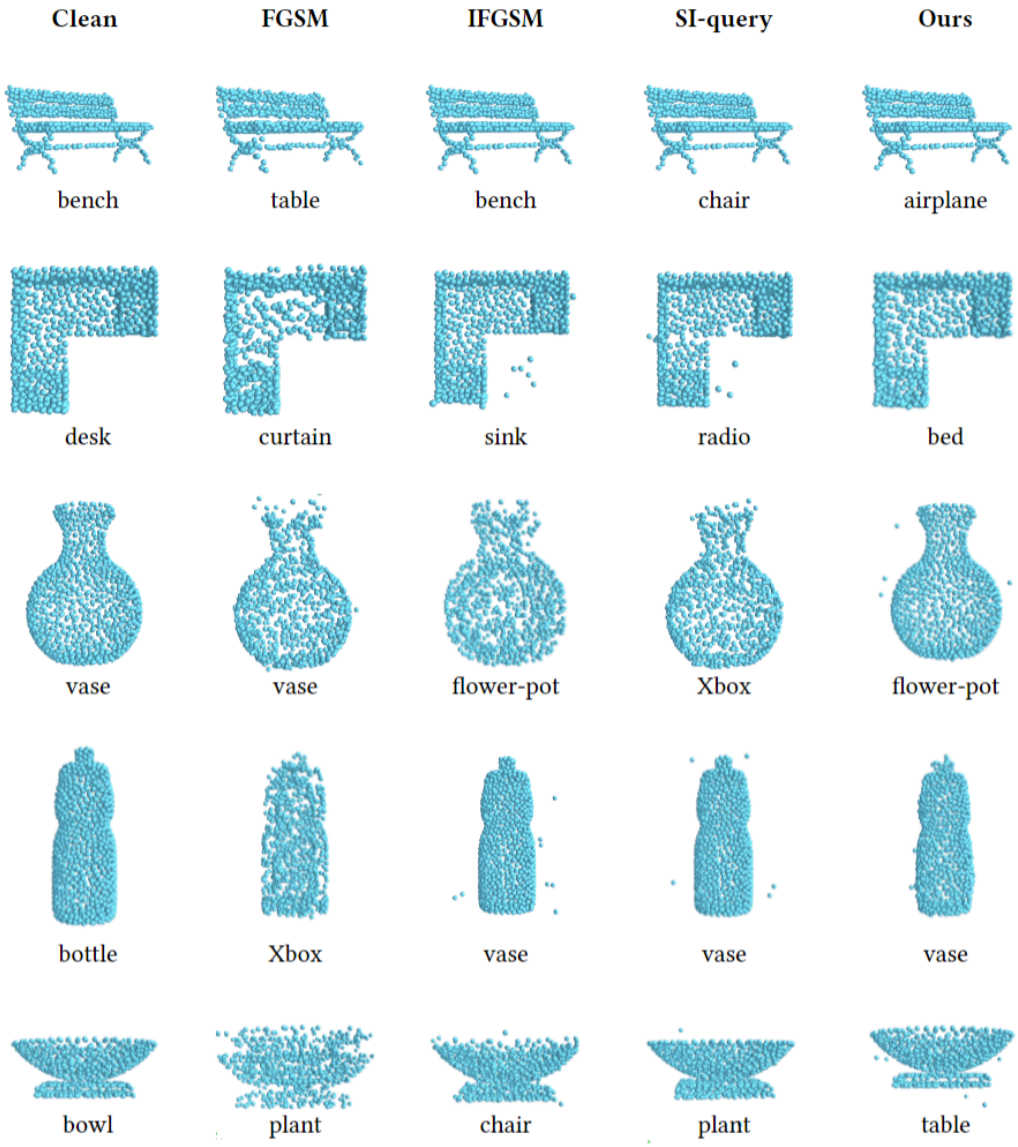

Figure 1: Comparison of various attack methods..

craft effective adversarial point clouds that mislead a target point cloud classifier F by adding imperceptible noise δ:

$$F(X + \delta) \neq y. \tag{5}$$

In this setting, the adversary has limited access to the internal details of the target model F, such as its parameters or outputs. Additionally, the generated adversarial samples are designed to be transferable, enabling them to deceive unseen classifiers as well.

4.2 Calculate latent representation z as guidance

Adversarial point clouds are meaningful structures that closely resemble the original input but include subtle perturbations. In standard diffusion models [40], the meaningful structure of the desired shape is encoded and represented by latent representations z, which are used during the reverse diffusion process for recovery. z is usually learned through a bottleneck layer from a variational autoencoder (VAE). However, unlike standard models, our task focuses on generating adversarial examples. Consequently, the latent representations z are extracted from point clouds belonging to different classes (i.e., classes that do not overlap with the original class of the adversarial examples).

In this step, latent representation refers to the encoded abstract features of input data in a lower-dimensional space, capturing its essential structure for generative tasks. As the diffusion model selects points from the Gaussian distribution, we use a transformer-based model A [72] to fuse the point cloud latent representation with the Gaussian noise to ensure that the added noise points are

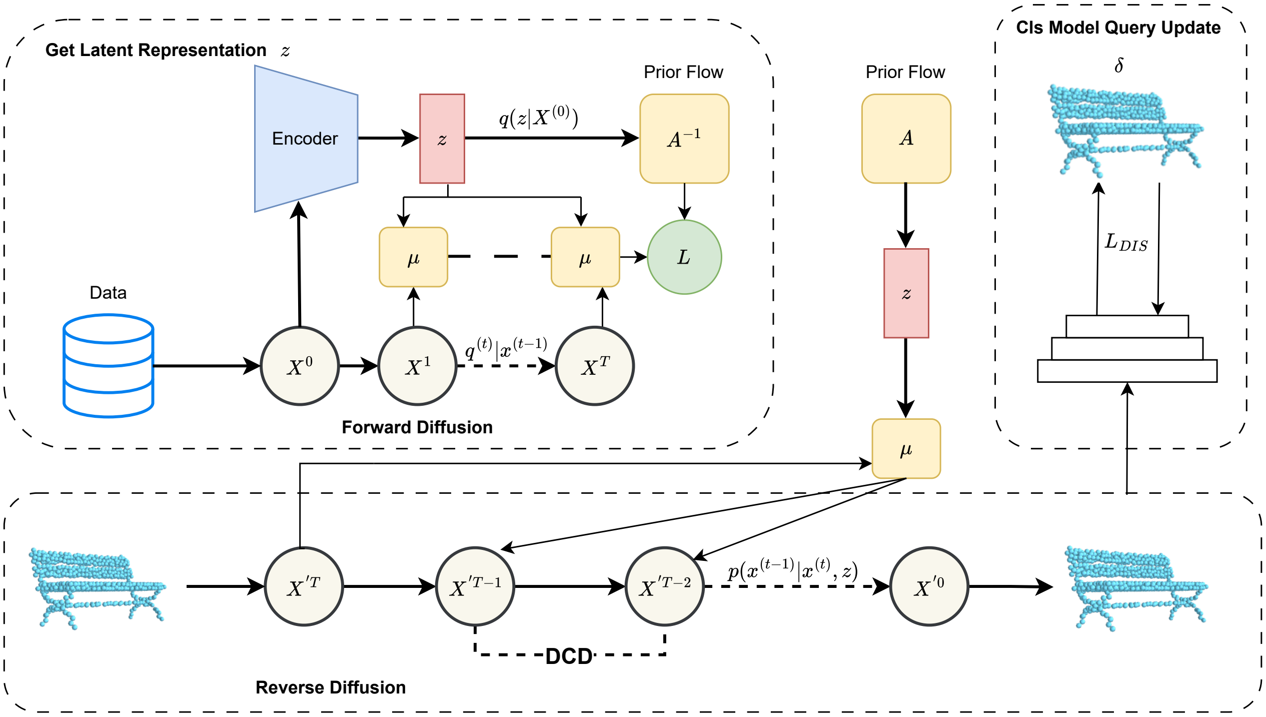

Figure 2: The overview of the proposed method.

the adversarial points we need [40], It can be expressed as:

$$p(z) = p_w\left(A_\alpha^{-1}(z)\right) \cdot \frac{1}{\left|\det J_{A_\alpha}\left(A_\alpha^{-1}(z)\right)\right|} \tag{6}$$

where p_w comes from Gaussian distribution, $F_\alpha^{-1}(z)$ represents the transformation from z through F_α^{-1} back to w, i.e. the transformation between the two. $J_{F_\alpha}(w)$ is the Jacobian matrix of F_α.

z represents latent representation, follows a conditional Gaussian distribution $q(z|x^{(0)})$, where μ serves as the mean of the distribution, encapsulating the global structure of the input point cloud. $q^{(t)}|x^{(t-1)}$ is the forward diffusion step, as described in Section 3.2. We use forward diffusion to learn the latent representation of the point cloud as prior knowledge. $\mathcal{L}$ denotes the loss function [40]. As each reverse diffusion step generates a point cloud close to the previous step, we exploit this feature and use the adversarial latent representation z of the current input to guide the AE generation process.

4.3 Reverse Diffusion for Point Clouds

In this step, we denote the process of transforming a clean point cloud into an adversarial point cloud as the reverse diffusion process. The process can be defined as follows:

$$x^{'} = Attack(x) = reverse(x^T) \cdots reverse(x^0), \tag{7}$$

$$\text{argmin}\, \mathcal{L}_{Attack} = \mathcal{L}_{DIS}(x', x), \tag{8}$$

where x is the input clean point cloud, $Attack(\cdot)$ denotes the reverse diffusion operation, x' is the adversarial examples, and reverse represents the diffusion of each time step t. We use $\mathcal{L}_{DIS}$ to conduct query attacks to improve the attack effectiveness of adversarial points in the reverse diffusion process, which will be introduced in the next step.

4.4 Suppressing the Diversity of Diffusion Model

To prevent the initial point cloud from losing its original shape due to the strong recovery capability of the reverse diffusion process to avoid the generation of excessive outlier points, and improve the concealment of adversarial point clouds we employ a Density-aware Chamfer Distance (DCD) optimization that increases with each reverse diffusion step. We use DCD distance as the loss function to control the diffusion process to ensure consistency between each successive point cloud. We demonstrate the effect of this in the subsequent experiment result section. The Density-aware Chamfer Distance (DCD) is defined as follows:

$$\begin{aligned}\mathcal{L}_{DCD}\left(X, X^{'}\right) = \min \frac{1}{2}\Bigg(\frac{1}{|X|}\sum_{x\in X}\left(1 - \frac{1}{n_{\hat{y}}}e^{-\alpha||x-\hat{y}||_2}\right)\\ +\frac{1}{|X'|}\sum_{y\in X'}\left(1 - \frac{1}{n_{\hat{x}}}e^{-\alpha||y-\hat{x}||_2}\right)\Bigg),\end{aligned} \tag{9}$$

This distance extracts global features in the first stage and introduces local features with rich geometric information in the second stage to realize density perception. The detailed method can be found in [80]. We use the distance as the loss function and minimize this loss in each step of the $reverse(\cdot)$ operation acting on the diffusion process, where $\hat{y} = min_{y \in X'} \|x - y\|_2$, $\hat{x} = min_{y \in X} \|y - x\|_2$, and α denotes a temperature scalar. Here $n_{\hat{y}} = |X_1^y|$, Each y contributes $\left| -\frac{1}{n_y} \sum_{x \in X^y} e^{-\|x-y\|_2} \right| \in [0, 1]$ to the overall distance metric before averaging. This integration ensures that the generated adversarial samples remain imperceptible while effectively deceiving the classification model, thus enhancing the robustness and effectiveness of the diffusion attack scheme.

In addition, we use the Mean Squared Error (MSE) loss function to remove outliers. During the diffusion process, we use DCD instead of MSE because DCD maintains spatial coherence and reduces excessive outliers by focusing on the local structure of the point cloud. For final optimization, MSE is used to ensure global alignment. This balanced approach leverages DCD for local integrity during diffusion and MSE for overall alignment, resulting in high-quality adversarial point clouds with preserved geometric features. We present the results with different loss settings in subsequent ablation experiments. The MSE is defined as follows:

$$\mathcal{L}_{MSE}\,(X, X') = \min(\frac{1}{n} \sum_{i=1}^{n} \|X_i - X_i'\|_2^2). \tag{10}$$

where X' denotes the generated adversarial point cloud. The n stands for the total number of points in the point cloud, and X_i and X_i' are the i-th points in the original and generated point clouds, respectively. The term $\|X_i - X_i'\|_2^2$ represents the squared Euclidean distance between X_i and X_i', quantifying the error for each point pair. We minimize MSE to achieve our goal because we found through experiments that minimizing MSE has little impact on the attack's success rate, but it can improve the concealment of the point cloud. We combine $\mathcal{L}_{DCD}$ with $\mathcal{L}_{MSE}$ to get the final optimization goal, as shown below:

$$\mathcal{L}_{DIS} = \lambda_1 \mathcal{L}_{DCD} + \lambda_2 \mathcal{L}_{MSE} \tag{11}$$

5 EXPERIMENT

In this section, we present the experiment settings, experimental results, and ablation studies. Before elaborating on experimental details, we first set up research questions.

5.1 Research Questions

In this work, we explored generating point cloud AEs with diffusion models. To examine the attack success rate, concealment, and transferability of our method. We evaluate the method with the following research questions (RQs).

- *RQ1:* Given the limited research on generating 3D adversarial examples, can we use diffusion models to generate effective adversarial examples across different datasets and models, Such as PointNet++ and the ModelNet40 dataset?
- *RQ2:* Can the 3D diffusion scheme generate adversarial examples resistant to defense when faced with different models?
- *RQ3:* Explore whether different diffusion steps and noise constraints affect the performance of generated adversarial examples?

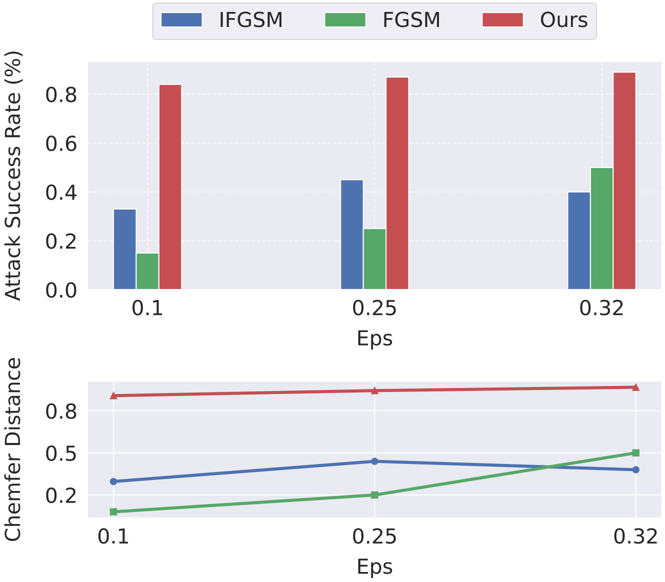

Figure 3: Effect of Eps on attack success rate (ASR) and Chamfer Distance(CD)

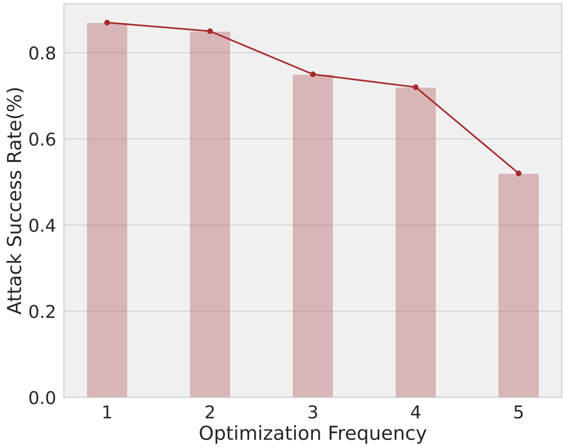

Figure 4: Fix the level of noise added by diffusion at each step and optimize the effect of the number of times on the performance of the attack, targeting PointNet++.

5.2 Experimental Settings

Datasets. For a fair comparison, we evaluate AEs generated by our method on ShapeNet and ModelNet40 as prior works [26]. We assess the attack performance on ShapeNet, which includes approximately 50,000 3D CAD models across 14 major and 55 subcategories, with each model containing at least 2,000 points [5]. Additionally, we evaluate our approach on ModelNet40, consisting of 12,311 CAD models from 40 object categories, with 9,843 for training and 2,468 for testing [63]. Each object in ModelNet40 is uniformly sampled to 2,048 points and rescaled to a unit cube. Data augmentation techniques, such as random scaling and jittering, are applied to preprocess the point clouds in the test set.

RQ1: To test whether our proposed method can generate effective adversarial point clouds on different datasets, we tested it on

Table 1: ASR (%) of different attack methods with and without defense on ModelNet40 (MN).

Proxy Model	data	Defense	Attack Method								
			Drop -400	CW (l_2)	CW (CD)	CW (HD)	GeoA3	AdvPC	LG -GAN	DPMA	Ours
PointNet	MN	-	59.64	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	99.22	93.80	94.20
		SRS	58.14	53.00	67.19	69.94	81.65	**98.87**	92.13	93.48	90.43
		SOR	56.28	13.82	15.63	15.80	42.79	46.19	67.25	87.00	**93.06**
Dgcnn	MN	-	45.91	**100.00**	**100.00**	**100.00**	**100.00**	94.58	86.08	97.45	95.23
		SRS	35.05	31.09	37.11	32.29	77.71	70.63	80.60	**94.73**	92.40
		SOR	15.03	2.26	2.92	3.13	56.25	11.04	50.17	93.11	**94.60**
PointConv	MN	-	37.12	**100.00**	**100.00**	**100.00**	96.09	98.54	78.04	94.98	94.50
		SRS	35.09	37.29	28.95	27.77	21.48	93.54	71.88	**94.06**	92.60
		SOR	34.44	18.13	17.29	19.16	18.35	91.25	63.88	90.71	**93.21**

Table 2: Quantitative comparison between our method and existing black-box transfer-based attacks in terms of attack success rate (ASR), Chamfer distance (CD), Hausdorff distance (HD), and the proxy model used, where CD is multiplied by 10^2 and HD is multiplied by 10^2 for better comparison.

Proxy Model	Attack	PointNet++ [52]			Curvenet [46]			PointConv [81]		
		ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})	ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})	ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})
PointNet [51]	FGSM [58]	68.2	2.93	12.3	75.7	**0.60**	**5.04**	71.9	2.18	17.06
	IFGSM [91]	78.0	2.15	16.7	77.0	1.62	12.7	76.0	**1.06**	11.8
	PGD [73]	70.17	9.8	39.1	67.6	9.83	46.9	61.2	9.7	46.2
Dgcnn [76]	FGSM [58]	41.3	2.52	9.13	50.2	2.52	9.13	55.2	4.09	14.29
	IFGSM [91]	57.6	3.5	12.5	72.5	3.49	12.5	60.2	3.5	12.52
	PGD [73]	61.6	9.63	38.1	62.5	9.73	46.9	62.8	9.8	48.3
Pct [20]	FGSM [58]	30	3.7	14	45.2	3.75	13.9	45.6	3.2	12.4
	IFGSM [91]	44.5	5.5	19.3	72.3	5.56	19.3	64.3	5.56	19.3
	PGD [73]	69	9.9	48.3	63.4	9.9	48	63.0	9.8	48.3
3D-Diffusion	(DifA) Ours	**86.0**	**1.7**	**7.4**	**87**	1.7	7.4	**86**	1.7	**7.4**

both ModelNet40 and ShapeNet datasets. We compare the attack success rate with four different SOTA and other optimization-based methods and conduct experimental comparisons on three different recognition models. The test results of ModelNet40 can be found in Table 1, and the test results of ShapeNet dataset are shown in Table 5.

RQ2: We report the attack access rate with non-defense on ModelNet40 and ShapeNet. While ModelNet40 shows more distinguishable results than ShapeNet, we report the results of the rest tasks on ModelNet40. For ModelNet40, we chose three surrogate models for black-box transfer-based attack evaluation and compared them with our proposed approach. For ShapeNet, we use the proposed method to evaluate the attack performance of two models.

Models. Our approach is evaluated on six benchmark 3D recognition models: PointNet [51], PointNet++(MSG) [52], PointConv [81], Pct [20], DGCNN [76], and Curvenet [46]. These models were chosen for their unique architectures and established performance in 3D recognition tasks:

1. PointNet uses a symmetric function for permutation invariance, while PointNet++ introduces hierarchical feature learning.
2. PointConv enhances feature representation with a point cloud-specific convolution, and Pct excels with a transformer-based architecture.
3. DGCNN captures local structures with dynamic graph construction.
4. Curvenet uses 3D curves for feature extraction.

We selected PointNet, DGCNN, and Pct as proxy models to conduct comparative tests on PointNet++, Curvenet, and PointConv, evaluating our approach's robustness with and without defense measures. This evaluation across diverse architectures allows us to assess the generalizability and effectiveness of our method, providing insights into its performance against various 3D recognition systems.

Baselines. We leverage a pre-trained open-source 3D-diffusion model, trained on extensive point cloud data, as the basis for our manifold attack and compare our approach (Ours) with eight baseline methods. These include the deletion-based method Drop-400 [98], which drops the most critical 400 points, and perturbation-based methods using optimization like C&W under l_2-norm, Chamfer distance (CD), and Hausdorff distance (HD) constraints [82]. Additionally, we consider GeoA3 [77], which applies geometric-aware constraints, and AdvPC [22], which focuses on high transferability. We also compare against generative-based methods such as LG-GAN [100] and DPMA [65]. For attack performance testing, we select the best configuration of these adversarial attack methods to achieve the best attack success rate[65] they can achieve. Due to time constraints, we selected only FGSM, IFGSM, and PGD as comparative experiments for evaluating stealthiness.

Evaluation Metrics. We begin by statistically analyzing the attack success rate (ASR) of the generated adversarial samples on the 3D point cloud recognition model to evaluate the method's effectiveness. To assess the imperceptibility and efficacy of the generated adversarial point clouds, we compute the Hausdorff Distance and Chamfer Distance between the original point cloud and the adversarial output. The Hausdorff Distance quantifies the maximum deviation between point sets, while the Chamfer Distance measures the average displacement, providing insights into the subtlety of the perturbations. The formula is defined as follows.

$$\text{ASR} = \frac{\sum_{i=1}^{N} \mathbb{I}(Y_i \neq \hat{Y}_i)}{N} \times 100\%, \tag{12}$$

$$\mathcal{D}_C \left(X, X'\right) = \frac{1}{\|X'\|_0} \sum_{y \in X'} \min_{x \in X} \|x - y\|_2^2 \tag{13}$$

$$\mathcal{D}_H \left(X, X'\right) = \max_{y \in X'} \min_{x \in X} \|x - y\|_2^2, \tag{14}$$

In these formulas, X represents the original point cloud, and X' denotes the adversarial point cloud. Equation 12 defines the Attack Success Rate (ASR), which quantifies the percentage of perturbed samples misclassified by the model. Y_i and $\hat{Y}_i$ represent the true and predicted labels, respectively, and N is the total number of samples. Equation 13 defines the Chamfer Distance ($\mathcal{D}_C$), which calculates the average displacement between points in X and X'. Equation 14 defines the Hausdorff Distance ($\mathcal{D}_H$), which measures the maximum deviation between points in the two sets. The ASR, defined in Equation 12, quantifies the ratio of perturbed examples incorrectly classified by the black-box transfer-based attacked model. Evaluating our attack in terms of effectiveness, stealthiness, and transferability offers a comprehensive understanding of its impact on 3D recognition models.

- *Effectiveness*: Measured by the attack success rate (ASR), which reflects the percentage of adversarial examples that successfully mislead the target model. A higher ASR indicates a more effective attack.
- *Stealthiness*: Assessed using the Hausdorff distance and Chamfer distance to ensure that the generated adversarial examples are imperceptible and maintain a high degree of similarity to the original point clouds.
- *Transferability*: Evaluated by testing the adversarial examples on different models to determine the robustness and generalization of the attack across various 3D recognition systems.

RQ3: We use ablation experiments to find out the impact of different diffusion steps and noise constraints on the attack performance of generated adversarial examples. Details can be seen in Ablation Studies.

5.3 Experimental Results - RQ1

Performance Comparison with White Box Attacks and Generative Attacks. The result in Table 1 shows that Drop-400 performs the worst. Among them, the three CW attacks (using l2, Chamfer Distance, and Hausdorff Distance as loss functions), GeoA3, and AdvPC all showed a 100% success rate on PointNet. In particular, LG-GAN [100], DPMA [65] and our method also successfully attacked these models, but the success rate was slightly lower than the previous methods. This is because the previous methods are white-box attack methods, which can use the gradient information of the attacked model, which greatly improves the success rate of the attack. However, our proposed method differs from LGAN and DPMA in that our method does not use any information from the attacked model, and our method is dataset-oriented rather than sample-oriented. In addition, the attack success rate of AdvPC on Dgcnn and PointConv did not reach 100%, due to the trade-off imposed by auto-encoder for transferability.

Black-box Performance And Comparison. We comprehensively compare our 3D diffusion black-box attack with various baselines, including regular optimization-based attacks such as FGSM, IFGSM, and PGD. Specifically, our method is implemented with a reverse diffusion step size of 100 and $1t$ DCD optimization iterations, while all baselines adopt untargeted attacks with ϵ set to 0.32. The comparisons are conducted on the same RTX 3050 GPU, evaluating metrics such as attack success rate (ASR), Chamfer distance (CD), and Hausdorff distance (HD). The results listed in Table 2 show that our method incurs the least geometric distance cost to achieve nearly 90% ASR, with a lower time budget compared to regular optimization-based attacks. This aligns with our intuition that the diffusion model better preserves the point features from the original point clouds during adversarial example generation.

To evaluate the generalizability of these attack methods, we assessed the performance of adversarial point clouds after careful data preprocessing and measured the attack's success rate (ASR) across different models. The results demonstrate that our proposed diffusion attack consistently outperforms other methods in most cases in terms of ASR, CD, and HD, indicating its robustness and effectiveness across various 3D recognition models.

5.4 Results on Defense Approaches - RQ2

To further validate the robustness of each attack method, we perform performance evaluation tests on the produced adversarial examples after the defense methods. We consider common point cloud input preprocessing defense methods [65]. We demonstrate the superiority of our proposed scheme by testing it against other attack methods (FGSM, IFGSM, PGD). Input defense methods We choose two common input preprocessing schemes (SOR, SRS) for point clouds, The results are shown in Tables 3 and 4. Moreover, the attack effect of our proposed method after SOR defense is almost the highest success rate among all attacks, and it also performs best in Chamfer Distance and Hausdorff Distance.

Performance comparison under SOR defense approach We use the SOR defense method with k set to 2 and α using 1.1 as the defense parameter, and the results obtained show that our proposed scheme still has good aggressiveness after the point cloud defense treatment. In our experiments, we found that if the proxy model is Pointnet, the success rate in attacking PointNet++, Curvenet, and PointConv models is higher than if the proxy model is the other two. Using Dgcnn as the proxy model, the attack performance for Curvenet, PointConv is higher than that for PointNet++. This is an interesting phenomenon, suggesting that point cloud attacks can be ideally achieved by using similar models to generate adversarial examples for robustness tests. And the experimental results show that the attack method based on the diffusion generation model we proposed can produce better attack performance on different models through one-time proxy model generation, and the Chamfer Distance and Hausdorff Distance are smaller than other attack methods, indicating the superiority of our method. As shown in the figure, we show the visualization of adversarial point clouds of our proposed method and Shape-Invariant(SI) for comparison in Table **??**. The SI method is set to $\epsilon = 0.16$ and step size = 0.007, The noise scale of our method is set to $0.05t$. Our adversarial generation scheme based on the diffusion model has fewer outliers and a

Table 3: Quantitative comparison between our method and existing black-box transfer-based attacks in terms of attack success rate (ASR), Chamfer distance (CD), Hausdorff distance (HD), the proxy model and defense method SOR used, where CD is multiplied by 10^2 and HD is multiplied by 10^2 for better comparison.

Proxy Model	Attack	Defense	PointNet++ [52]			Curvenet [46]			PointConv [81]		
			ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})	ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})	ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})
PointNet	FGSM [58]	SOR	54.3	**1.2**	16.3	66.5	1.6	**2.05**	54.3	1.2	16.3
	IFGSM [91]		75.3	4.8	38.5	80	4.3	30.5	72	**1.02**	12.2
	PGD [73]		69.5	9.9	48.2	82	9.95	48.2	80	9.7	46.4
Dgcnn	FGSM [58]	SOR	45.05	2.72	11.68	61	2.7	11.6	64.8	3.84	14.5
	IFGSM [91]		48.5	5.07	18.4	73.2	3.84	14.5	73	3.84	14.4
	PGD [73]		69.5	9.9	48.2	83	9.75	48.1	82	9.8	47.6
Pct	FGSM [58]	SOR	46.8	2.75	11.68	61.8	2.7	11.6	54.9	2.7	11.6
	IFGSM [91]		47.8	5.09	18.5	77.8	5.09	18.5	65.6	5.06	18.4
	PGD [73]		72.7	9.8	48.3	72.5	9.8	48.3	81	9.8	48.3
3D-Diffusion	(DifA)Ours	SOR	**84.7**	1.4	**6.8**	**86.2**	**1.4**	6.8	**85.0**	1.4	**6.8**

Table 4: Quantitative comparison between our method and existing black-box transfer-based attacks in terms of attack success rate (ASR), Chamfer distance (CD), Hausdorff distance (HD), the proxy model and defense method SRS used, where CD is multiplied by 10^2 and HD is multiplied by 10^2 for better comparison.

Proxy Model	Attack	Defense	PointNet++ [52]			Curvenet [46]			PointConv [80]		
			ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})	ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})	ASR↑ (%)	CD↓ (10^{-2})	HD↓ (10^{-2})
PointNet	FGSM [58]	SRS	68.9	9.89	4.83	66.5	3.6	20.5	67.6	2.29	25.4
	IFGSM [91]		75.3	4.8	38.5	70	4.8	23.5	80.1	1.6	16.1
	PGD [73]		68.9	9.89	4.83	65.2	9.8	4.83	64.6	9.89	**4.8**
Dgcnn	FGSM [58]	SRS	23.4	4.0	14.0	41.1	4.0	14.0	51.8	4.0	14.0
	IFGSM [91]		34.7	5.7	19.2	70	5.7	19.2	68.6	5.7	19.2
	PGD [73]		68.6	9.89	48.3	65.5	9.89	48.3	64.8	9.8	48.1
pct	FGSM [58]	SRS	22	3.9	13.9	42.3	3.9	13.9	48.7	3.9	13.9
	IFGSM [91]		33.1	5.8	19.27	68.8	5.8	19.2	68.0	5.8	19.2
	PGD [73]		69.5	9.8	48.3	65.5	9.7	47.2	64.9	9.7	47.2
3D-Diffusion	(DifA)Ours	SRS	**85.0**	**1.5**	**6.7**	**82.3**	**1.5**	**6.7**	**81.2**	**1.5**	6.7

smoother surface than the SI attack. Regardless of whether it has undergone the point cloud defense method, the point cloud quality generated by our adversarial sample generation scheme is superior to that of the SI attack.

Performance comparison under SRS defense approach: Although our proposed method shows good performance against SOR defense methods, it may be insufficient against some well-designed adversarial examples, for this reason, we conducted SRS defense experiments to improve the persuasiveness of our method. For SRS defense method, we set the drop num points to 500, experimental results show that our proposed method has better attack performance and transferability in the face of both defense methods, and the distance shows that our attack does not significantly damage the original point cloud. Moreover, after the adversarial point cloud generated by other attack methods is passed through the SRS defense method, the attack success rate drops significantly. It can be seen that randomly discarding 500 points is effective in reducing adversarial losses. Overall our 3D diffusion attack has strong resistance and good transferability against these defenses.

5.5 Ablation Studies - RQ3

The amount of noise added by the attack is a critical and configurable parameter that affects the performance of our method. We evaluated different numbers of DCD optimizations with a fixed number of diffusion model steps, and the results are shown in Fig 4. Our method is directly affected by the number of optimizations, and the performance of the attack progressively decreases as the number of optimizations decreases from $5t$ to $1t$.

Table 5: Attack success rate (ASR) and Chamfer Distance of our method with PointNet++ and Curvenet on ShapeNet.

Attack		PointNet++ [52]		Curvenet [46]	
		ASR↑ (%)	CD↓ (10^{-2})	ASR↑ (%)	CD↓ (10^{-2})
Ours	Chair	70	0.8	64	0.8
	Airplane	65	0.3	75	0.3
	Bench	82	1	93	1

Table 6: Different loss function settings are used in the diffusion process and the impact of the order on the results.

Loss		PointNet++ [52]		
		ASR↑	CD↓	HD↑
Ours	CDC	73	0.13	0.71
	MSE+CDC	70	0.06	0.46
	CDC+MSE	71	0.06	0.23

In order to find the appropriate number of optimizations, we combine the success rate of the attack and the various metrics of the resistance to defense and the stealthiness of the generated adversarial examples, and use the 2t optimization as our experimental benchmark.

To investigate the impact of different loss functions during the diffusion process on the generation of adversarial samples, we conducted ablation experiments focusing on the sequence and combination of these loss functions. As shown in Table 6, the combination of DCD and MSE demonstrates a strong constraint on the noise added during the diffusion process while minimally affecting the attack success rate. These results suggest that the DCD+MSE combination effectively balances the imperceptibility of the perturbations and the overall attack performance. Due to time constraints, we only selected the Chair classification test set data for evaluation.

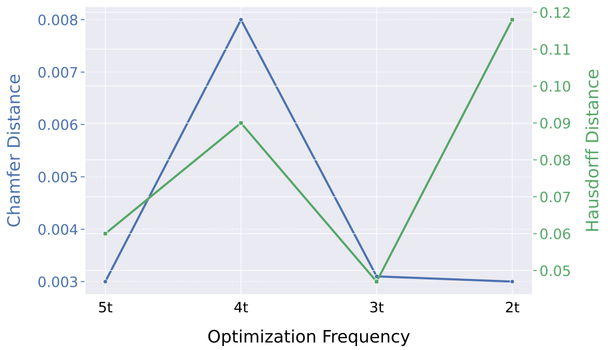

Figure 5: Comparison experiments of Chamfer Distance and Hausdorff Distance under different settings of the number of DCD optimizations for diffusion attacks, with the attack target network as PointNet++.

6 CONCLUSION

The deceptive and imperceptible point cloud was successfully generated using 3D diffusion modeling to attack the target autopilot system. This attack proves to be effective, causing the system to misjudge in the perception phase, potentially leading to dangerous behaviors. The attack point cloud is imperceptible. The generated point cloud demonstrates robustness and adaptability across various scenes, lighting conditions, and vehicle states, effectively compromising the target system's perception and decision-making modules. Our proposed black-box adversarial sample generation method is capable of producing deceptive adversarial examples (AEs) with a certain degree of transferability. However, the method incurs significant time costs during execution. While generating black-box adversarial examples using generative models seems feasible, there remains room for improvement in reducing deformation compared to more advanced white-box approaches. In future work, we will continue to explore methods for generating black-box adversarial examples with less deformation from the perspective of the diffusion model encoder.

REFERENCES

[1] Atrin Arya, Hanieh Naderi, and Shohreh Kasaei. 2023. Adversarial attack by limited point cloud surface modifications. In *2023 6th International Conference on Pattern Recognition and Image Analysis (IPRIA)*. IEEE, 1–8.

[2] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. 2021. Invisible for both camera and lidar: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *2021 IEEE symposium on security and privacy (SP)*. IEEE, 176–194.

[3] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. 2021. Invisible for both camera and lidar: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *2021 IEEE symposium on security and privacy (SP)*. IEEE, 176–194.

[4] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*. Ieee, 39–57.

[5] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012* (2015).

[6] Jianqi Chen, Hao Chen, Keyan Chen, Yilan Zhang, Zhengxia Zou, and Zhenwei Shi. 2023. Diffusion models for imperceptible and transferable adversarial attack. *arXiv preprint arXiv:2305.08192* (2023).

[7] Xi Chen, Diederik P Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. Variational lossy autoencoder. *arXiv preprint arXiv:1611.02731* (2016).

[8] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. 2017. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 1907–1915.

[9] Zhiyu Chen, Feng Chen, Yiming Sun, Mingjie Wang, Shangdong Liu, and Yimu Ji. 2024. Local aggressive and physically realizable adversarial attacks on 3D point cloud. *Computers & Security* 139 (2024), 103539.

[10] Wenda Chu, Linyi Li, and Bo Li. 2022. Tpc: Transformation-specific smoothing for point cloud models. In *International Conference on Machine Learning*. PMLR, 4035–4056.

[11] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9185–9193.

[12] Yinpeng Dong, Jun Zhu, Xiao-Shan Gao, et al. 2022. Isometric 3d adversarial examples in the physical world. *Advances in Neural Information Processing Systems* 35 (2022), 19716–19731.

[13] Yueqi Duan, Yu Zheng, Jiwen Lu, Jie Zhou, and Qi Tian. 2019. Structural relational reasoning of point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 949–958.

[14] Mahmood H Enad, Omar I Dallal Bashi, Shymaa Mohammed Jameel, Asaad A Alhasoon, Yasir Mahmood Al Kubaisi, and Husamuldeen K Hameed. 2024. Detecting and tracking a road-drivable area with three-dimensional point clouds and IoT for autonomous applications. *Service Oriented Computing and Applications* (2024), 1–11.

[15] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. 2018. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1625–1634.

[16] Zhongbin Fang, Xiangtai Li, Xia Li, Joachim M Buhmann, Chen Change Loy, and Mengyuan Liu. 2024. Explore in-context learning for 3d point cloud understanding. *Advances in Neural Information Processing Systems* 36 (2024).

[17] Aleksey Golovinskiy, Vladimir G Kim, and Thomas Funkhouser. 2009. Shape-based recognition of 3D point clouds in urban environments. In *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2154–2161.

[18] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

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

[20] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. 2021. Pct: Point cloud transformer. *Computational Visual Media* 7 (2021), 187–199.

[21] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. 2020. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence* 43, 12 (2020), 4338–4364.

[22] Abdullah Hamdi, Sara Rojas, Ali Thabet, and Bernard Ghanem. 2020. Advpc: Transferable adversarial perturbations on 3d point clouds. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*. Springer, 241–257.

[23] Abdullah Hamdi, Sara Rojas, Ali K. Thabet, and Bernard Ghanem. 2019. AdvPC: Transferable Adversarial Perturbations on 3D Point Clouds. *ArXiv* abs/1912.00461 (2019). https://api.semanticscholar.org/CorpusID:208527476

[24] Bangyan He, Jian Liu, Yiming Li, Siyuan Liang, Jingzhi Li, Xiaojun Jia, and Xiaochun Cao. 2023. Generating transferable 3d adversarial point cloud via random perturbation factorization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 764–772.

[25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

[26] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Hang Zhou, Weiming Zhang, and Nenghai Yu. 2022. Shape-invariant 3d adversarial point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 15335–15344.

[27] Tianxin Huang, Qingyao Liu, Xiangrui Zhao, Jun Chen, and Yong Liu. 2024. Learnable Chamfer Distance for point cloud reconstruction. *Pattern Recognition Letters* 178 (2024), 43–48.

[28] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34.

8018–8025.

[29] Jaeyeon Kim, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. 2021. Minimal adversarial examples for deep learning on 3d point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7797–7806.

[30] Kibok Lee, Zhuoyuan Chen, Xinchen Yan, Raquel Urtasun, and Ersin Yumer. 2020. Shapeadv: Generating shape-aware adversarial 3d point clouds. *arXiv preprint arXiv:2005.11626* (2020).

[31] Peizheng Li, Jagdeep Singh, Han Cui, and Carlo Alberto Boano. 2024. BmmW: A DNN-based joint BLE and mmWave radar system for accurate 3D localization with goal-oriented communication. *Pervasive and Mobile Computing* (2024), 101944.

[32] Hongxin Lin, Zelin Xiao, Yang Tan, Hongyang Chao, and Shengyong Ding. 2019. Justlookup: One millisecond deep feature extraction for point clouds by lookup tables. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 326–331.

[33] Daizong Liu and Wei Hu. 2022. Imperceptible transfer attack and defense on 3d point cloud classification. *IEEE transactions on pattern analysis and machine intelligence* 45, 4 (2022), 4727–4746.

[34] Daniel Liu, Ronald Yu, and Hao Su. 2019. Extending adversarial attacks and defenses to deep 3d point cloud classifiers. In *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2279–2283.

[35] Daniel Liu, Ronald Yu, and Hao Su. 2020. Adversarial shape perturbations on 3d point clouds. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. Springer, 88–104.

[36] Ye Liu, Yaya Cheng, Lianli Gao, Xianglong Liu, Qilong Zhang, and Jingkuan Song. 2022. Practical evaluation of adversarial robustness via adaptive auto attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15105–15114.

[37] Yongcheng Liu, Bin Fan, Gaofeng Meng, Jiwen Lu, Shiming Xiang, and Chunhong Pan. 2019. Densepoint: Learning densely contextual representation for efficient point cloud processing. In *Proceedings of the IEEE/CVF international conference on computer vision*. 5239–5248.

[38] Tianrui Lou, Xiaojun Jia, Jindong Gu, Li Liu, Siyuan Liang, Bangyan He, and Xiaochun Cao. 2024. Hide in thicket: Generating imperceptible and rational adversarial perturbations on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 24326–24335.

[39] Shitong Luo and Wei Hu. 2021. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2837–2845.

[40] Shitong Luo and Wei Hu. 2021. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2837–2845.

[41] Zhaoyang Lyu, Jinyi Wang, Yuwei An, Ya Zhang, Dahua Lin, and Bo Dai. 2023. Controllable Mesh Generation Through Sparse Latent Point Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 271–280.

[42] Chengcheng Ma, Weiliang Meng, Baoyuan Wu, Shibiao Xu, and Xiaopeng Zhang. 2020. Efficient joint gradient based attack against sor defense for 3d point cloud classification. In *Proceedings of the 28th ACM International Conference on Multimedia*. 1819–1827.

[43] Irfan Manisali, Okyanus Oral, and Figen S Oktem. 2024. Efficient physics-based learned reconstruction methods for real-time 3D near-field MIMO radar imaging. *Digital Signal Processing* 144 (2024), 104274.

[44] Luke Melas-Kyriazi, Christian Rupprecht, and Andrea Vedaldi. 2023. Pc2: Projection-conditioned point cloud diffusion for single-image 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12923–12932.

[45] Shayan Mesdaghi, Reza PR Hasanzadeh, and Farrokh Janabi-Sharifi. 2024. Finger-hand Rehabilitation using DNN-based Gesture Recognition of Low-cost Webcam Images. In *2024 13th Iranian/3rd International Machine Vision and Image Processing Conference (MVIP)*. IEEE, 1–6.

[46] AAM Muzahid, Wanggen Wan, Ferdous Sohel, Lianyao Wu, and Li Hou. 2020. CurveNet: Curvature-based multitask learning deep networks for 3D object recognition. *IEEE/CAA Journal of Automatica Sinica* 8, 6 (2020), 1177–1187.

[47] Hanieh Naderi, Chinthaka Dinesh, Ivan V Bajic, and Shohreh Kasaei. 2022. Model-free prediction of adversarial drop points in 3D point clouds. *arXiv preprint arXiv:2210.14164* (2022).

[48] Trung Nguyen, Quang-Hieu Pham, Tam Le, Tung Pham, Nhat Ho, and Binh-Son Hua. 2021. Point-set distances for learning representations of 3d point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*. 10478–10487.

[49] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. 2022. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751* (2022).

[50] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. 2022. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460* (2022).

[51] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 652–660.

[52] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems* 30 (2017).

[53] Shyam Nandan Rai, Fabio Cermelli, Dario Fontanel, Carlo Masone, and Barbara Caputo. 2023. Unmasking anomalies in road-scene segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4037–4046.

[54] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* 1, 2 (2022), 3.

[55] Satti RG Reddy, GP Saradhi Varma, and Rajya Lakshmi Davuluri. 2024. Deep neural network (dnn) mechanism for identification of diseased and healthy plant leaf images using computer vision. *Annals of Data Science* 11, 1 (2024), 243–272.

[56] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10684–10695.

[57] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems* 35 (2022), 36479–36494.

[58] Jaydip Sen and Subhasis Dasgupta. 2023. Adversarial attacks on Image classification models: FGSM and patch attacks and their impact. *arXiv preprint arXiv:2307.02055* (2023).

[59] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, 2256–2265.

[60] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, 2256–2265.

[61] Jiachen Sun, Yulong Cao, Qi Alfred Chen, and Z Morley Mao. 2020. Towards robust {LiDAR-based} perception in autonomous driving: General black-box adversarial sensor attack and countermeasures. In *29th USENIX Security Symposium (USENIX Security 20)*. 877–894.

[62] Jiachen Sun, Karl Koenig, Yulong Cao, Qi Alfred Chen, and Z Morley Mao. 2020. On adversarial robustness of 3d point cloud classification under adaptive attacks. *arXiv preprint arXiv:2011.11922* (2020).

[63] Jiachen Sun, Qingzhao Zhang, Bhavya Kailkhura, Zhiding Yu, Chaowei Xiao, and Z Morley Mao. 2022. Modelnet40-c: A robustness benchmark for 3d point cloud recognition under corruption. In *ICLR 2022 Workshop on Socially Responsible Machine Learning*, Vol. 7.

[64] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* (2013).

[65] Keke Tang, Jianpeng Wu, Weilong Peng, Yawen Shi, Peng Song, Zhaoquan Gu, Zhihong Tian, and Wenping Wang. 2023. Deep manifold attack on point clouds via parameter plane stretching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 2420–2428.

[66] Keke Tang, Jianpeng Wu, Weilong Peng, Yawen Shi, Peng Song, Zhaoquan Gu, Zhihong Tian, and Wenping Wang. 2023. Deep Manifold Attack on Point Clouds via Parameter Plane Stretching. *Proceedings of the AAAI Conference on Artificial Intelligence* 37, 2 (Jun. 2023), 2420–2428. https://doi.org/10.1609/aaai.v37i2.25338

[67] Gusi Te, Wei Hu, Amin Zheng, and Zongming Guo. 2018. Rgcnn: Regularized graph cnn for point cloud segmentation. In *Proceedings of the 26th ACM international conference on Multimedia*. 746–754.

[68] Tzungyu Tsai, Kaichen Yang, Tsung-Yi Ho, and Yier Jin. 2020. Robust adversarial objects against deep learning models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 954–962.

[69] James Tu, Mengye Ren, Sivabalan Manivasagam, Ming Liang, Bin Yang, Richard Du, Frank Cheng, and Raquel Urtasun. 2020. Physically realizable adversarial examples for lidar object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 13716–13725.

[70] James Tu, Mengye Ren, Sivabalan Manivasagam, Ming Liang, Bin Yang, Richard Du, Frank Cheng, and Raquel Urtasun. 2020. Physically Realizable Adversarial Examples for LiDAR Object Detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13713–13722. https://doi.org/10.1109/CVPR42600.2020.01373

[71] Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. 2022. Lion: Latent point diffusion models for 3d shape generation. *Advances in Neural Information Processing Systems* 35 (2022), 10021–10039.

[72] A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).

[73] William Villegas-Ch, Angel Jaramillo-Alcázar, and Sergio Luján-Mora. 2024. Evaluating the Robustness of Deep Learning Models against Adversarial Attacks: An Analysis with FGSM, PGD and CW. *Big Data and Cognitive Computing* 8, 1 (2024), 8.

[74] Haiyan Wang and Yingli Tian. 2024. Sequential point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[75] Min Wang, Jiehui Jiang, Zhuangzhi Yan, Ian Alberts, Jingjie Ge, Huiwei Zhang, Chuantao Zuo, Jintai Yu, Axel Rominger, Kuangyu Shi, et al. 2020. Individual brain metabolic connectome indicator based on Kullback-Leibler Divergence Similarity Estimation predicts progression from mild cognitive impairment to Alzheimer's dementia. *European journal of nuclear medicine and molecular imaging* 47 (2020), 2753–2764.

[76] Yue Wang and Justin M Solomon. 2021. Object dgcnn: 3d object detection using dynamic graphs. *Advances in Neural Information Processing Systems* 34 (2021), 20745–20758.

[77] Yuxin Wen, Jiehong Lin, Ke Chen, CL Philip Chen, and Kui Jia. 2020. Geometry-aware generation of adversarial point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 6 (2020), 2984–2999.

[78] Matthew Wicker and Marta Kwiatkowska. 2019. Robustness of 3d deep learning in an adversarial setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 11767–11775.

[79] Kevin Tirta Wijaya, Dong-Hee Paek, and Seung-Hyun Kong. 2024. Advanced feature learning on point clouds using multi-resolution features and learnable pooling. *Remote Sensing* 16, 11 (2024), 1835.

[80] Tong Wu, Liang Pan, Junzhe Zhang, Tai Wang, Ziwei Liu, and Dahua Lin. 2021. Density-aware chamfer distance as a comprehensive metric for point cloud completion. *arXiv preprint arXiv:2111.12702* (2021).

[81] Wenxuan Wu, Zhongang Qi, and Li Fuxin. 2019. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition.* 9621–9630.

[82] Chong Xiang, Charles R Qi, and Bo Li. 2019. Generating 3d adversarial point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 9136–9144.

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

[84] Songsong Xiong, Georgios Tziafas, and Hamidreza Kasaei. 2023. Enhancing fine-grained 3D object recognition using hybrid multi-modal vision transformer-CNN models. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE, 5751–5757.

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

[86] Mutian Xu, Runyu Ding, Hengshuang Zhao, and Xiaojuan Qi. 2021. PAConv: Position Adaptive Convolution with Dynamic Kernel Assembling on Point Clouds. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 3172–3181. https://doi.org/10.1109/CVPR46437.2021.00319

[87] Jiancheng Yang, Qiang Zhang, Rongyao Fang, Bingbing Ni, Jinxian Liu, and Qi Tian. 2019. Adversarial attack and defense on point sets. *arXiv preprint arXiv:1902.10899* (2019).

[88] Jiancheng Yang, Qiang Zhang, Bingbing Ni, Linguo Li, Jinxian Liu, Mengdie Zhou, and Qi Tian. 2019. Modeling point clouds with self-attention and gumbel subset sampling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 3323–3332.

[89] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. 2018. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 206–215.

[90] Hakan Yekta Yatbaz, Mehrdad Dianati, Konstantinos Koufos, and Roger Woodman. 2024. Run-time Monitoring of 3D Object Detection in Automated Driving Systems Using Early Layer Neural Activation Patterns. *arXiv preprint arXiv:2404.07685* (2024).

[91] Haotian You, Yufang Lu, and Haihua Tang. 2023. Plant disease classification and adversarial attack using SimAM-EfficientNet and GP-MI-FGSM. *Sustainability* 15, 2 (2023), 1233.

[92] Xuejing Yuan, Yuxuan Chen, Yue Zhao, Yunhui Long, Xiaokang Liu, Kai Chen, Shengzhi Zhang, Heqing Huang, Xiaofeng Wang, and Carl A Gunter. 2018. {CommanderSong}: A systematic approach for practical adversarial voice recognition. In *27th USENIX security symposium (USENIX security 18).* 49–64.

[93] Xiangyu Yue, Bichen Wu, Sanjit A Seshia, Kurt Keutzer, and Alberto L Sangiovanni-Vincentelli. 2018. A lidar point cloud generator: from a virtual world to autonomous driving. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval.* 458–464.

[94] Dejun Zhang, Fazhi He, Soonhung Han, Lu Zou, Yiqi Wu, and Yilin Chen. 2017. An efficient approach to directly compute the exact Hausdorff distance for 3D point sets. *Integrated Computer-Aided Engineering* 24, 3 (2017), 261–277.

[95] Jinlai Zhang, Lyujie Chen, Binbin Liu, Bo Ouyang, Qizhi Xie, Jihong Zhu, Weiming Li, and Yanmei Meng. 2023. 3d adversarial attacks beyond point cloud. *Information Sciences* 633 (2023), 491–503.

[96] Jianping Zhang, Wenwei Gu, Yizhan Huang, Zhihan Jiang, Weibin Wu, and Michael R Lyu. 2024. Curvature-Invariant Adversarial Attacks for 3D Point Clouds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 7142–7150.

[97] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. 2019. Pointweb: Enhancing local neighborhood features for point cloud processing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 5565–5573.

[98] Tianhang Zheng, Changyou Chen, Junsong Yuan, Bo Li, and Kui Ren. 2019. Pointcloud saliency maps. In *Proceedings of the IEEE/CVF international conference on computer vision.* 1598–1606.

[99] Boxuan Zhong, He Huang, and Edgar Lobaton. 2020. Reliable vision-based grasping target recognition for upper limb prostheses. *IEEE Transactions on Cybernetics* 52, 3 (2020), 1750–1762.

[100] Hang Zhou, Dongdong Chen, Jing Liao, Kejiang Chen, Xiaoyi Dong, Kunlin Liu, Weiming Zhang, Gang Hua, and Nenghai Yu. 2020. Lg-gan: Label guided adversarial network for flexible targeted attack of point cloud based deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 10356–10365.

[101] Yuan Zhuang, Qipeng Li, Yiwen Chen, Jianzhu Huai, Miao Li, Tianbing Ma, Yufei Tang, and Xinlian Liang. 2024. 3D-SeqMOS: A Novel Sequential 3D Moving Object Segmentation in Autonomous Driving. *IEEE Transactions on Intelligent Transportation Systems* (2024).