ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume X-1/W2-2025
13th International Conference on Mobile Mapping Technology (MMT 2025)
"Mobile Mapping for Autonomous Systems and Spatial Intelligence", 20–22 June 2025, Xiamen, China

Generating Transferable Traffic Object Adversarial 3D Point Clouds via Momentum-based Decompose Perturbation

Weiquan Liu, Min Xie, Xingwang Huang *, Jiasheng Su, Yanwen Sun, Shiwei Lin, Jinhe Su, Zongyue Wang, Guorong Cai

College of Computer Engineering, Jimei University, China

Keywords: 3D point cloud, Intelligent driving, Adversarial attack, Adversarial transferability

Abstract

With the rapid development of mobile mapping technology, 3D point cloud data is widely used in the field of intelligent driving. In intelligent driving systems, the recognition ability of point cloud objects is crucial for achieving safe driving. However, existing deep neural networks are prone to making incorrect judgments when subjected to adversarial attacks, which may lead to serious consequences. Most of the existing point cloud perturbation methods are based on white box attacks and cannot successfully attack models with unknown parameters, which is still different from real usage scenarios. In this paper, we focus on studying the transferability of point cloud perturbation, that is, successful attacks on a model can also be transferred to models that have not participated in generating perturbations, making them make incorrect judgments. We propose a new method for generating adversarial point clouds, named MBDP, which decomposes the adversarial point cloud into two sub-perturbations using the decomposition perturbation method. The momentum iterative fast sign algorithm is used to optimize both the sub-perturbation and the main-perturbation simultaneously, generating adversarial samples that are far from the decision boundary and more transferable. Experimental results show that both on real and synthetic 3D datasets, our proposed MBDP achieve the hightest attack success rate and transferability score.

1. Introduction

With the rapid development of artificial intelligence technology and mobile mapping technology, deep neural networks are widely used in visual tasks such as 2D and 3D object detection and recognition. Due to the rapid development of mobile measurement technology, 3D point cloud data has gradually become the mainstream data format in the field of intelligent driving. The correct recognition and classification of 3D point clouds is an important guarantee for the application of artificial intelligence systems in related fields.

In the field of intelligent driving, vehicle-mounted LiDAR can capture surrounding 3D point cloud data with high accuracy and precision. Deep neural network models can accurately identify and locate surrounding vehicles, pedestrians, and other traffic objects by analyzing and processing 3D point cloud data. However, existing deep neural network models exhibit fragility when facing perturbation attacks (Zheng et al., 2023). Adversarial perturbation refers to the addition of carefully designed small noises to 3D point cloud data, which are difficult to detect visually by humans but can mislead deep learning models and cause them to output incorrect results. Perturbation attacks refer to the use of adversarial perturbation techniques to attack a 3D point cloud model, with the aim of making the model unable to correctly identify targets when faced with perturbed point cloud data. This kind of disturbance attack will seriously affect the intelligent driving system, causing it to make incorrect judgments (Zheng et al., 2024). Such erroneous judgments and decisions often lead to serious traffic safety hazards and even traffic accidents, so it is very important to study 3D point cloud perturbation attacks in the field of intelligent driving.

Most existing perturbation attacks are based on white box attacks, where attackers can fully access the structure and parameters of the model and directly generate adversarial samples using gradient information. However, in real-world traffic scenarios, most attacks can only query the model and cannot obtain detailed parameters, which cannot meet the requirements of white box attacks (Guo et al., 2025). Therefore, studying the transferability of perturbation attacks has more important practical significance. Transferability refers to the similar misleading ability of the same adversarial sample between different models, and the transferability of adversarial point clouds generated by one model can also deceive other models. Figure 1 illustrates the concept of transferability. Studying the transferability of perturbation attacks is an important guarantee for the successful deployment of artificial intelligence models in the field of intelligent driving.

In this paper, we propose a new method, Momentum-based Decompose Perturbation (MBDP), to enhance the transferability of point cloud perturbation attacks. We use the decomposition perturbation method to decompose the adversarial perturbation into two sub-perturbations, and then use momentum-based optimization methods to constrain the direction of the adversarial perturbation, iterative optimization generates transferable adversarial samples far from the decision boundary.

The main contributions of this work are as follows:

- We propose an attack method, MBDP, that optimizes perturbations and their decomposed sub-perturbations to generate more transferable 3D adversarial samples.
- We embed momentum iterative fast gradient sign algorithm to optimize perturbations and sub-perturbations, which effectively improve their information capture in high-dimensional space.
- The experiments both on real and synthetic datasets achieve the hightest attack success rate and transferability score.

* Corresponding author, Email: huangxw@jmu.edu.cn

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-W2-2025-83-2025 | © Author(s) 2025. CC BY 4.0 License.

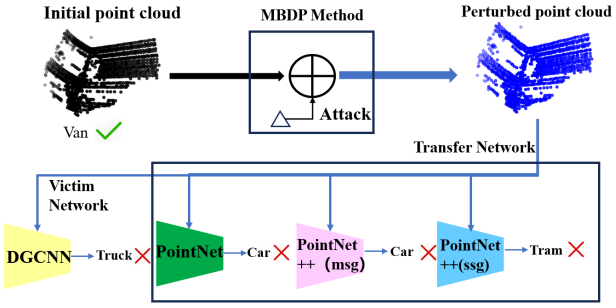

Figure 1. A schematic figure of the transferability of 3D point cloud adversarial perturbations. We deceive the victim network DGCNN by adding perturbations to the clean van point cloud; The generated perturbation point cloud can not only make the target network make incorrect judgments, but also cause other transfer network classification models that did not participate in generating perturbations to produce incorrect categories.

2. Related Work

2.1 Progress of Deep Learning in 3D Point Clouds

As a core representation of 3D data, point clouds have been extensively applied across a wide range of practical scenarios. With the continuous advancement of deep learning techniques, 3D point cloud processing has progressively emerged as a pivotal task within the realms of computer vision and robotics. This section aims to provide an overview of the recent research progress in the application of deep learning to 3D point cloud processing.

Point cloud enhancement, which is predicated on the utilization of deep learning methods, is primarily concerned with transforming low-quality raw point clouds into dense, clean, and complete point clouds. This transformation is achieved through a combination of denoising (Pistilli et al., 2020), completion , and upsampling techniques (Lin et al., 2020), thereby significantly enhancing the overall performance of point clouds.

Point cloud classification, as a fundamental task in point cloud analysis, is focused on assigning labels to individual points within a point cloud to identify their respective attributes. With the evolution of deep learning technology, direct processing methods for point clouds have gradually become the mainstream approach. Notably, PointNet (Qi et al., 2017a) and its variants, such as PointNet++ (Qi et al., 2017b), have been widely adopted and have demonstrated remarkable effectiveness in this domain. Significant progress has also been made in point cloud semantic segmentation through the application of deep learning. Researchers have proposed a variety of methods based on convolutional neural networks (CNNs) (Chua, 1997), graph neural networks (GNNs) (Chen et al., 2019), and attention mechanisms (Niu et al., 2021). These methods are designed to better capture both local and global features of point clouds, thereby improving the accuracy and robustness of semantic segmentation. Furthermore, some researchers have optimized the reconstruction of point clouds by integrating deep learning with clustering models. For instance, variational methods (Pinheiro Cinelli et al., 2021) have been employed to enhance the reconstruction process, thereby achieving more accurate and efficient point cloud reconstruction.

2.2 Adversarial Perturbation Attack on 3D Point Cloud

Point cloud data plays a crucial role in various fields such as object recognition, autonomous driving, and robot navigation. However, recent studies have highlighted that deep learning-based point cloud models are vulnerable to adversarial attacks. These attacks, by introducing meticulously crafted perturbations into the input data, can significantly disrupt the model's prediction accuracy, thereby posing a severe threat to the model's security and robustness. Consequently, investigating adversarial perturbations in point clouds is essential for assessing and enhancing the robustness of point cloud models.

Existing adversarial attack methods for point clouds can be broadly categorized into point-based attacks, optimization-based attacks, and gradient-based attacks. Xiang et al. (Xiang et al., 2019) pioneered an adversarial attack approach for point cloud classification, proposing four distinct attack strategies: point displacement, point addition, point cluster generation, and adversarial object insertion. (Wen et al., 2020) introduced $GeoA^3$, a geometric perception-based optimization method that generates adversarial point clouds with desirable set properties, making them less perceptible to human observers. (Liu et al., 2019) adapted the fast gradient sign method (Goodfellow et al., 2015), commonly used in 2D image attacks, to 3D point clouds by constraining the perturbation magnitude across different dimensions, thereby enhancing the effectiveness of the adversarial samples. Liu et al. (Liu et al., 2025) proposed to exploit the interpretability of 3D deep networks to construct 3D adversarial attacks on salient regions. Zheng et al. (Zheng et al., 2025) proposed to use simulated smoke and water mist superimposed on real targets to achieve adversarial attacks on 3D target recognition.

2.3 The Transferability of Perturbation Attacks

The transferability of point cloud perturbation attacks refers to the characteristic that adversarial perturbations generated for one model can effectively attack other models or datasets. Existing methods to improve the transferability of point cloud perturbation attacks are mainly divided into three categories: generator-based methods, data augmentation and optimization, and transmembrane state transferability.

Xiao et al. (Xiao et al., 2018) trained a generator using a GAN generative adversarial network framework and directly synthesized adversarial perturbations. Their method performed well in black box attacks. Jandial et al. (Jandial et al., 2019) once again proposed AdvGAN++, which introduces the intermediate layer features of the target model as inputs to the generator and improves its cross dataset transfer performance through hidden layer features. Dong et al. (Dong et al., 2019) further improved the transfer ability of adversarial samples across defense models by optimizing perturbation generation through translational invariance. Luo et al. (Luo et al., 2024) found that the task vectors of visual language models can transfer across text and image modalities. By using attention mechanisms to address their multimodal features, the transferability of perturbations can be improved. Guo et al. (Guo et al., 2025) proposed to analyze the target features from the perspective of hypothesis space to achieve transferable adversarial attacks.

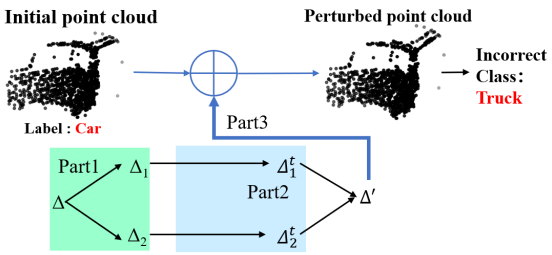

Figure 2. Our method MBDP pipeline diagram, Part 1 (green area) is the decomposition perturbations, Part 2 (blue area) is the momentum iteration optimization T times, and Part 3 is the addition of the final perturbations to generate more transferable adversarial samples

3. Method

3.1 Preparation

The pipeline of MBDP is shown in Figure 2, which consists of the perturbation factorization method and the momentum-based iterative optimization method. First, we use the perturbation factorization method to decompose the adversarial perturbation into two sub-perturbations. Then, we employ the momentum-based iterative optimization method to constrain the direction of the sub-perturbations. Finally, we iteratively optimize both the main perturbation and the sub-perturbations simultaneously to generate adversarial point clouds that are farther away from the decision boundary and thus more transferable.

We define a 3D point cloud $P \in R^{(N*3)}$, where N is the number of points in the point cloud, which is set to 1024 in our experiments. Given the point cloud P with its ground-truth label y, we denote f as the point cloud classification model and Δ as the added perturbation. We aim to find the perturbation Δ such that the model's classification changes before and after the perturbation is added, i.e., $f(P+\Delta) =\neq y$. We consider such a perturbation as a successful one.

We use the perturbation factorization method to decompose the perturbation into two sub-perturbations:

$$\Delta = \Lambda \odot \Delta_1 + (1-\Lambda) \odot \Delta_2, \tag{1}$$

where $\odot$ denotes element-wise multiplication, and Λ is an N-th order arithmetic mask matrix with elements in $\{0,1\}$. Specifically, each element of Λ can be represented as:

$$\Lambda_{xy} \sim \text{Bernoulli}(p) \quad \text{for } x \in \{1,2,\ldots,N\} \text{ and } y \in \{1,2,3\}. \tag{2}$$

In order to reduce the computational burden, we randomly select a Λ' for sampling during multiple iterations.

3.2 Momentum Iterative

In the process of iterative optimization, we used a momentum gradient based method. Unlike common single-step gradient based methods, the momentum iterative gradient method accumulates historical gradient directions, avoiding the randomness of single-step updates and forming a more stable update direction.

Algorithm 1 Momentum-based Decomposed Perturbation Attack

INPUT: Clean sample x, target model f, iterations T, step size α, momentum μ, budget ϵ
OUTPUT: Adversarial example x_{adv}
Initialize velocity vectors $v_1^{(0)} \leftarrow 0$ and $v_2^{(0)} \leftarrow 0$
Initialize perturbations $\Delta_1^{(0)} \leftarrow 0$ and $\Delta_2^{(0)} \leftarrow 0$
Initialize adversarial example $x_{adv}^{(0)} \leftarrow x$
for $t = 0$ **TO** $T-1$ **do**
  Decompose perturbation: $\delta^{(t)} = \Delta_1^{(t)} + \Delta_2^{(t)}$
  for $i = 1, 2$ **do**
    Compute gradient: $g_i^{(t)} \leftarrow \nabla_{\Delta_i} \mathcal{L}(f(x_{adv}^{(t)}), y)$
    Update velocity: $v_i^{(t+1)} \leftarrow \mu \cdot v_i^{(t)} + \frac{g_i^{(t)}}{\|g_i^{(t)}\|_1}$
    Update sub-perturbation: $\Delta_i^{(t+1)} \leftarrow \Delta_i^{(t)} + \alpha \cdot \text{sign}(v_i^{(t+1)})$
  end for
  Aggregate perturbations: $\delta^{(t+1)} \leftarrow \Delta_1^{(t+1)} + \Delta_2^{(t+1)}$
  Project perturbation: $\delta^{(t+1)} \leftarrow \text{clip}(\delta^{(t+1)}, -\epsilon, \epsilon)$
  Update adversarial example: $x_{adv}^{(t+1)} \leftarrow x + \delta^{(t+1)}$
end for
RETURN $x_{adv}^{(T)}$

Firstly, we calculate the gradient $g_i^{(t)}$ of the standard model at the current adversarial sample $x_{adv}^{(t)}$. The direction of the gradient indicates how to adjust δ_i to update the value of the objective function $\mathcal{L}$, thereby generating adversarial samples:

$$g_i^{(t)} = \nabla_{\Delta_i} \mathcal{L}(f(x_{adv}^{(t)}), y) \tag{3}$$

Then we initialize the momentum term. For a given adversarial perturbation, we use Equation 1 to decompose it into two sub-perturbations, calculate gradients for each sub-perturbations, and update the momentum term. The introduction of the momentum term makes the gradient smooth and updates, while normalizing the gradient using the $L1$ norm to ensure that the direction of the update is not affected by the magnitude of the gradient:

$$v_i^{(t+1)} = \mu \cdot v_i^{(t)} + \frac{g_i^{(t)}}{\|g_i^{(t)}\|_1} \tag{4}$$

Use the updated momentum term to update the sub perturbations, and control the update amplitude using the step size parameter α to ensure that the perturbations are updated along the optimization direction with a fixed step size, thereby improving the efficiency of optimization:

$$\Delta_i^{(t+1)} = \Delta_i^{(t)} + \alpha \cdot \text{sign}(v_i^{(t+1)}) \tag{5}$$

In the end, we aggregate the perturbations and project them, update the adversarial samples with new perturbations, and after T iterations, generate the final adversarial samples with more transferability.

The traditional single-step gradient perturbation generation method is prone to interference from local gradient noise in the parameter space, resulting in unstable perturbation directions. We introduce momentum terms to accumulate historical gradient directions, weaken the noise influence of single-step gradients, make the perturbation direction more consistent, focus on key feature regions, gradually approach the decision boundary of the model, and generate adversarial samples that are more likely to deceive the model, achieving better attack success rates.

Victim Network	$\epsilon = 0.18$				$\epsilon = 0.45$			
	PointNet	PointNet++ (SSG)	PointNet++ (MSG)	DGCNN	PointNet	PointNet++ (SSG)	PointNet++ (MSG)	DGCNN
PointNet	100.0	56.1	98.9	77.2	100	57.2	95.8	78.0
PointNet++ (SSG)	69.2	100.0	80.3	75.5	70.3	100.0	80.0	76.4
PointNet++ (MSG)	86.6	70.2	100.0	78.6	87.3	74.0	100.0	76.0
DGCNN	71.1	62.9	79.4	100.0	72.0	63.3	80.1	100.0

Table 1. Attack success rates (%) of MBDP method with different ϵ values on KITTI

3.3 Loss Function

In calculating disturbance loss, we use prediction probability for optimization. We use $g(P)_y$ to represent the probability of the $y-th$ class predicted by the deep neural network model, and use $d(P, \Delta)$ to calculate the difference between the highest probability outside the correct class and the true class probability:

$$d(P, \Delta) = \max_{y' \neq y} \left(g(P+\Delta)_{y'} - g(P+\Delta)_y\right)^2 \tag{6}$$

The preliminary geometric constraint term l_1 is obtained by accumulating the distance loss between the sub-perturbations and the main-perturbation:

$$l_1 = d(P, \Lambda') + d(P, 1-\Lambda') + d(P) \tag{7}$$

We give l_1 a constraint parameter β and introduce both Chamfer distance and Hausdorff distance. The complete geometric constraint terms l_2 are as follows:

$$l_2 = l_{cd} + l_{hd} + \beta * l_1 \tag{8}$$

Among the Chamfer distance measures the average distance between two sets of points and is calculated as follows:

$$\ell_{\text{cd}}(P, P') = \frac{1}{n}\sum_{i=1}^{n} \min_{j=1,\ldots,n} \|P_i - P'_j\| + \frac{1}{n}\sum_{j=1}^{n} \min_{i=1,\ldots,n} \|P'_j - P_i\| \tag{9}$$

And the Hausdorff distance calculates the maximum of the minimum distance from the adversarial point cloud P' to P, and the maximum of the minimum distance from P to P' and the calculation method is as follows:

$$\ell_{\text{hd}}(P, P') = \max\left(\max_{i=1,\ldots,n} \min_{j=1,\ldots,n} \|P_i - P'_j\|, \max_{j=1,\ldots,n} \min_{i=1,\ldots,n} \|P'_j - P_i\|\right) \tag{10}$$

We use the cross entropy loss function as a benchmark and incorporate geometric constraint terms. The loss function we ultimately attempted to optimize is:

$$\min_{\Delta} \ell_{\text{final}} = -\ell_{\text{cls}}(f(P'), y_{\text{true}}) + \tau \cdot l_2 \quad \text{s.t.} \quad \|\Delta\|_\infty \leq \epsilon, \tag{11}$$

where τ is the penalty parameter used to adjust the weight of the entire geometric constraint term. To better understand our method, we propose Algorithm 1.

4. Experiments

4.1 Experimental Preparation

4.1.1 Dataset We conducted our undifferentiated adversarial attack experiment using the real-world scenario dataset KITTI (Wu et al., 2015) and the synthetic dataset ModelNet40. We conducted point cloud extraction based on 3dbbox on KITTI Street scenic spot cloud data, selecting six categories including cars, vans, trucks, pedestrians, bicycles, and trams for classification experiments. We selected 2000 car point clouds and 800 other categories, totaling 6000 point clouds, as our processed KITTI dataset, with 4800 as training samples and 1200 as test samples. ModelNet40 is a widely used dataset for training and evaluating model performance, consisting of 12311 CAD models and 40 different object categories, with 9843 samples for training and 2468 samples for testing.For the KITTI dataset, we randomly selected 40 point clouds from each category, totaling 240, to generate adversarial samples. For the ModelNet40 dataset, we randomly selected 25 point clouds from ten categories, totaling 250, to generate adversarial samples.

4.1.2 Model We used the common PointNet (Qi et al., 2017a) and its variants, $PointNetPP++(SSG)$ and $PointNetPP++(MSG)$ (Qi et al., 2017b), as well as the DGCNN (Wang et al., 2019), as the attacked and evaluated models. The PointNet family of networks is trained strictly according to $GeoA^3$ (Wen et al., 2020), and the parameters of the DGCNN are set as follows: k=20, emb-dims=1024, dropout=0.5.

4.1.3 Parameter Settings We use PF-Attack (He et al., 2023) as the baseline for our experiment, with the following specific parameter settings: $\tau = 10$, $\beta = 0.5$, $\eta = 0.01$, $p = 0.5$, $\epsilon \in \{0.18, 0.45\}$. Momentum optimization part (Dong et al., 2018): mometum=0.9 and $step - szie$=0.01;

4.1.4 Evaluation Metrics We use attack success rate(ASR) and transferability score to evaluate the effectiveness of perturbation attacks. The success rate of attacks refers to the proportion of samples that result in misclassification of the model after adding adversarial measures in the total number of test samples. The transferability score is calculated by weighting the success rate of attacks on a total of four experimental models with perturbations. We believe that the higher the success rate and transferability score of perturbation attacks, the stronger the transferability of perturbation attacks.

4.2 Experimental Results

4.2.1 Results on KITTI The KITTI dataset we processed contains 6000 point clouds of traffic objects, including 2000 car labeled point clouds, 800 cyclist, pedestrian, tram, truck, and van labeled point clouds each, divided into 4800 point clouds

Victim Network	Attack Method	$\epsilon = 0.18$				$\epsilon = 0.45$			
		PointNet	PointNet++ (MSG)	PointNet++ (SSG)	DGCNN	PointNet	PointNet++ (MSG)	PointNet++ (SSG)	DGCNN
PointNet	3D-Adv	100	8.4	10.4	6.8	100	8.8	9.6	8.0
	KNN	100	9.6	10.8	6.0	100	9.6	8.4	6.4
	$GeoA^3$	100	20.0	19.6	7.2	100	23.6	20.8	7.2
	AdvPC	98.8	20.4	27.6	22.4	98.8	18.0	26.8	20.4
	MBDP (Ours)	100	**30.8**	**46.1**	**46.2**	100	**32.5**	**46.6**	**47.3**
PointNet++ (MSG)	3D-Adv	6.8	100	28.4	11.2	7.2	100	29.2	11.2
	KNN	6.4	100	22.0	8.8	6.4	100	23.2	7.6
	$GeoA^3$	4.4	100	14.4	6.4	4.4	100	13.6	6.0
	AdvPC	13.2	97.2	54.8	39.6	18.4	98	58.0	39.2
	MBDP (Ours)	**27.2**	100	**93.7**	**55.7**	**27.0**	100	**92.5**	**55.4**
PointNet++ (SSG)	3D-Adv	7.6	9.6	100	6.0	7.2	10.4	100	7.2
	KNN	6.4	9.2	100	6.4	6.8	7.6	100	6.0
	$GeoA^3$	5.2	10.4	100	2.2	4.8	9.2	100	4.0
	AdvPC	12.0	27.2	100	22.8	14.0	30.8	100	27.6
	MBDP(Ours)	**27.5**	**83.5**	**100**	**56.0**	**25.1**	**91.3**	**100**	**54.1**
DGCNN	3D-Adv	9.2	11.2	31.2	100	9.6	12.8	30.4	100
	KNN	7.2	9.6	14.0	99.6	6.8	10.0	11.2	99.6
	$GeoA^3$	4.4	27.2	27.6	100	4.4	26.8	25.6	100
	AdvPC	19.6	46.0	64.4	94.8	32.8	48.8	64.4	97.2
	MBDP(Ours)	**35.0**	**74.0**	**81.1**	100	**47.7**	**92.0**	**92.6**	100

Table 2. The presentation of the success rates of various attack methods on the ModelNet40 dataset. The results of 3D-Adv, KNN and AdvPC are reported in (Hamdi et al., 2020). Number in bold indicates the best.

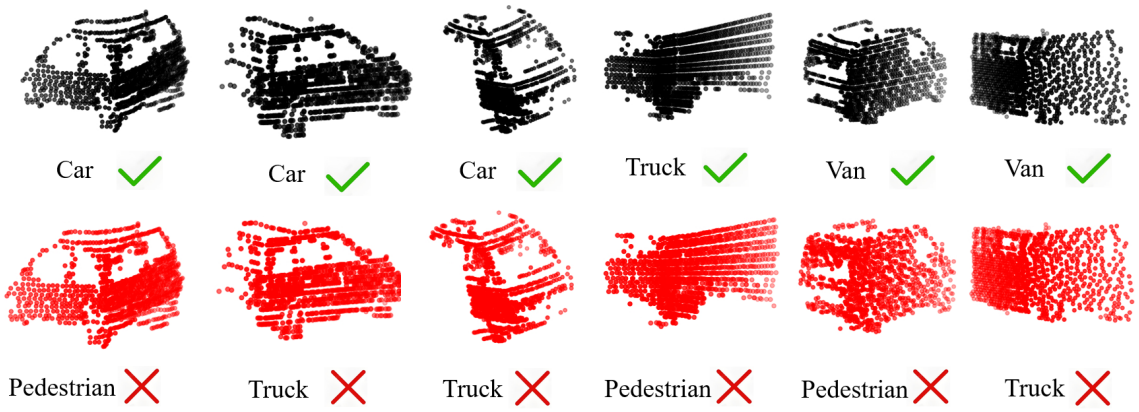

Figure 3. The visualization effect of point clouds on the KITTI dataset shows that the first row (black) of point clouds is a clean initial point cloud, and the second row (red) of point clouds is an adversarial point cloud perturbed by our MBDP method.

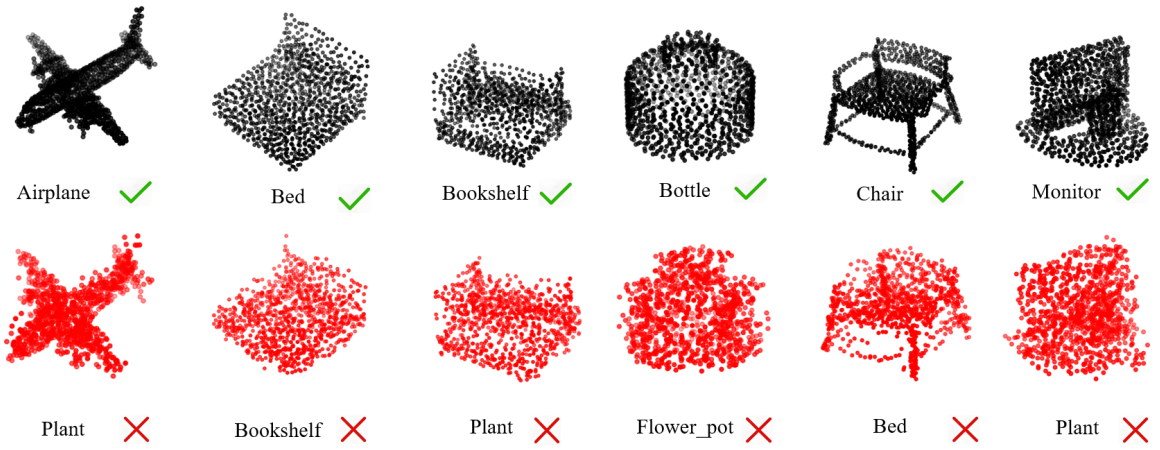

Figure 4. The visualization effect of point clouds on the ModelNet40 dataset shows that the first row (black) of point clouds is a clean initial point cloud, and the second row (red) of point clouds is an adversarial point cloud perturbed by our MBDP method.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-W2-2025-83-2025 | © Author(s) 2025. CC BY 4.0 License.

ϵ	Transferability Score(%)			
	3D-Adv	KNN	$GeoA_3$	MBDP(Ours)
0.18	12.2	9.7	12.4	**54.7**
0.45	12.6	9.2	12.5	**58.3**

Table 3. Compare different attack methods in transferability score on ModelNet40.

for the training set and 1200 point clouds for the testing machine. We randomly selected 40 point clouds from each category, totaling 240 point clouds, for attack experiments. The experimental results are shown in Table 1. At the same time, we calculated the transferability score of perturbation attacks, which is 75.5 when $\epsilon = 0.18$ and 75.9 when $\epsilon = 0.45$.

In order to verify the effectiveness of our attack method from multiple perspectives, we visualized the initial point clouds and successfully perturbed adversarial point clouds of some categories. It can be observed that while the attack success rate and transferability score are high, the added perturbations still have good imperceptibility, as shown in Figure 3.

4.2.2 Results on ModelNet40 For the ModelNet40 dataset, we strictly followed the previous work for parameter settings, normalized and sampled all point clouds to the same 1024 points. We use the entire training dataset to train the victim model. We also randomly selected ten out of 40 categories from the ModelNet40 dataset in the test set, and randomly selected 25 samples from each category to form the data used to evaluate perturbation attack methods. The experimental results are shown in Table 2. Based on the success rate of the attack, we calculated the transferability score of a at different values, as shown in Table 3.

Similarly, we randomly selected several categories of point cloud samples from the ModelNet40 dataset for visualization, verifying that our method achieves high attack success rates while also having good imperceptibility, as shown in Figure 4.

5. Conclusion

In this paper, we propose a method called MBDP for generating 3D point clouds with transferability to counteract perturbations. We explore the effective information contained in the sub perturbations generated by random decomposition adversarial perturbations. By using the momentum iterative fast gradient sign algorithm to optimize both the main perturbation and sub perturbations, we can more effectively capture information in high-dimensional space. Introducing a momentum term can accumulate historical gradient directions, making the perturbation direction closer to the negative gradient main direction of the loss function, thereby minimizing the adversarial loss function more efficiently and generating adversarial samples that are more transferable away from the decision boundary. The research on adversarial sample generation technology is beneficial for improving the robustness of neural network models and enhancing the safety of intelligent driving systems, and the research on the transferability of adversarial samples has, to some extent, accelerated the efficient implementation of various applications of artificial intelligence systems in the physical world.

6. Acknowledgement

This work was supported by the Natural Science Foundation of Xiamen, China (No. 3502Z202472018, 3502Z202373035); the National Natural Science Foundation of China (No. 62401225); the Natural Science Foundation of Fujian Province, China (No. 2024J01115, 2024J01117, 2024J01723); the Jimei University Scientific Research Start-up Funding Project (No. ZQ2024034).

References

Chen, Z., Villar, S., Chen, L., Bruna, J., 2019. On the equivalence between graph isomorphism testing and function approximation with gnns. *Advances in Neural Information Processing Systems (NeuralIPS)*, 32.

Chua, L. O., 1997. CNN: A vision of complexity. *International Journal of Bifurcation and Chaos*, 7(10), 2219–2425.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., Li, J., 2018. Boosting adversarial attacks with momentum. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 9185–9193.

Dong, Y., Pang, T., Su, H., Zhu, J., 2019. Evading defenses to transferable adversarial examples by translation-invariant attacks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 4312–4321.

Goodfellow, I. J., Shlens, J., Szegedy, C., 2015. EXPLAINING AND HARNESSING ADVERSARIAL EXAMPLES. *stat*, 1050, 20.

Guo, Y., Liu, W., Xu, Q., Zheng, S., Huang, S., Zang, Y., Shen, S., Wen, C., Wang, C., 2025. Boosting adversarial transferability through augmentation in hypothesis space. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*.

Hamdi, A., Rojas, S., Thabet, A., Ghanem, B., 2020. Advpc: Transferable adversarial perturbations on 3d point clouds. *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, Springer, 241–257.

He, B., Liu, J., Li, Y., Liang, S., Li, J., Jia, X., Cao, X., 2023. Generating transferable 3d adversarial point cloud via random perturbation factorization. *Proceedings of the AAAI Conference on Artificial Intelligence(AAAI)*, 37number 1, 764–772.

Jandial, S., Mangla, P., Varshney, S., Balasubramanian, V., 2019. Advgan++: Harnessing latent layers for adversary generation. *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops(ICCV)*, 0–0.

Lin, Z.-H., Huang, S.-Y., Wang, Y.-C. F., 2020. Convolution in the cloud: Learning deformable kernels in 3d graph convolution networks for point cloud analysis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 1800–1809.

Liu, D., Yu, R., Su, H., 2019. Extending adversarial attacks and defenses to deep 3d point cloud classifiers. *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2279–2283.

Liu, W., Liu, M., Zheng, S., Shen, S., Bian, X., Zang, Y., Zhong, P., Wang, C., 2025. Interpreting Hidden Semantics in the Intermediate Layers of 3D Point Cloud Classification Neural Network. *IEEE Transactions on Multimedia*, 27, 965-977.

Luo, G., Darrell, T., Bar, A., 2024. Task Vectors are Cross-Modal. *arXiv preprint arXiv:2410.22330*.

Niu, Z., Zhong, G., Yu, H., 2021. A review on the attention mechanism of deep learning. *Neurocomputing*, 452, 48–62.

Pinheiro Cinelli, L., Araújo Marins, M., Barros da Silva, E. A., Lima Netto, S., 2021. Variational autoencoder. *Variational Methods for Machine Learning with Applications to Deep Networks*, Springer, 111–149.

Pistilli, F., Fracastoro, G., Valsesia, D., Magli, E., 2020. Learning robust graph-convolutional representations for point cloud denoising. *IEEE Journal of Selected Topics in Signal Processing(JSTSP)*, 15(2), 402–414.

Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 652–660.

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in Neural Information Processing Systems(NeuralIPS)*, 30.

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

Wen, Y., Lin, J., Chen, K., Chen, C. P., Jia, K., 2020. Geometry-aware generation of adversarial point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence(TPAMI)*, 44(6), 2984–2999.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J., 2015. 3d shapenets: A deep representation for volumetric shapes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 1912–1920.

Xiang, C., Qi, C. R., Li, B., 2019. Generating 3d adversarial point clouds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 9136–9144.

Xiao, C., Li, B., Zhu, J. Y., He, W., Liu, M., Song, D., 2018. Generating adversarial examples with adversarial networks. *27th International Joint Conference on Artificial Intelligence, IJCAI 2018*, International Joint Conferences on Artificial Intelligence, 3905–3911.

Zheng, S., Liu, W., Guo, Y., Zang, Y., Shen, S., Wang, C., 2025. A new adversarial perspective for lidar-based 3d object detection. *Proceedings of the AAAI Conference on Artificial Intelligence(AAAI)*.

Zheng, S., Liu, W., Guo, Y., Zang, Y., Shen, S., Wen, C., Cheng, M., Zhong, P., Wang, C., 2024. SR-Adv: Salient Region Adversarial Attacks on 3D Point Clouds for Autonomous Driving. *IEEE Transactions on Intelligent Transportation Systems(TITS)*.

Zheng, S., Liu, W., Shen, S., Zang, Y., Wen, C., Cheng, M., Wang, C., 2023. Adaptive local adversarial attacks on 3D point clouds. *Pattern Recognition(PR)*, 144, 109825.