AE-Morpher: Improve Physical Robustness of Adversarial Objects against LiDAR-based Detectors via Object Reconstruction

Shenchen Zhu, *Institute of Information Engineering, Chinese Academy of Sciences, China; School of Cyber Security, University of Chinese Academy of Sciences, China;*
Yue Zhao, *Institute of Information Engineering, Chinese Academy of Sciences, China;*
Kai Chen, *Institute of Information Engineering, Chinese Academy of Sciences, China; School of Cyber Security, University of Chinese Academy of Sciences, China;*
Bo Wang, *Huawei Technologies Co., Ltd.;* Hualong Ma and Cheng'an Wei, *Institute of Information Engineering, Chinese Academy of Sciences, China; School of Cyber Security, University of Chinese Academy of Sciences, China*

https://www.usenix.org/conference/usenixsecurity24/presentation/zhu-shenchen

This paper is included in the Proceedings of the 33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

978-1-939133-44-1

Open access to the Proceedings of the 33rd USENIX Security Symposium is sponsored by USENIX.

AE-Morpher: Improve Physical Robustness of Adversarial Objects against LiDAR-based Detectors via Object Reconstruction

Shenchen Zhu[1,2], Yue Zhao[1]*, Kai Chen[1,2]*, Bo Wang[3], Hualong Ma[1,2], Cheng'an Wei[1,2]
[1]*Institute of Information Engineering, Chinese Academy of Sciences, China,*
[2]*School of Cyber Security, University of Chinese Academy of Sciences, China,*
[3]*Huawei Technologies Co., Ltd.*

Abstract

LiDAR-based perception is crucial to ensure the safety and reliability of autonomous driving (AD) systems. Though some adversarial attack methods against LiDAR-based detectors perception models have been proposed, deceiving such models in the physical world is still challenging. While existing robustness methods focus on transforming point clouds to embed more robust adversarial information, our research reveals how to reduce the errors during the LiDAR capturing process to improve the robustness of adversarial attacks. In this paper, we present AE-Morpher, a novel approach that minimizes differences between the LiDAR-captured and original adversarial point clouds to improve the robustness of adversarial objects. It reconstructs the adversarial object using surfaces with regular shapes to fit the discrete laser beams. We evaluate AE-Morpher by conducting physical disappearance attacks that use a mounted adversarial ornament to conceal a car from models' detection results in both SVL Simulator environments and real-world LiDAR setups. In the simulated world, we successfully deceive the model up to 91.1% of the time when LiDAR moves towards the target vehicle from 20m away. On average, our method increases the ASR by 38.64% and reduces the adversarial ornament's projection area by 67.59%. For the real world, we achieve an average attack success rate of 71.4% over a 12m motion scenario. Moreover, adversarial objects reconstructed by our method can be easily physically constructed by human hands without the requirement of a 3D printer.

1 Introduction

Advancements in artificial intelligence and sensor technology have significantly accelerated the development of autonomous driving (AD) in recent years, bringing it closer to widespread adoption. Presently, a considerable number of vehicles on the road incorporate autonomous features, with their prevalence continually expanding. Systems like Tesla's Autopilot and GM's Super Cruise can even partially take over vehicle control, handling specific driving tasks. Consequently, the safety of autonomous driving systems has emerged as an increasingly critical concern.

LiDAR sensors and LiDAR-based perception models are critical components of AD systems' perception modules. However, the vulnerability of deep neural networks (DNNs) to adversarial examples (AEs) has raised concerns regarding the security of LiDAR-based perception models. Some studies [9, 12, 13, 15, 16, 40, 43, 49] have explored the use of adversarial object attacks to deceive LiDAR-based perception models to output incorrect perceptions that could potentially lead to accidents. Nevertheless, conducting an effective physical-world attack against LiDAR-based perception models is challenging. Because these models typically operate in dynamic environments, deceiving them demands more robust AEs capable of handling varying relative distances and viewing angles.

For the physical adversarial attack against camera-based perception models, many robustness improvement methods [17–19, 24, 25, 27, 29, 30, 32, 47] have been proposed. However, to the best of our knowledge, effective robustness improvement methods for physical adversarial object attacks against LiDAR-based perception models remain limited. Some approaches [20, 28, 33, 39, 44] introduce a surface smoothness function during adversarial optimization. However, these methods are designed for dense 3D scanner point clouds and are unfeasible for sparse LiDAR point clouds. Transformation-based methods [11, 45, 46], such as EOT [11], has been employed in MSF-ADV [13]. However, the effectiveness is mainly proved on small objects like traffic cones or boxes and presents additional challenges in terms of convergence.

Therefore, this paper aims to propose an effective robustness improvement method to enable more threatening and realistic physical adversarial attacks against LiDAR-based detection models. To achieve it, we need to consider the following two key questions:

Q1: What causes the loss in the robustness of physical

*The corresponding authors.

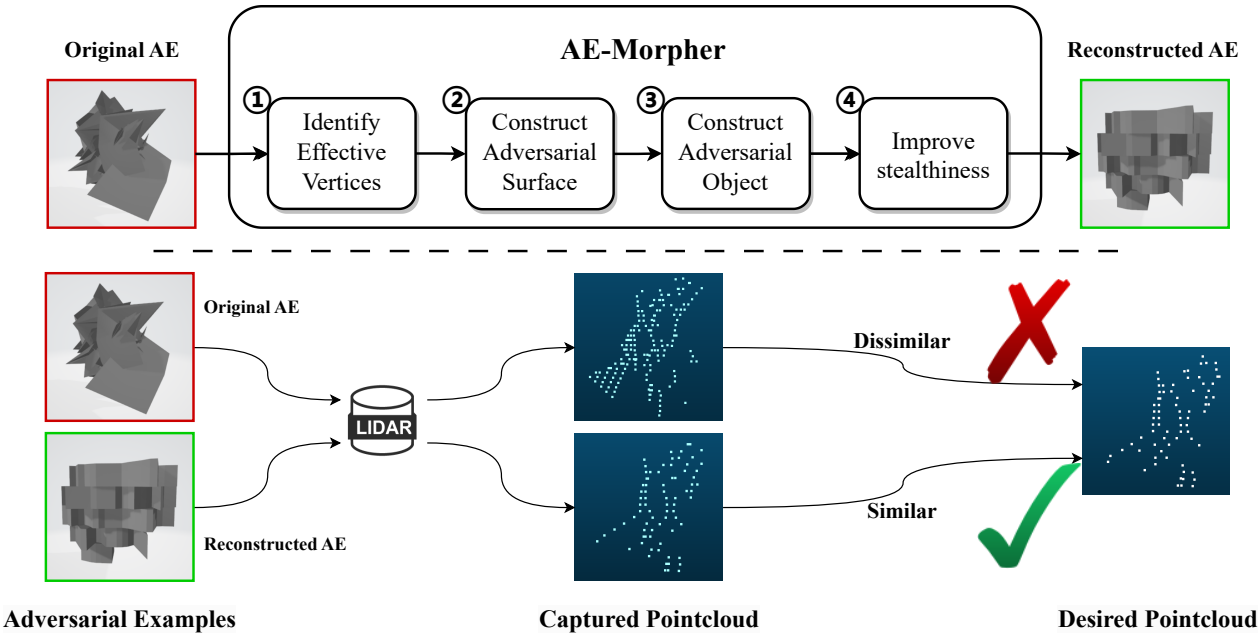

Figure 1: Overview of AE-Morpher. The desired adversarial point cloud p_{adv} can successfully deceive the target model in the digital realm, representing how the attack expects the original AE to be perceived by LiDAR. However, the captured point cloud of the original adversarial object often differs from this desired representation. Therefore, we propose AE-Morpher, a method aimed at reconstructing the original adversarial object to minimize the discrepancies between the captured point cloud and the desired point cloud.

adversarial attacks against LiDAR-based detection models?

In order to attack a LiDAR-based perception model in the real world, the attacker needs to generate an adversarial point cloud p_{adv}, transform it into a 3D object x_{3D}, and 3d print it as x'_{3D}. The LiDAR then captures the beam reflected from x'_{3D}, generating point cloud p'_{adv} and feeds it to the perception model.

Upon comparing multiple pairs of p_{adv} and p'_{adv}, we have unexpectedly found that they are likely to be quite different. According to our analysis in Section 3, the primary factor behind it is the difficulty of sparse laser beams in accurately capturing the scattered perturbations of adversarial objects, particularly those with irregular surfaces and sharp protrusions. We present an example to facilitate the initial understanding. For an object shown in Figure 2, the captured point clouds differ when the laser beams hit various locations. We also conduct a real-world experiment to confirm this insight, please refer to Appendix A.2 for more details. In a physical attack scenario, due to factors such as the location offset between the digital and physical adversarial objects during placement, and the change in relative position between the LiDAR and the physical adversarial objects, it is hard to ensure that the point cloud p'_{adv} captured by the LiDAR is the desired point cloud p_{adv}. The discrepancies between p_{adv} and p'_{adv} compromise the robustness of the adversarial object. For a more detailed analysis, please refer to Section 3.

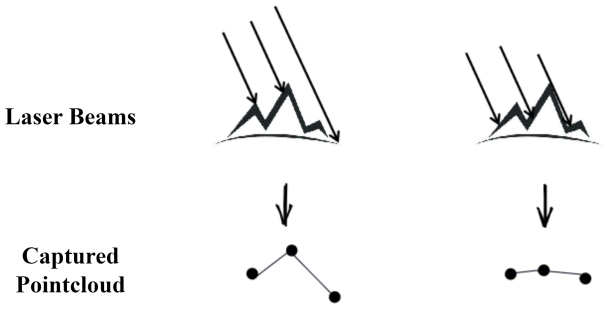

Figure 2: The LiDAR-captured point cloud differs when the hit points of laser beams change.

Q2: How can we improve the robustness of physical adversarial attacks against LiDAR-based perception models?

Based on Q1, scattered adversarial perturbations are difficult to effectively capture by sparse LiDAR signals and result in the discrepancies between p_{adv} and p'_{adv}, compromising the robustness of the adversarial object. To address it, we propose AE-Morpher, a novel AE Reconstruction method. Rather than seeking more robust adversarial perturbations, we aim to optimize the presentation of adversarial perturbations to minimize perturbation distortions during the LiDAR

capturing process. Specifically, as depicted in Figure 1, we first identify the effective perturbations of a given adversarial object, scattered points on the object. Then, we expand them by constructing adversarial surfaces for each perturbation to make them easier to capture while eliminating redundant details of the adversarial object. With adversarial surfaces, we construct a new adversarial object and enhance its stealthiness by reducing the volume at selective positions. By doing so, the captured point cloud can closely resemble the digital point cloud and maintain the adversarial attack capability in the physical world. Importantly, our method does not interfere with the adversarial optimization stage, thereby avoiding any additional challenges in model convergence.

We validate our proposed robustness improvement method with disappear attacks against LiDAR-based perception models. Specifically, we generate an adversarial ornament and attach it to a car to conceal the car from the detection of the following cars. We conduct extensive experiments in both a simulated environment and the real world. Our results show that the reconstructed adversarial ornaments achieve an attack success rate of 100% over a 20m motion scenario in Apollo. On average, our method increases the ASR by 38.64% and reduces the adversarial ornament's projection area by 67.59% compared to the original adversarial ornament. For the real world, we achieve an average attack success rate of 71.4% over a 12m motion scenario. Notably, our reconstructed objects do not require expensive 3D printing technology. While a 3D printer capable of producing an object measuring approximately 60*58*38 cm is priced at $29,999 on Amazon [5], our method can be achieved through simple manual cutting of low-cost materials, such as corrugated cardboard or wood board, and the whole cost is less than $15.

In summary, this work makes the following contributions:

- *New finding.* By delving into LiDAR capturing principles, we are the first to reveal that both the sparse LiDAR laser beams and the scattered adversarial perturbations jointly play a crucial role in the physical adversarial attack robustness against LiDAR-based perception models.
- *New technique.* We propose AE-Morpher, a novel approach to improve the robustness of AEs against LiDAR-based perception models in the physical world. AE-Morpher adjusts the geometric properties of adversarial objects to fit the discrete LiDAR signals by reconstructing their surfaces, thereby improving the physical attack robustness of AEs.
- We evaluate AE-Morpher by conducting physical disappearance attacks on cars in both SVL Simulator environments and the real world. The results show that our approach can effectively improve the attack robustness of adversarial objects. Video demonstrations can be found at https://sites.google.com/view/ae-morpher.

2 Background and Related Work

In this section, we provide a summary of LiDAR's development and applications, as well as an overview of existing attacks against LiDAR-based object detection models and related robustness improvement methods.

2.1 LiDAR Sensors and Detection Principles

LiDAR is a remote sensing technology that employs laser pulses to measure distances and generate 3D maps. It's extensively utilized in autonomous driving to provide precise information about nearby objects, including vehicles, pedestrians, road signs, and lane markings. Prominent autonomous driving platforms, including Baidu Apollo [1] and Autoware [3], incorporate LiDAR as their principal sensor for object detection. Although there are different types of LiDAR, including spinning LiDAR [2], rotating mirror LiDAR [6], and optical phased array LiDAR [23], which rely on different ways of manipulating the laser beam, they all follow a similar detection principle.

The detection principles of LiDAR. The LiDAR sensor first emits a laser beam from a transmitter, which is directed toward the target direction using a rotating mirror or scanning mechanism. When the pulse encounters an obstacle, it reflects back toward the LiDAR, where it is detected by a receiver. The time taken for the pulse to travel back and forth can be used to calculate the distance between the transmitter and the obstacle, considering the speed of light is constant, as illustrated in Figure 3.

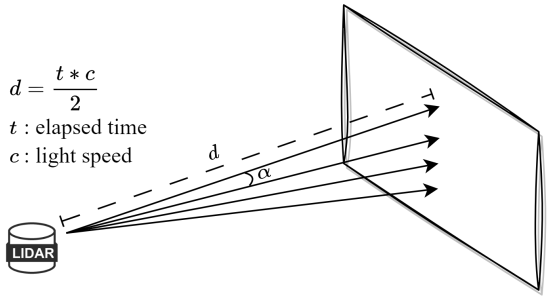

Figure 3: LiDAR detects the position of obstacles by recording the time spent on the round trip of the laser beam.

Angular (horizontal/vertical) resolution is a crucial metric of LiDAR sensors, referring to the angular interval between laser beams. To achieve a wider field of view, the laser emitter rotates horizontally and emits laser light at specific intervals, which represents the horizontal resolution. To determine the height of obstacles, LiDAR sensors adopt an x-line design, emitting multiple laser beams from top to bottom simultaneously, with a certain angle between each beam. This angle, denoted as α in Figure 3, represents the vertical resolution of

the LiDAR system. A higher angular resolution corresponds to more emitted beams and a denser point cloud. Currently, LiDAR sensors equipped on vehicles typically have an angular resolution ranging from 0.1 to 0.4 for horizontal and 0.17 to 2 for vertical. To illustrate, an angular resolution of 0.2° implies a distance of 17.5cm between two adjacent points at a range of 50 m. As a result, LiDAR is likely to acquire different point clouds for an object at a distance of 50m when its surfaces are irregular (e.g., with significant protrusions or depressions within a 17.5cm range), even with minimal movement. Therefore, the instability of point cloud acquisition may occur when the surface irregular deformation of the object is smaller than the radar's resolution, which is a primary issue our robustness improvement method aims to address in this paper.

2.2 Attacks against LiDAR-based Object Detection Models

Due to the unique scanning principles of LiDAR, several LiDAR-specific attack methods have been proposed. These attacks can be broadly divided into three categories: spoofing attacks [14, 21, 22, 34, 37, 38, 42], arbitrary object attacks [48, 49] and adversarial object attack [9, 12, 13, 15, 16, 40, 43].

In spoofing attacks, the attacker employs an additional laser emitter to project laser beams toward the LiDAR system, thereby introducing spoofed reflection points into the point cloud generated by the LiDAR to deceive the perception models. As for arbitrary object attacks, they utilize the location of the objects to mislead the models. To be specific, [49] intricately devises processes to identify the correct locations. These locations are typically distributed in the space surrounding the target vehicle, necessitating the use of drones. Conversely, adversarial object attacks use the shape of the object for this purpose. Attackers employing adversarial object attacks need to optimize the shape of the adversarial object. Each type of attack has its own set of advantages and disadvantages.

In terms of practicality, spoofing attacks require a complex attack system that includes devices such as signal generators and laser emitters. Additionally, the attacker needs to precisely aim the laser emitted by the laser transmitter at the vehicle's LiDAR. These factors introduce difficulties in launching such attacks, reducing their practicality [26, 38]. For arbitrary object attacks and adversarial object attacks. the arbitrary object attack necessitates the use of multiple extra objects, while the adversarial object attack does not. Furthermore, when the target vehicle is in motion, the drones in [49] must adjust their position according to the vehicle to successfully mislead the model, whereas an adversarial object attack does not require such adjustments. However, although adversarial objects can yield favorable results in simulators, their physical robustness is lacking [41]. This prompts us to propose robustness improvement methods.

In terms of stealthiness, spoof attacks themselves are the most stealthy since the laser beams cannot be perceived by humans. However, the required attack system has a relatively large volume and may raise suspicion when placed on the roadside. arbitrary object attacks are also stealthy as the objects are placed around the target vehicle, although the presence of multiple objects might appear unusual. These objects, however, do not directly contact the target vehicle. The stealthiness of adversarial object attacks is determined by the size of the adversarial object used, which can range from highly stealthy to very noticeable. This consideration compels us to take stealthiness into account when proposing robustness improvement methods.

2.3 Robustness-improvement Method for Attacks against LiDAR-based Object Detection Models

In recent years, several robustness improvement methods specifically designed for point clouds have been proposed [20, 28, 33, 39, 44]. These methods employ techniques such as surface smoothness loss functions or limiting the magnitude of point cloud deformation. However, they primarily focus on dense point clouds generated by 3D scanners, which are substantially different from the sparse point clouds produced by LiDAR. For instance, a 3D scanner-generated point cloud of a car may comprise more than 100k points, while LiDAR-generated point clouds typically contain only hundreds or thousands of points due to resolution differences (millimeter-level for 3D scanners vs centimeter or even ten-centimeter level for LiDAR). Sparse point clouds imply large intervals between points and a loss of detailed structural information. Assessing the smoothness of a region with insufficient proximity points is challenging, rendering these methods unsuitable for LiDAR attack scenarios.

For classical generic robustness improvement methods [11, 45, 46], exemplified by EOT [11], these methods improve the generalization of adversarial objects by applying various transformations, making them robust against small discrepancies between the original point cloud and the captured point cloud. They perform well in some situations [13], but they are initially designed for images and lack targeted optimization for LiDAR scenes, potentially causing difficulties in the convergence of adversarial objects [11].

This situation prompts us to design a robustness improvement method utilizing LiDAR's detection principles to further improve obstacle robustness beyond existing methods.

3 Preliminary Analysis

A 3D object, whether it is a benign object or an adversarial object, is typically described by two main components: vertices, which define the points in 3D space that make up the

object, and faces, which define the polygons connecting these vertices, as shown in Figure 4. Based on this background knowledge, we further discuss the three limitations that the mainstream adversarial objects may have that compromise their robustness and stealthiness.

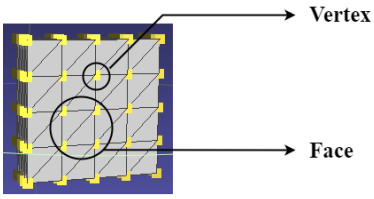

Figure 4: A 3D object consists of vertices (i.e., point clouds) and faces (i.e., polygons). And vertices here are not necessarily to be the corner of the object.

Scattered perturbations are difficult for LiDAR to capture. The adversarial perturbations are mainly introduced into the vertices of the adversarial object, which are scattered points on the object's surface. In the physical world, LiDAR captures points based on laser signals reflected from the surfaces of the object. Due to the sparsity of laser beams, they are more likely to hit the faces of the adversarial object rather than the vertices. However, these faces are not well-designed and do not carry effective perturbations. This means that many effective adversarial perturbations are missed during the LiDAR capturing process.

Irregular surfaces amplify errors during the LiDAR capturing process. Since the vertices carry adversarial perturbations, the ideal adversarial point cloud should exclusively consist of vertices. However, the captured point cloud consistently deviates from the ideal one due to discrepancies between the laser beams' hit points and the object's vertices (as shown in Figure 5). Moreover, irregular surfaces and sharp protrusions can exacerbate these discrepancies. Compared to flat surfaces, the disparity between the coordinates of two adjacent points on an irregular surface with sharp protrusions is more pronounced. Consequently, more errors are introduced when the laser beams do not hit the vertices but instead hit adjacent points on the surface. Ultimately, these discrepancies undermine the physical robustness of the adversarial object.

Figure 5: The LiDAR-captured point cloud (left) usually differs from the desired one (right).

Redundant vertices enlarge the volume of adversarial objects. During the adversarial optimization process, not all vertices of the original object are involved, which means many of these vertices are actually redundant, and carry minimal adversarial information. Nevertheless, these redundant vertices introduce additional surface details, such as sharp protrusions, on the adversarial object. While these protrusions contribute less to the effectiveness of the adversarial attack, they add unnecessary volume to the object and pose challenges during the physical creation of the adversarial object. As a result, the presence of these protrusions undermines the stealthiness and practicality of the adversarial object.

4 Approach Overview

In this section, we describe our high-level intuition for improving the physical attack robustness of adversarial objects and the threat model of our method.

Key intuition. Our approach is derived from the analysis discussed in Section 3. We recognize that the scattering of perturbations and the irregularity of surfaces play a joint role in compromising the robustness of the adversarial object in the physical world. Additionally, the presence of redundant vertices results in unnecessary volume in the adversarial object, compromising the robustness of the adversarial object in the physical world. Intuitively, if we can solve these problems, both the physical robustness and the stealthiness of the adversarial object will increase.

Firstly, we identify and remove vertices with minimal contribution to the attack efficacy from the original adversarial object in order to decrease its volume. To accomplish this, we simulate the LiDAR capture process by sampling multiple times from all vertices of the original adversarial object. We evaluate the attack performance of these sampled vertices and select the ones that are most effective.

Secondly, we incorporate adversarial perturbations from vertices into edges or surfaces, so the adversarial perturbations are not limited to discrete spots but expand to continuous lines and areas, which makes the adversarial perturbations more easily captured by LiDAR. To achieve this, we reconstruct the surface of the adversarial object using flat polygons, as illustrated in Figure 6. The coordinates of these polygons are derived from the remaining vertices, thereby inheriting the adversarial perturbations. Consequently, the new object exhibits adversarial perturbations using surfaces instead of vertices.

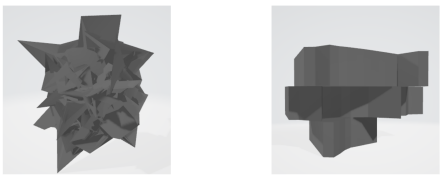

Figure 6: The original (left) and the reconstructed (right) adversarial object.

Threat Model. We assume that the attacker has successfully generated a digital adversarial object using a known method. To apply our method and enhance the object's robustness in the physical world, the attacker requires additional information. This information includes the position of the LiDAR on the victim vehicle, as well as its vertical and horizontal angle resolution. Acquiring these details is relatively straightforward since the attacker can refer to the public manual of the target LiDAR to obtain its performance parameters and examine similar vehicles to determine its installation position. Additionally, for the adversarial object to function effectively, the attacker must have the capability to physically create the object and position it as desired.

Overview. As depicted in Figure 1, AE-Morpher consists of four main steps: (1) Identifying effective adversarial vertices: Given an original adversarial object M, we identify the vertices that carry effective adversarial perturbations. (2) Constructing adversarial faces: We generate an adversarial face f_{adv} for each effective vertex, ensuring that each point on f_{adv} possesses the corresponding adversarial perturbation. (3) Constructing the adversarial object: We obtain an adversarial surface by connecting the adversarial faces into a folded surface. This surface is then translated in space to give it thickness, ultimately resulting in a 3D object. (4) Enhancing stealthiness: We fine-tune the size of the surface to strike a balance between the object's robustness and stealthiness. This enables customization of the object according to specific requirements.

5 Approach Details

5.1 Identifying Effective Adversarial Vertices

The first step is identifying effective vertices from the original adversarial object. Since not all vertices of the original adversarial object effectively attack the LiDAR-based detection model M, there is no need to generate adversarial faces for all vertices. Furthermore, by using fewer vertices, we can construct a smaller adversarial object, thereby enhancing its stealthiness. Our objective is to select a set of fewer vertices that can effectively attack M.

We utilize Ray Cast rendering [7] to help select effective vertices from all vertices of a given original adversarial object. Ray casting simulation emulates the LiDAR capture process to generate the rendered point cloud of the original adversarial object. It selectively preserves specific vertices of the 3D object and utilizes them to generate a point cloud corresponding to the 3D object. The vertex selection process primarily takes into account the object's position, angle, and occlusion. Next, we input the rendered point cloud into M and analyze the output of M. If the rendered point cloud successfully deceives M, we consider the vertices comprising the rendered point cloud as the effective vertices.

To avoid missing effective vertices and further improve the robustness of the reconstructed adversarial object, we render the original adversarial object at various distances and collect the effective vertices for each distance. More specifically, we divide the interval where we expect the adversarial object to be effective into N equal segments, determining N positions at the center of these segments. We then render the adversarial object at these N positions. At each position, we slightly move the adversarial object to generate a series of point clouds. From these, we select the point cloud that has the minimum number of points while still capable of fooling the model. By leveraging the correspondence between the points of the selected point clouds and the vertices of the object, we can identify effective vertices at each position. Then, we take the union of effective vertices at all positions to obtain a set of effective vertices, denoted as $V = \left\{v_1^{'}, v_2^{'}, ..., v_n^{'}\right\}$.

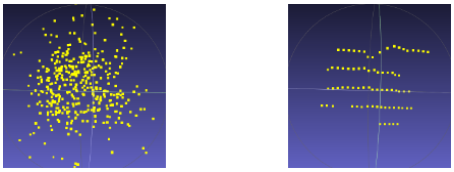

Figure 7: From all possible vertices (left), we identify the effective adversarial ones (right).

5.2 Constructing Adversarial Faces

With the set of effective vertices V, we can construct the adversarial surface to enhance the robustness of the adversarial object in the physical world. The question is, how to create a reasonable and effective adversarial surface?

Assume an ideal position for the LiDAR, where all laser beams precisely hit the effective vertices, thereby capturing an effective adversarial point cloud. However, when there is relative motion between the LiDAR and the adversarial object, such as the LiDAR approaching the object, the hit points of laser beams deviate from the effective vertices and shift towards their adjacent points, resulting in a new LiDAR reflection signal with altered coordinates. Notably, when the effective vertex is located on sharp protrusions, the disparity between its coordinates and those of its adjacent point becomes substantial. Conversely, if the effective vertex is positioned on flat surfaces, the discrepancies between its coordinates and those of its adjacent point are minimal. Therefore, we create a flat region centered around each effective vertex as an adversarial surface to reduce the discrepancies between the captured point cloud and the desired point cloud, thus improving the physical robustness.

Formally, we provide a definition for the adversarial face f_{adv}, which is represented as a rectangle. We consider f_{adv} as a rectangle because the LiDAR scans an object in rows, and using a rectangle maintains consistency in both horizontal

and vertical directions. The rectangle can be represented by its four vertices, as shown in Equation 1.

$$\begin{cases} f_{adv} & = & [v_{n1}^{'}, v_{n2}^{'}, v_{n3}^{'}, v_{n4}^{'}] \\ v_{n1}^{'} & = & t(v_n^{'}, l_1, d_1) \\ v_{n2}^{'} & = & t(v_n^{'}, l_2, d_2) \\ v_{n3}^{'} & = & t(v_n^{'}, l_3, d_3) \\ v_{n4}^{'} & = & t(v_n^{'}, l_4, d_4) \end{cases} \quad (1)$$

Here, $v_n^{'}$ denotes the original adversarial vertex, and the function $t(\cdot)$ represents the translation operation. $v_{n1}^{'}$, $v_{n2}^{'}$, $v_{n3}^{'}$ and $v_{n4}^{'}$ are the new vertices generated by translating $v_n^{'}$ in the direction of d_1, d_2, d_3 and d_4 by distances l_1, l_2, l_3 and l_4, respectively.

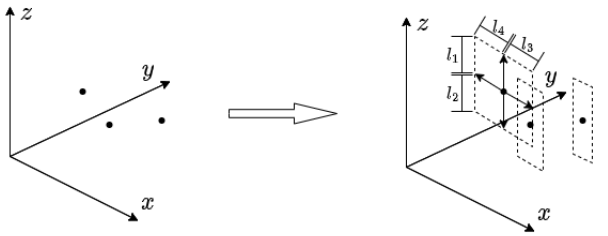

Figure 8: Translate the effective vertices to create adversarial faces.

We adopt a right-hand coordinate system, where the victim vehicle moves from the negative direction of the Y-axis to the positive direction, and the laser beams are emitted from the same negative direction of the Y-axis. As depicted in Figure 8, all effective vertices are expanded to form faces. These faces are perpendicular to the XOZ plane and oriented towards the Y-direction. In this scenario, $d_1 = [0,0,1]$, $d_2 = [0,0,-1]$, $d_3 = [1,0,0]$ and $d_4 = [-1,0,0]$. Here, l_1, l_2, l_3 and l_4 represent the translation magnitudes in the d_1, d_2, d_3 and d_4 directions, respectively. In this case, l_1 and l_2 are the same for all vertices, considering that the height of all adversarial faces should be equal. l_3 and l_4 are adjusted based on individual vertices to ensure that adjacent adversarial faces do not block each other.

5.3 Constructing Adversarial Object

In the previous step, we generate a series of adversarial faces corresponding to the effective vertices, all sharing the same orientation. Subsequently, these faces need to be connected to form a single continuous surface. As depicted in the right half of Figure 8, the adversarial faces are initially scattered. To achieve a continuous surface, we connect the two closest pairs of vertices between each pair of adjacent faces, creating new quadrilateral faces. The result is illustrated in the left half of Figure 9.

Next, we proceed to construct a 3D object using this surface. We duplicate the surface and then translate the duplicated surface along the position direction of the Y-axis by a distance d. The original surface and the duplicated surface are then connected using quadrilateral faces, which can be constructed by connecting the closest vertices on the outer edges of both surfaces. The value of d determines the thickness of the newly created 3D object and can be minimized based on stealthiness requirements.

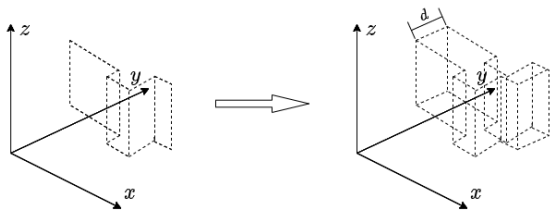

Figure 9: Transform adversarial faces into a 3D object.

As shown in the right half of Figure 7, the effective vertices are arranged in rows. Figure 9 shows the reconstruction result of one of these rows. We can reconstruct each row and stack them like layers of a cake to obtain the reconstructed adversarial object. Furthermore, we can modify the back side of the adversarial object (the side not facing the LiDAR) to enhance its practicality in the physical world. For example, we can flatten the back side to facilitate its physical production or attach hooks to it, allowing the object to be hung somewhere. These modifications are feasible because the LiDAR sensor does not capture the back of the object, ensuring that the robustness of the adversarial object remains unaffected.

5.4 Stealthiness Enhancement

To enhance the stealthiness of the adversarial object, we can adjust the height of each layer. While generating adversarial faces in Section 5.1, we establish the translation magnitude as the margin between adjacent vertices. However, as shown in the right half of Figure 7, there is a noticeable gap between two vertically adjacent vertices, allowing us to moderately reduce the value of l_1 and l_2 to improve the stealthiness of the adversarial object.

More specifically, as depicted in Figure 10, when the LiDAR approaches or moves away from the adversarial object, the hit points of the laser beams move up and down along its surface. Consequently, by maximizing l_1 and l_2 (as depicted in Figure 8) up to the full margin to enlarge the LiDAR reflection surface, we can enhance the robustness of the adversarial object. Conversely, reducing l_1 and l_2 to decrease the volume of the adversarial object improves its stealthiness.

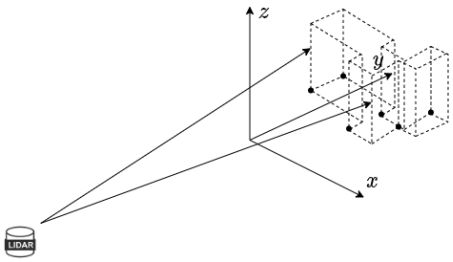

Figure 10: The hit points of the laser beams move up and down along the surface of the adversarial object.

6 Evaluations in Simulated World

In this section, we apply our method to the adversarial object generated by Tu's method [41]. The process of generating the adversarial object is discussed in detail in Section 6.2. Our method can also be employed with other attack methods, provided that an effective adversarial object is available. We evaluate the effectiveness of our method in enhancing the survival rate of digital adversarial objects, and the attack success rate in dynamic scenarios with and without deflection angles. All experiments presented in this section are conducted within a simulator environment. Furthermore, the adversarial ornaments in this section solely utilize their shape to convey adversarial information. For physical experiments, please refer to Section 8. Additionally, we assess the printability of adversarial objects both before and after applying our method, please refer to Section A.3.

6.1 Concepts in Evaluations

Before presenting our experiments, we clarify several key concepts used throughout this section:

Original adversarial ornaments: These are the adversarial ornaments directly transferred from the digital adversarial object.

Reconstructed adversarial ornaments: These adversarial ornaments are reconstructed based on the Original adversarial object.

Victim vehicle: This is the car equipped with a LiDAR and the corresponding LiDAR-based perception model, which we aim to deceive using adversarial objects.

Target vehicle: This is the car attached with an adversarial object, and we intend to make it disappear from the victim vehicle's detection results.

SVL Simulator: The SVL Simulator is an open-source autonomous vehicle simulator developed by LG Electronics America R&D Lab. Researchers can create their own maps and control cars within the simulator to obtain sensor data closely resembling real-world conditions. Numerous papers have successfully conducted their experiments using the SVL Simulator [12–14, 21]. Our experiments mentioned in this section are also performed in the SVL Simulator environment to acquire a relatively large amount of experimental data that closely approximates real-world conditions. Additionally, we showcase the results of our physical experiments in Section 8.

Apollo: Apollo is an open-source autonomous driving framework capable of receiving data from sensors and detecting vehicles, pedestrians, and other obstacles within the sensor data. We used Apollo 7.0 to conduct our dynamic experiment, as described in Section 6.4 and 6.5.

6.2 Evaluation Setup

Attack scenarios. We consider an attacker who generates an adversarial object to deceive the LiDAR-based perception model of an autonomously driving vehicle. The generated adversarial object is attached to the target vehicle as an ornament, causing the model to fail in detecting the target vehicle. The LiDAR and the target vehicle may be in a stationary position at varying distances or in motion with different angles.

Generate the adversarial object. We basically follow Tu's method [41] to construct the original adversarial ornament. After the adversarial optimization process, we obtain an effective adversarial object with a complex surface, where the coordinates of its vertices carry the adversarial information. However, LiDAR cannot accurately capture such objects, which can cause discrepancies between the LiDAR echo signals and the physical object, thereby reducing its adversarial robustness, as explained in Section 3.

Reconstruct the adversarial object. To improve the robustness of the generated adversarial object in the physical world, we apply our method described in Section 5. In practice, we find that the points of the LiDAR point cloud are relatively dense in the horizontal direction. For instance, considering the "00000.bin" file in the KiTTi dataset. At a distance of about 5.95m, the horizontal gap between adjacent points measures approximately 0.17m, while the vertical gap is around 0.49m. Therefore we set l_3 and l_4 (the horizontal translation magnitude) to zero. Under such conditions, adversarial faces are reduced to adversarial edges. Additionally, we set $l_1 + l_2$ to 50% of the margin between adjacent vertical vertices to increase the stealthiness of the adversarial object.

6.3 Efficacy in Static Attack

In this experiment, we select two widely used models: PointPillars [10] and PointRCNN [36], as our target models. These two models represent the two mainstream approaches in the field of object detection for autonomous driving systems, namely voxel-based object detection and point-based object detection. Notably, PointPillars has been adopted in Apollo and Autoware, which are among the world's leading open-source software projects for autonomous driving.

Experimental Setup. We generate 200 adversarial ornaments using Tu's method [41] for both PointPillars and PointRCNN. All of these adversarial ornaments are tested to be effective in the digital world at a distance of 15m, which is the braking distance [8] when the vehicle speed is 50 km/h, a common speed on city roads. The confidence thresholds for each model are set to their default values, i.e., 0.1 for PointPillars [50] and 0.3 for PointRCNN [35].

Subsequently, we reconstruct these adversarial ornaments with our methods and place the reconstructed ornaments at the back of the target vehicle in the SVL Simulator's map. Then, we control the victim vehicle equipped with a LiDAR to record the LiDAR signal points of the adversarial ornaments and the target vehicle from different distances, as shown in Figure 11. Finally, we export the recorded point cloud data and feed them into PointPillars or PointRCNN and observe how many adversarial ornaments can still deceive these two models.

Figure 11: Screenshot of the target and the victim vehicle. The target vehicle with an adversarial ornament is positioned at the front, and the victim vehicle is positioned at the back.

Evaluation Metrics. Considering practical application scenarios, we adopt the *survival rate* as the evaluation metric for our proposed method. The *survival rate* refers to the proportion of the 200 adversarial ornaments that maintain their effectiveness, successfully concealing the car at the specified distance, when transitioning to a more realistic environment. A high survival rate indicates that a significant portion of the adversarial ornaments maintain their effectiveness in a more realistic environment. This, in turn, enables attackers to execute large-scale attacks more easily and efficiently.

Evaluation Results. The results of this experiment are presented in Table 1. The experiment demonstrates that our method enhances the survival rate of adversarial ornaments by 38% for PointPillars and 22% for PointRCNN when they are placed at their designed distance, i.e., 15m, in the physical world. This distance is equivalent to the braking distance [8] when a vehicle's speed is 50 km/h, a common speed on city roads. Failing to detect the target car within a 15m distance leaves the victim vehicle with insufficient time to slow down and avoid a collision. In other words, the survival rate within this distance is positively correlated with the crashing rate. Furthermore, our method also improves the survival rate of the adversarial ornaments at other distances.

This experiment demonstrates the effectiveness of our proposed method across models utilizing different detection principles, further solidifying its applicability and value in enhancing the robustness of adversarial objects.

Methods	Model	Distance			
		10m	14m	15m	16m
Original	PointPillars	46.5%	45.5%	**46.0%**	34.5%
	PointRCNN	38.5%	55.0%	**58.0%**	44.5%
Reconstruction	PointPillars	51.5%	65.0%	**84.0%**	41.0%
	PointRCNN	41%	80.0%	**80.0%**	50.5%

Table 1: Percentage of adversarial objects that are still effective in the simulator environment with and without our method.

6.4 Efficacy in Dynamic Attacks

In this experiment, we aim to evaluate the effectiveness of our methods in scenarios where there is relative motion between the victim and target vehicles. The evaluation is conducted using a combination of the SVL Simulator and Apollo.

Experimental Setup. First, we establish a connection between the SVL Simulator and Apollo 7.0. Next, we position the victim vehicle at various distances behind the target vehicle and move the victim vehicle towards the target vehicle. Finally, we analyze Apollo's detection results. We record frames every 0.13m and calculate the number of frames in which the target vehicle is detected and the number of frames in which it is not detected as the victim vehicle moves towards the target vehicle. This experiment is conducted only on PointPillars, as Apollo does not integrate PointRCNN. We test three distinct starting distances: 10m, 15m, and 20m and the ending distance is 2m.

Evaluation Metrics. To assess the effectiveness and stealthiness of our method, we employ the *attack success rate* and *projection area* as evaluation metrics, respectively. The *attack success rate* is calculated as the ratio of the number of missing detection frames to the total number of frames. The *projection area* represents the size of the adversarial ornament as viewed by the driver of the victim vehicle, and is determined by projecting the ornament onto the XOZ plane.

Evaluation Results. The results of this experiment are presented in Table 2. Our method demonstrates notable improvements in the attack success rate of adversarial ornaments at different distances. On average, the attack success rate increases by 38.64%, 21.8%, and 15.7% at each distance, respectively. Additionally, our method significantly reduces the

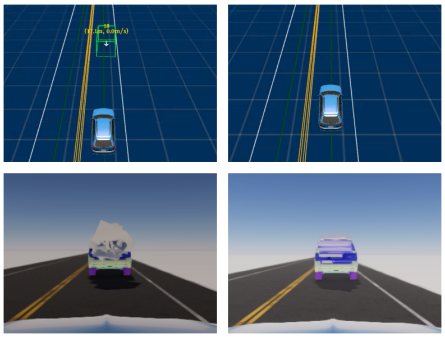

Figure 12: Visualization of Apollo's detection results: The model successfully detects the target (top left) equipped with the original adversarial ornament (bottom left), yet fails to detect the target vehicle (top right) equipped with the reconstructed adversarial ornament (bottom right).

Methods	Distance (m)			Projection (m^2)
	10m→2m	15m→2m	20m→2m	
Original	54.7%	66.4%	71.9%	2.79
Reconstruction	**100.0%**	**85.0%**	**82.2%**	**0.78** (↓72.04%)
Original	62.7%	69.0%	71.1%	2.59
Reconstruction	**97.3%**	**88.5%**	**88.9%**	**0.83** (↓67.95%)
Original	68.0%	68.1%	71.9%	2.83
Reconstruction	**92.0%**	**89.4%**	**91.1%**	**1.00** (↓64.66%)
Original	42.7%	58.4%	65.2%	3.32
Reconstruction	**100.0%**	**89.4%**	**85.3%**	**1.02** (↓69.28%)
Original	68.0%	69.9%	72.6%	2.53
Reconstruction	**100.0%**	**88.5%**	**83.7%**	**0.91** (↓64.03%)

Table 2: The attack success rates in different distances with and without our method.

projection area by 67.59% on average, thereby enhancing the stealthiness of the adversarial object. Notably, if we do not apply our method but restrict the projection area during the adversarial optimization stage, the generated adversarial ornaments cannot deceive the perception model, Consequently, achieving adversarial ornaments with such a level of stealthiness is challenging without employing our methods.

Figure 12 shows one of the screenshots during the experiment. We observe that even though the original adversarial ornament is much larger than the reconstructed one and appears to obstruct more details of the target vehicle , it fails to deceive the perception model while the reconstructed one succeeds. This is because the point cloud of the original adversarial ornament captured by the LiDAR is quite different from the effective adversarial point cloud and does not carry enough adversarial information. The captured point cloud of the reconstructed adversarial ornament, the original adversarial ornament, and the desired adversarial point cloud, are shown in our website. We observe that the shape and contour of the LiDAR signal points of the reconstructed adversarial ornament are well preserved during motion and are similar to the desired adversarial point cloud. In contrast, the LiDAR signal points of the original adversarial ornament are not.

As described in Section 5, LiDAR captures an object's information through multiple laser beams rather than providing a panoramic view like a camera. The hit points of the laser beams change as the distance between the LiDAR and the object varies, resulting in variations in the generated point cloud. The reconstructed adversarial ornament maintains a simple and smooth structure, whereas the original adversarial ornament does not. As a result, our reconstructed adversarial ornament can preserve the relative stability of the LiDAR reflection points when the distance between the object and the LiDAR changes, while the original adversarial ornament cannot. Our method demonstrates a consistent attack success rate across various distances in this experiment, indicating its robustness in different scenarios.

6.5 Efficacy in Angle Robustness

In this experiment, we aim to evaluate the effectiveness of our method in different angles. The evaluation is also conducted using a combination of the SVL Simulator and Apollo.

Experiment Setup. To simulate real driving behavior, we did not fix the relative angles of the target and victim vehicle. Instead, we control the victim vehicle to approach the target vehicle in the adjacent lane, as shown in Figure 13. The angle between the victim and the target vehicle is dynamic during this process, which is more realistic. All other settings were similar to those in Section 6.4.

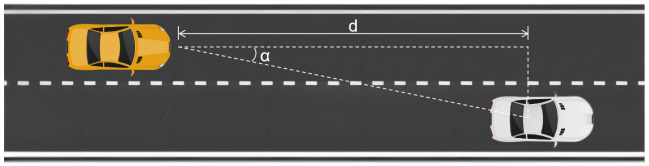

Figure 13: The angle (α) between the target vehicle (white) and the victim vehicle (yellow) dynamically changes as the distance (d) varies.

Evaluation Results. The results of this experiment are presented in Table 3. Compared to moving straightforwardly toward the target vehicle, approaching the target vehicle in the adjacent lane and deceiving the perception model is a more challenging scenario, particularly when the victim is close to the target vehicle, which implies a relatively large angle between the victim and the target vehicle. Nevertheless, our method demonstrates notable improvements in the attack suc-

cess rate of adversarial ornaments at different distances. On average, the attack success rate increases by 23.72%, 23.86%, and 23.70% at each distance, respectively. Additionally, restricting the projection area during the adversarial optimization stage still results in ineffective adversarial ornaments. This experiment showcases the robustness of our method against varying angles.

Methods	Distance (m)			Projection (m^2)
	10m (19.29°)	15m (13.13°)	20m (9.93°)	
Original	54.7%	67.3%	68.1%	2.88
Reconstruction	**80.0%**	**86.7%**	**88.9%**	**0.86** (↓70.14%)
Original	62.7%	54.9%	60.7%	3.50
Reconstruction	**82.7%**	**86.7%**	**88.9%**	**0.76** (↓78.29%)
Original	62.7%	54.0%	56.3%	3.26
Reconstruction	**89.3%**	**85.8%**	**88.1%**	**0.87** (↓73.31%)
Original	60.0%	55.8%	58.5%	3.22
Reconstruction	**82.7%**	**82.3%**	**85.1%**	**0.96** (↓70.19%)
Original	65.3%	62.8%	63.7%	2.38
Reconstruction	**89.3%**	**72.6%**	**74.8%**	**0.71** (↓70.17%)

Table 3: The attack success rates in different angles with and without our method

6.6 Efficacy Compared with Other Methods

In this experiment, we compare our proposed method with an existing attack method, denoted as **AdvLo** [49], which aims to conceal the target vehicle from the LiDAR-based perception model's detection results. The method is proposed by Miao et al., where they employ drones to fill several calculated positions around the vehicle to attack the model.

Experiment Setup. For AdvLo, we utilize the data mentioned in their paper, where they achieve a 74% success rate against PointPillars in 100 examples. Specifically, they use the algorithm AdvLo to generate adversarial locations for each example and 74% of all examples are attacked successfully. In comparison, we generate 100 adversarial ornaments following the steps outlined in Section 6.2 and reconstruct these ornaments using our method. The success rate of the adversarial ornaments before reconstruction is denoted as "Original" while the success rate after reconstruction is denoted as "Reconstruction" Since AdvLo does not specify the distance used in their experiment, we considered their 74% success rate as their best result for PointPillars and compared it with our best results.

Evaluation Results. The results of this experiment are presented in Table 4. The effective rates for the three conditions are 45.0%, 83.0%, and 74.0%, respectively. We can see that before reconstruction, the original adversarial ornament does not achieve a competitive result. However, after reconstruction, the reconstructed adversarial ornament outperforms AdvLo by 9.0%. Additionally, Our approach does not need to hover several drones exactly at some specified locations above the target vehicle, like what AdvLo does, which can facilitate the attacker when the target vehicle is in motion.

Methods	Success Rate
Original	45.0%
Reconstruction	83.0%
AdvLo [49]	74.0%

Table 4: A comparison of the attack success rate of our method and AdvLo [49].

7 Evaluations on Fusion Attack

In this experiment, we investigate the potential of employing the reconstructed adversarial object to conduct a fusion attack against both LiDAR-based and camera-based perception models simultaneously. As previously mentioned, our method generates adversarial objects with flat surfaces, which are subsequently colored to convey adversarial information to camera-based perception models. To summarize, we utilize the shape of these adversarial ornaments to carry adversarial information against LiDAR-based perception models, while their color is employed to transmit adversarial information against camera-based perception models.

Experiment Setup. The fusion attack evaluations involve both camera-based and LiDAR-based perception models. The LiDAR-based perception model used remains PointPillars, similar to the dynamic attack evaluations. As for the camera-based perception model, we choose SMOKE [31], a state-of-the-art camera-based perception model. SMOKE is also the default camera-based perception model used in Apollo 7.0. To convey adversarial information against SMOKE, we adopt an existing method [40] to optimize the texture of the adversarial ornament, resulting in a colorful adversarial ornament as shown in Figure 14. We then conduct experiments in the SVL Simulator, maneuvering the victim vehicle to approach the target vehicle. During this process, we record the number of frames in which Apollo fails to detect the target vehicle. Similar to Section 6.4, we use the *attack success rate* to measure the effectiveness of the colorful adversarial ornament. The attack success rate is calculated as the ratio of the number of missing detection frames to the total number of frames.

Evaluation Results. We observe that the flat surface generated by our method effectively serves as a carrier of adversarial information against camera-based models. During

Figure 14: The white adversarial ornament does not carry adversarial information against SMOKE, while the colorful adversarial ornament is designed to convey adversarial information specifically targeting SMOKE.

Lidar's movement from 20m away towards the target vehicle, we achieved a 75.6% attack success rate against the fusion perception module.

8 Evaluations in Real World

In this section, we evaluate the attack's effectiveness and feasibility in the physical world. The attack aims to hide the target vehicle from the LiDAR detection system.

8.1 Nighttime Experiments

Experiment Setup. We utilize the RS-LiDAR-16 to collect point clouds during the experiment. The adversarial object used in this experiment is cut from cardboard without the involvement of a 3D printer. The target model is PointPillars and the target vehicle is a Toyota Levin. The adversarial ornament is placed on the roof of the target vehicle as shown in Figure 15. We put the LiDAR around 12m away from the target vehicle and move it towards the target vehicle slowly. We record the point cloud data captured during this process and feed them to the model to calculate the attack success rate. Under the same condition, we also design a comparison experiment, i.e., placing a benign cardboard on the roof of the car. More detailed pictures can be found in our website.

Figure 15: The adversarial ornament is placed on the roof of the target vehicle.

Evaluation Results. Our method achieves a 71.4% attack success rate during this process. In contrast, when a benign cardboard is attached, the LiDAR consistently detects the car. Figure 16 illustrates the scan results of the surrounding environment by the LiDAR centered on the victim vehicle. The detection results of the model are highlighted with black bounding boxes. We observe that the model successfully detects the target car with the benign ornament (left), but fails to detect the same target car with the adversarial ornament (right). It is worth noting that the total cost of physically creating an adversarial ornament reconstructed by our method is less than $15. This implies that adversarial ornaments can be produced quickly and inexpensively in large quantities, posing a significant threat in the real world.

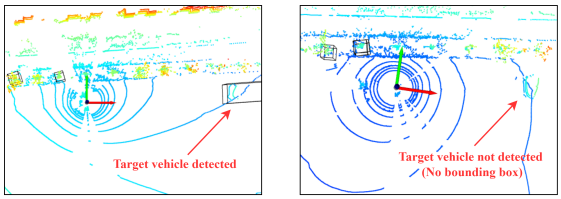

Figure 16: The detection results of the target vehicle with the benign (left) and the adversarial ornament (right).

8.2 Daytime Experiments

To comprehensively evaluate the effectiveness of our method and enhance the diversity of experimental settings, we conduct an additional real-world experiment during daylight hours. In this experiment, we target a new vehicle and place the adversarial ornament in new locations.

Experiment Setup: For this experiment, we select the Aion S as the target vehicle. The reconstructed adversarial ornaments are positioned on the trunk lid, as depicted in Figure 17. We continue to employ the RS-LiDAR-16 for collecting point clouds. All other settings remain identical to those described in Section 8.1. Extra pictures can be found in our website.

Figure 17: The adversarial ornament is placed on the truck lid of the target vehicle.

Evaluation Results. Our method achieves a success rate of 80.8% during motion from 12m to 3m right behind the target

vehicle. This value is slightly higher than that of the nighttime experiment. One possible contributing factor is the use of a greater layer height for reconstructing adversarial ornaments (about 18cm per layer in this experiment compared to about 14cm per layer in the nighttime experiment), which highlights the trade-off between robustness and stealthiness once again.

8.3 Analysis

It has been observed that there is a discrepancy between the attack success rate observed in simulated experiments and real-world experiments. We consider that the following factors contribute to this gap.

The position of adversarial ornaments. When generating and placing adversarial ornaments in the digital realm, these ornaments can be affixed to any part of the target vehicle to pursue the best performance, disregarding gravitational forces and other physical constraints. However, in the real world, adhesive tape is often insufficient for this purpose, necessitating compromises in the placement of these ornaments. Consequently, we are often unable to place the ornaments in the most optimal locations, which compromises their performance. This issue can be mitigated through the use of improved adhesive solutions.

The complexity of the environment. On the one hand, the real-world environment is significantly larger than the simulated environment (about 450,000 LiDAR reflection points versus about 7,000 LiDAR reflection points). Influencing a considerably larger environment presents greater challenges. On the other hand, the simulated environment remains relatively static, with no pedestrian or other moving vehicles present. In contrast, the real-world environment is dynamic, with passing cars and people. These factors have the potential to disrupt the effectiveness of adversarial ornaments and influence the experimental outcomes. This issue can be mitigated through the use of more realistic simulators.

The resolution of the LiDAR. The simulator employs a 64-line LiDAR, whereas the real-world LiDAR consists of 16 lines. While our method is effective for LiDARs with varying resolutions, there may be some performance differences.

9 The Effect of Reconstructed Adversarial Ornaments' Volume

In this section, we analyze the effect of the volume of reconstructed adversarial ornaments. Intuitively, increasing the volume is expected to enhance robustness while diminishing stealthiness, and vice versa. We replicate the experiments conducted in Section 6.4 and Section 6.5 to validate this intuition.

Experiment Setup. In this experiment, we utilize the same reconstructed adversarial ornaments as those employed in Section 6.4 and 6.5. Subsequently, we adjust the volume of these reconstructed adversarial ornaments by modifying the height, represented by the values of $l_1 + l_2$, for each layer. Specifically, we set the height of each layer of the reconstructed adversarial ornaments to a different percentage of the margin between the layers of the point clouds, as illustrated in Figure 18. All other configurations remain consistent with those described in Section 6.4 and 6.5.

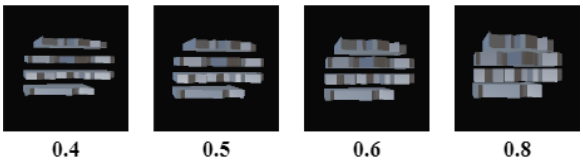

Figure 18: Reconstructed adversarial ornaments with different layer heights.

Evaluation Metrics: Two metrics, namely the *attack success rate* and the *stealthiness*, are utilized in this experiment. The definition of the *attack success rate* remains the same as the one provided in Section 6.4 and 6.5, (i.e. the ratio of the number of missing detection frames to the total number of frames). As for the *stealthiness*, it is defined as $1 - R_{area}$, where R_{area} represents the area ratio between the back view of the adversarial ornament and the back view of the target vehicle.

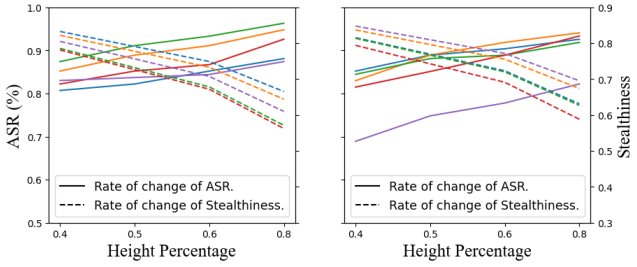

Figure 19: Trade-off between robustness and stealthiness in dynamic attacks (left) and angle attacks (right).

Evaluation Results: The results of this experiment are presented in Figure 19. Each color represents a distinct reconstructed adversarial ornament. The solid line represents the change in its attack success rate during the motion from 20m to 2m with respect to the height percentage, while the dotted line represents the change in its stealthiness with respect to the height percentage. It can be observed from both line charts that as stealthiness decreases, the attack success rate increases. In other words, there exists a trade-off between robustness and stealthiness. Decreasing stealthiness can effectively increase robustness, and vice versa. However, even with the highest observed stealthiness (83.27%, indicating that the back view of the reconstructed adversarial ornament accounts for only 16.73% of the back view of the vehicle), the ornament still achieves a dynamic attack success rate of 80.7%. Similarly, for angle attacks, the success rate with the highest stealthiness is 68.9%, which is also considered acceptable.

10 Discussion

D1: What are the differences in robustness loss of adversarial objects for LiDAR- and camera-based perception models?

Discrepancies between the captured information and the physical object are among the primary causes of robustness loss in both situations. However, the sources of these discrepancies in capture processes differ significantly between cameras and LiDAR. For cameras, discrepancies stem from occlusions, lighting conditions, printing inaccuracies, and environmental factors like weather and glare. These factors make it difficult to capture accurate color and shape information of adversarial patches, thereby compromising the effectiveness of these patches. In contrast, LiDAR-based models are less affected by environmental factors, but discrepancies occur due to missing details caused by the sparse laser beams.

Based on the above analysis, our proposed method reconstructs the surface of adversarial objects to help LiDAR capture more adversarial details, thus preserving the attack effectiveness of adversarial objects in the physical world.

D2: What are the differences between the existing robustness improvement methods and our method?

Existing robustness improvement methods aim to enhance the generalization of adversarial objects by applying transformations to the digital object to tolerate the discrepancies between the digital and the captured object, as shown in the left half of Figure 20. In contrast, our approach focuses on reducing the discrepancies between the digital and captured object by reconstructing the surface of the original object to fit the sparse laser beams, as shown in the right half of Figure 20. Our method improves the robustness of the adversarial object without interfering with the adversarial optimization stage, allowing it to be applied to various attack methods as long as a digital adversarial object is available.

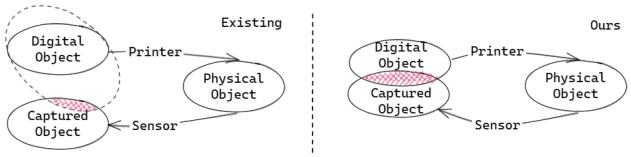

Figure 20: The red part represents the attack effectiveness of the captured object. Existing methods enhance the effectiveness by improving the generalization of the digital object, whereas our method focuses on reducing the disparities between the digital and captured objects.

D3: How do real humans perceive the stealthiness of reconstructed adversarial ornaments?

To assess the stealthiness of our physical reconstructed ornaments, we have devised a questionnaire with pictures like the ones in Section 8.2 attached. The questionnaire encompasses four aspects: 1) Does the cardboard attached to the ornament strike you as unusual or abrupt; 2) Do you think the cardboard ornament can lead to a car crash; 3) Do you perceive this ornament as a personalized decoration chosen by the car owner; 4) Would you feel compelled to notify the driver if their vehicle displayed such an ornament. We also ask participants to self-evaluate their knowledge of AI security and adversarial examples.

In total, we collect 25 questionnaires, with 28% of participants being unfamiliar with adversarial objects, 28% having some familiarity, and 44% being very familiar with adversarial objects. We observe that 84% of all participants do not think our ornaments have the potential to cause a car crash, even among those who are very familiar with adversarial objects. Furthermore, 72% of participants regard the ornaments as personalized decorations chosen by the car owner, while only 40% think it is abrupt. Through further communication with these individuals, we found that most of them believe the apparent abruptness is due to the bare cardboard being too ugly, and they think we can alleviate people's suspicion by painting and drawing on the cardboard. Finally, only 32% feel it necessary to remind the car driver. The statistical data mentioned above indicate that the majority of individuals do not exhibit excessive concern regarding our reconstructed adversarial ornaments. This finding suggests that our ornaments possess stealthiness and practical viability. Our studies receive approval from the IRB of our affiliations.

D4: What are the limitations of our method?

On one hand, our method aims to enable LiDAR to accurately capture the adversarial perturbations of the adversarial object. In other words, our goal is to ensure that the captured point clouds resemble the desired point clouds. However, the effectiveness of the desired point cloud is contingent upon other attack methods and may be compromised by certain defense methods. On the other hand, our current method only involves one LiDAR and its performance may be compromised in scenarios with multiple LiDARs. We will further investigate this aspect in future work. Additionally, the reconstructed adversarial object can only hide one vehicle at a time, and it is visible, which may raise suspicion among drivers and other pedestrians — these are also limitations.

11 Conclusion

This paper presents the first endeavor to utilize LiDAR principles to enhance the robustness of adversarial objects. We reconstruct the surface of the original adversarial object to fit the sparse laser beams, thereby reducing errors during the LiDAR capturing process and improving the robustness of the adversarial object. Experimental results demonstrate significant achievements in both simulated environments and real-world scenarios. Furthermore, our method does not rely on expensive 3D printing technology and can be implemented using simple manual cutting of affordable materials like cardboard.

Acknowledgments

The IIE authors are supported in part by NSFC (92270204, 62302498), Youth Innovation Promotion Association CAS and a research grant from Huawei.

References

[1] ApolloAuto/apollo: An open autonomous driving platform. https://github.com/ApolloAuto/apollo.

[2] Autonomous Driving Vehicle System Using LiDAR Sensor | SpringerLink. https://link.springer.com/chapter/10.1007/978-981-16-7610-9_25.

[3] Autowarefoundation/autoware: Autoware - the world's leading open-source software project for autonomous driving. https://github.com/autowarefoundation/autoware.

[4] Formlabs Software | Formlabs. https://formlabs.com/software/.

[5] JG MAKER Industrial SLA 3D Printer JG-A600. https://www.amazon.com/JG-MAKER-Industrial-23-6x22-8x14-9in-Accuracy/dp/B0B6ZYFGDW.

[6] LiDAR sensor | Autonomous vehicle sensors | Valeo. https://www.valeo.com/en/valeo-scala-lidar/.

[7] Ray Casting and Rendering. https://ocw.mit.edu/courses/6-837-computer-graphics-fall-2012/resources/mit6_837f12_lec11/.

[8] Vehicle Stopping Distance Calculator. http://www.csgnetwork.com/stopdistcalc.html.

[9] Mazen Abdelfattah, Kaiwen Yuan, Z. Jane Wang, and Rabab Ward. Towards universal physical attacks on cascaded camera-lidar 3d object detection models. In 2021 IEEE International Conference on Image Processing (ICIP), pages 3592–3596, 2021.

[10] Holger Caesar Alex H. Lang, Sourabh Vora. Pointpillars: Fast encoders for object detection from point clouds. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, pages 12697–12705, 2019.

[11] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In Proceedings of the 35th International Conference on Machine Learning, volume 80, pages 284–293, 2018.

[12] Yulong Cao, S. Bhupathiraju, Pirouz Naghavi, Takeshi Sugawara, Z. Mao, and Sara Rampazzi. You can't see me: Physical removal attacks on LiDAR-based autonomous vehicles driving frameworks. In 32nd USENIX Security Symposium, pages 2993–3010, Anaheim, CA, 2023.

[13] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. 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), pages 176–194, 2021.

[14] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z. Morley Mao. Adversarial Sensor Attack on LiDAR-based Perception in Autonomous Driving. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, pages 2267–2281, 2019.

[15] Yulong Cao, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Mingyan D. Liu, and Bo Li. Adversarial objects against lidar-based autonomous driving systems. http://arxiv.org/abs/1907.05418, 2019.

[16] Shang-Tse Chen, Cory Cornelius, Jason Martin, and Duen Horng (Polo) Chau. Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector. In Michele Berlingerio, Francesco Bonchi, Thomas Gärtner, Neil Hurley, and Georgiana Ifrim, editors, Machine Learning and Knowledge Discovery in Databases, pages 52–68, Cham, 2019.

[17] Zhiyuan Cheng, James Liang, Hongjun Choi, Guanhong Tao, Zhiwen Cao, Dongfang Liu, and Xiangyu Zhang. Physical attack on monocular depth estimation with optimal adversarial patches. In Computer Vision – ECCV 2022, 2022.

[18] Alesia Chernikova, Alina Oprea, Cristina Nita-Rotaru, and BaekGyu Kim. Are self-driving cars secure? evasion attacks against deep neural networks for steering angle prediction. In 2019 IEEE Security and Privacy Workshops (SPW), 2019.

[19] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[20] Kuofeng Gao, Jiawang Bai, Baoyuan Wu, Mengxi Ya, and Shu-Tao Xia. Imperceptible and robust backdoor attack in 3d point cloud. IEEE Transactions on Information Forensics and Security, 19:1267–1282, 2024.

[21] R. S. Hallyburton, Yupei Liu, Yulong Cao, Z. Mao, and M. Pajic. Security analysis of Camera-LiDAR fusion against Black-Box attacks on autonomous vehicles. In 31st USENIX Security Symposium (USENIX Security 22), pages 1903–1920, Boston, MA, 2022.

[22] Zhongyuan Hau, Kenneth T. Co, Soteris Demetriou, and Emil C. Lupu. Object removal attacks on lidar-based 3d object detectors. Proceedings Third International Workshop on Automotive and Autonomous Vehicle Security, 2021.

[23] Ching-Pai Hsu, Boda Li, Braulio Solano-Rivas, Amar R. Gohil, Pak Hung Chan, Andrew D. Moore, and Valentina Donzella. A Review and Perspective on Optical Phased Array for Automotive LiDAR. IEEE Journal of Selected Topics in Quantum Electronics, 27(1):1–16, 2021.

[24] Chengyin Hu, Weiwen Shi, Ling Tian, and Wen Li. Adversarial neon beam: A light-based physical attack to dnns. Computer Vision and Image Understanding, 238:103877, 2024.

[25] Steve T.K. Jan, Joseph Messou, Yen-Chen Lin, Jia-Bin Huang, and Gang Wang. Connecting the digital and physical world: Improving the robustness of adversarial attacks. Proceedings of the AAAI Conference on Artificial Intelligence, 33(01):962–969, 2019.

[26] Zizhi Jin, Xiaoyu Ji, Yushi Cheng, Bo Yang, Chen Yan, and Wenyuan Xu. Pla-lidar: Physical laser attacks against lidar-based 3d object detection in autonomous vehicle. In 2023 IEEE Symposium on Security and Privacy (SP), 2023.

[27] Pengfei Jing, Qiyi Tang, Yue Du, Lei Xue, Xiapu Luo, Ting Wang, Sen Nie, and Shi Wu. Too good to be safe: Tricking lane detection in autonomous driving with crafted perturbations, 2021.

[28] Sihyeon Kim, Sanghyeok Lee, Dasol Hwang, Jaewon Lee, Seong Jae Hwang, and Hyunwoo J. Kim. Point cloud augmentation with weighted local transformations. In 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021.

[29] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. PhysGAN: Generating Physical-World-Resilient Adversarial Examples for Autonomous Driving. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 14242–14251, 2020.

[30] Juncheng Li, Frank R. Schmidt, and J. Zico Kolter. Adversarial camera stickers: A physical camera-based attack on deep learning systems. In Proceedings of the 36th International Conference on Machine Learning, ICML, volume 97, pages 3896–3904, 2019.

[31] Zechen Liu, Zizhang Wu, and Roland Tóth. SMOKE: single-stage monocular 3d object detection via keypoint estimation. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, pages 4289–4298, 2020.

[32] Yanmao Man, Ming Li, and Ryan M. Gerdes. Ghostimage: Remote perception attacks against camera-based image classification systems. In Manuel Egele and Leyla Bilge, editors, 23rd International Symposium on Research in Attacks, Intrusions and Defenses, RAID, pages 317–332, 2020.

[33] Yibo Miao, Yinpeng Dong, Jun Zhu, and Xiao-Shan Gao. Isometric 3d adversarial examples in the physical world. In NeurIPS, 2022.

[34] Takami Sato, Yuki Hayakawa, Ryo Suzuki, Yohsuke Shiiki, Kentaro Yoshioka, and Qi Alfred Chen. Poster: Towards Large-Scale Measurement Study on LiDAR Spoofing Attacks against Object Detection. In Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, pages 3459–3461, 2022.

[35] Shaoshuai Shi. PointRCNN. https://github.com/sshaoshuai/PointRCNN, 2023.

[36] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[37] Hocheol Shin, Dohyun Kim, Yujin Kwon, and Yongdae Kim. Illusion and Dazzle: Adversarial Optical Channel Exploits Against Lidars for Automotive Applications, page 445–467. Cham, 2017.

[38] Jiachen Sun, Yulong Cao, Qi Alfred Chen, and Z. Morley Mao. Towards robust lidar-based perception in autonomous driving: General black-box adversarial sensor attack and countermeasures, 2020.

[39] Tzungyu Tsai, Kaichen Yang, Tsung-Yi Ho, and Yier Jin. Robust adversarial objects against deep learning models. Proceedings of the AAAI Conference on Artificial Intelligence, 34(01):954–962, 2020.

[40] James Tu, Huichen Li, Xinchen Yan, Mengye Ren, Yun Chen, Ming Liang, Eilyan Bitar, Ersin Yumer, and Raquel Urtasun. Exploring adversarial robustness of multi-sensor perception systems in self driving. In Proceedings of the 5th Conference on Robot Learning, volume 164, pages 1013–1024, 2022.

[41] James Tu, Mengye Ren, Sivabalan Manivasagam, Ming Liang, Bin Yang, Richard Du, Frank Cheng, and Raquel Urtasun. Physically realizable adversarial examples for lidar object detection. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[42] Wei Wang, Yao Yao, Xin Liu, Xiang Li, Pei Hao, and Ting Zhu. I can see the light: Attacks on autonomous vehicles using invisible lights. In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, New York, NY, USA, 2021.

[43] Xupeng Wang, Mumuxin Cai, Ferdous Sohel, Nan Sang, and Zhengwei Chang. Adversarial point cloud perturbations against 3d object detection in autonomous driving systems. Neurocomputing, 466:27–36, 2021.

[44] Yuxin Wen, Jiehong Lin, Ke Chen, C. L. Philip Chen, and Kui Jia. Geometry-aware generation of adversarial point clouds. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(6):2984–2999, 2022.

[45] Weibin Wu, Yuxin Su, Michael R. Lyu, and Irwin King. Improving the transferability of adversarial samples with adversarial transformations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 9024–9033, 2021.

[46] Wenzhao Xiang, Hang Su, Chang Liu, Yandong Guo, and Shibao Zheng. Improving the robustness of adversarial attacks using an affine-invariant gradient estimator. Computer Vision and Image Understanding, 229(C), 2023.

[47] Mingfu Xue, Chengxiang Yuan, Can He, Jian Wang, and Weiqiang Liu. Naturalae: Natural and robust physical adversarial examples for object detectors. Journal of Information Security and Applications, 57:102694, 2021.

[48] Yi Zhu, Chenglin Miao, Foad Hajiaghajani, Mengdi Huai, Lu Su, and Chunming Qiao. Adversarial attacks against lidar semantic segmentation in autonomous driving. In Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems, New York, NY, USA, 2021.

[49] Yi Zhu, Chenglin Miao, Tianhang Zheng, Foad Hajiaghajani, Lu Su, and Chunming Qiao. Can we use arbitrary objes to attack lidar perception in autonomous driving? In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, New York, NY, USA, 2021.

[50] ZhuLifa. PointPillars: Fast Encoders for Object Detection from Point Clouds. https://github.com/zhulf0804/PointPillars, 2023.

A Appendix

A.1 Adaptive Defense

Our method is the first to enhance the robustness of adversarial objects by reducing the discrepancies between the digital point clouds and the captured point clouds. To the best of our knowledge, there are no existing countermeasures specifically designed for this. Therefore, we propose a potential countermeasure and evaluate its performance.

A.1.1 Point Clouds Smooth Defense

We observed that, although our reconstructed adversarial object allows for a more stable LiDAR capturing of adversarial perturbations, it exhibits irregular lines in LiDAR scans that differ from those of normal vehicles. Therefore, we naturally consider utilizing a smoothing algorithm on LiDAR scan results, which affects irregular lines more substantially than regular ones. Through the smoothing of irregular lines, we can disrupt adversarial perturbations and lead the attack to failure.

Experiment Setup. We implement a smoothing algorithm and apply it to both the point cloud files generated in Section 6.3 and the KITTI dataset. The former is utilized to assess the attack performance degradation resulting from the smoothing, while the latter is employed to evaluate the impact on AP (Average Precision) of detecting benign objects. The target model is the PointPillars.

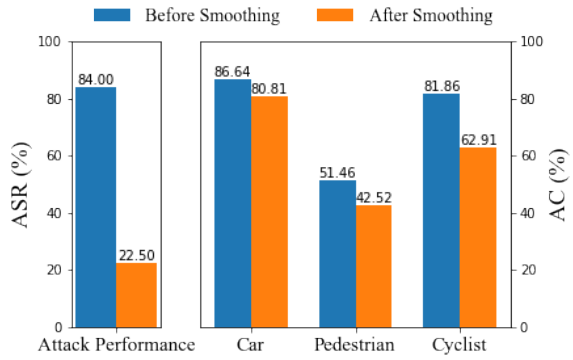

Figure A1: The ASR and the AP before smoothing (left) and after smoothing (right).

Evaluation Results. The results of this experiment are depicted in Figure A1. It is observed that the attack performance decreases from 84% to 22.5%, concurrently leading to a decline in AP for cars from 86.64% to 80.81%, for pedestrians from 51.46% to 42.52%, and for cyclists from 81.86% to 62.91%. While the smoothing algorithm effectively mitigates the impact of reconstructed adversarial objects, it does somewhat reduce the AP of detecting benign objects, especially those with complex surfaces such as pedestrians or cyclists. This trade-off between defending against adversarial attacks and the decline in AP for benign objects is non-negligible, as this reduction persists even in the absence of our reconstructed adversarial obstacles.

A.1.2 Multiple-LiDAR Defenses

As observed in the above experiments, point clouds captured at an angle exhibit a lower overall attack success rate compared to point clouds captured directly from behind. Additionally, utilizing multiple LiDAR-based perception models to detect obstacles simultaneously can help mitigate the risk of being misled. Building upon these intuitions, we propose a defense strategy: utilizing two LiDARs to capture point clouds from different perspectives and inputting these captured point clouds to separate models. This approach aims to minimize the risk of misdirection. We design an experiment to evaluate this defense strategy.

Experiment Setup. The experiment is conducted in the simulator. We utilize two LiDARs and two separate PointPillars models. If either of the two models successfully detects the target vehicle, we consider it a failure in terms of misleading the perception model. we employ the same five reconstructed adversarial ornaments as those used in Section 6.4, all other settings remain consistent with those mentioned in Section 6.4.

Evaluation Results. The results of this experiment are presented in Figure A2. It is observed that the utilization of two LiDARs reduces the attack success rate of our method by a maximum of 10.39%, thereby demonstrating the effectiveness of the adaptive countermeasure. However, this reduction is insufficient to render our method practically unexploitable. Additionally, equipping vehicles with multiple LiDARs and multiple models increases production costs for autonomous vehicles. Balancing safety and cost is a significant concern for carmakers.

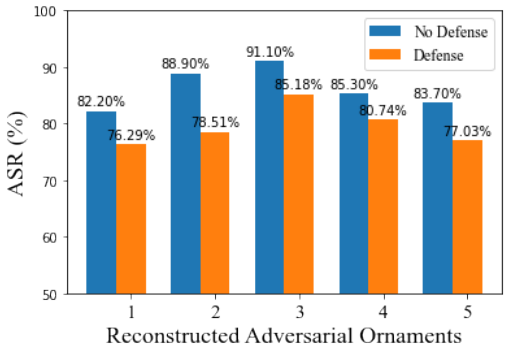

Figure A2: Evaluation results of our proposed countermeasure.

A.1.3 Other Defenses

Another potential countermeasure is to leverage the new IOV (Internet of Vehicles) and CVIS (Cooperative Vehicle Infrastructure System) technologies. With these technologies, vehicles no longer solely rely on their own obstacle-detection capabilities. Instead, they can receive obstacle information from other vehicles and even roadside streetlights. Attempting to deceive multiple sensors and models simultaneously becomes challenging, if not impossible, which limits the usage of our method. However, it is important to note that both IOV and CVIS require additional communication time, which can introduce delays.

A.2 Confirming of the Insight in the Preliminary Analysis

In this section, we compare the changes in point clouds of different objects when the LiDAR moves to confirm the insight mentioned in Section 1, namely, that it is difficult for sparse laser beams to accurately capture the scattered perturbations of adversarial objects, particularly those with irregular surfaces and sharp protrusions.

Firstly, we place an object with a complex surface - a PVC figurine - on the desk (as illustrated in the left half of Figure A3) and move the LiDAR back and forth to capture the point clouds.

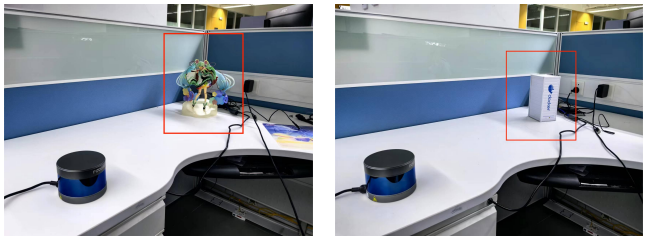

Figure A3: Different objects were considered.

The point clouds before and after the LiDAR's moving differs a lot.

Next, we position a regular object - a box - on the desk (as depicted in the right half of Figure A3) and proceed to move the LiDAR back and forth once more. This time, the point clouds captured before and after the LiDAR's movement exhibited minimal differences.

The pictures of captured points can be found in our website: https://sites.google.com/view/ae-morpher.

A.3 Printability Analysis

In this section, we assess the printability of adversarial ornaments reconstructed by our proposed method. Our evaluation focuses on two primary aspects: 1) the compatibility of these objects with widely-used 3D printers, and 2) the ease of 3D printing these objects.

Experiment Setup. To assess the printability of the adversarial object, we conducted an analysis before and after the surface reconstruction process using *PreForm* [4], a commercial tool for printability assessment. PreForm determines whether a given 3D mesh can be printed using their 3D-printing service. In addition, we utilized *watertightness* as another metric to evaluate the object. Watertightness assesses whether the object's mesh can hold water when filled, it requires the object to have a close and complete surface. That's to say, A 3D object must exhibit watertightness to have a valid volume and exist in the physical world. These two criteria are important metrics for evaluating the printability of meshes [13]. Furthermore, we computed the *self-intersection ratio* as an indicator of the ease of printing adversarial objects. The self-intersection ratio represents the proportion of all faces of an object that intersects each other. A higher surface self-intersection rate indicates a more complex surface and makes printing more challenging.

	Printable		Self-intersection
	Preform	Watertight	
Original	Error	False	1987.50%
Reconstruction	Success	True	0.00%

Table A1: The printability of the original and the reconstruction adversarial object.

Evaluation Results. The evaluation results for adversarial ornaments, with and without our method, are presented in Table A1. Before reconstruction, the adversarial ornament fails PreForm's test and is not watertight, indicating that it is difficult to 3D print. However, after reconstruction, our method not only enables the adversarial ornament to pass these tests but also reduces its self-intersection ratio from 1987.5% to 0%. Some other methods can also decrease the self-intersection ratio. For example, MSF-ADV [13] achieves a self-intersection ratio of 0.46% in their situation. However, these methods simultaneously decrease the attack success rate of the adversarial object (8% for MSF-ADV), whereas our method does not.