

Fusion is Not Enough: Single-Modal Attacks to Compromise Fusion Models in Autonomous Driving

Zhiyuan Cheng¹, Hongjun Choi², James Liang³, Shiwei Feng¹, Guanhong Tao¹,

Dongfang Liu³, Michael Zuzak³, Xiangyu Zhang¹

¹Purdue University, ²Daegu Gyeongbuk Institute of Science and Technology, ³Rochester Institute of Technology

{cheng443, feng292, taog, xyzhang}@purdue.edu

hongjun@dgist.ac.kr

{jcl3689, Dongfang.Liu, mjzeec}@rit.edu

Abstract—Multi-sensor fusion (MSF) is widely adopted for perception in autonomous vehicles (AVs), particularly for the task of 3D object detection with camera and LiDAR sensors. The rationale behind fusion is to capitalize on the strengths of each modality while mitigating their limitations. The exceptional and leading performance of fusion models has been demonstrated by advanced deep neural network (DNN)-based fusion techniques. Fusion models are also perceived as more robust to attacks compared to single-modal ones due to the redundant information in multiple modalities. In this work, we challenge this perspective with single-modal attacks that targets the camera modality, which is considered less significant in fusion but more affordable for attackers. We argue that the weakest link of fusion models depends on their most vulnerable modality, and propose an attack framework that targets advanced camera-LiDAR fusion models with adversarial patches. Our approach employs a two-stage optimization-based strategy that first comprehensively assesses vulnerable image areas under adversarial attacks, and then applies customized attack strategies to different fusion models, generating deployable patches. Evaluations with five state-of-the-art camera-LiDAR fusion models on a real-world dataset show that our attacks successfully compromise all models. Our approach can either reduce the mean average precision (mAP) of detection performance from 0.824 to 0.353 or degrade the detection score of the target object from 0.727 to 0.151 on average, demonstrating the effectiveness and practicality of our proposed attack framework.

Index Terms—3D object detection, adversarial attacks, multi-sensor fusion, autonomous driving.

1. Introduction

The rapid progress in machine learning and deep neural networks has been a key driver of innovation in autonomous driving. As a result, the autonomous driving industry has achieved Level-4 automation on public roads, as evidenced by Baidu Apollo [5], Google Waymo [15], and AutoX [3]. To attain full automation, autonomous vehicles (AVs) require an exceptional perception capacity to comprehend their surrounding environment. Given that perception accuracy

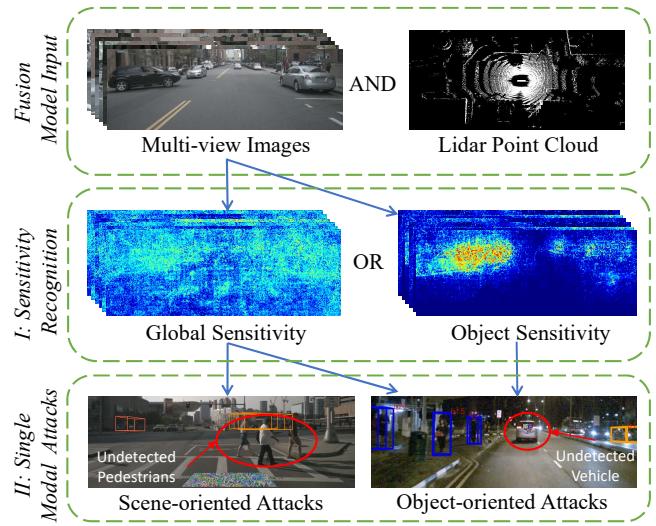


Figure 1: **Single-modal attacks** against camera-LiDAR fusion model using camera-modality.

directly influences the operational safety of autonomous driving, it has attracted significant attention in recent years.

In this study, we primarily concentrate on the safety-critical perception task of 3D object detection, presenting the first attack against advanced camera-LiDAR fusion models by exclusively utilizing the camera modality. In 3D object detection, AVs employ camera and/or LiDAR sensor input to predict the location, size, and categories of surrounding objects. Fusion models, which capitalize on both the high-resolution 2D color and texture information from RGB images provided by cameras and the rich 3D distance information from LiDAR point clouds, have surpassed their solely camera-based or LiDAR-based counterparts in detection accuracy [47], [50], [69]. Furthermore, multi-sensor fusion (MSF) techniques are generally perceived as more robust to attacks [24], [48], as the additional modality offers supplementary information for verifying detection results. Viewed in this light, a counter-intuitive yet innovative question arises: **① Can we attack fusion models through a single modality, even the less significant one, thereby**

directly challenging the security assumption of fusion model robustness? Yet, this fundamental question has not been sufficiently answered in the literature.

Previous research has demonstrated successful attacks against camera-LiDAR fusion models by targeting either multiple modalities [24], [59] or the LiDAR modality alone [39]. However, these approaches are not easy to implement and require additional equipment such as photodiodes, laser diodes [39], and industrial-grade 3D printers [24], [59] to manipulate LiDAR data, thus increasing the deployment cost for attackers. Consequently, we explore the possibility of attacking fusion models via the camera modality, as attackers can more easily perturb captured images using affordable adversarial patches. Nevertheless, as detailed in Section 4.1, this attack design presents additional challenges. For example, the camera modality is considered less significant in fusion models for 3D object detection, because LiDAR provides abundant 3D information. The performance of both state-of-the-art LiDAR-based models and ablations of fusion models using LiDAR surpasses their solely camera-based counterparts significantly [11], [33], [48], [50]. The less significance of camera modality in fusion limits its impact on detection results. Moreover, different fusion models may exhibit distinct vulnerabilities in the camera modality, necessitating varying attack strategies. The cutting-edge patch optimization technique [30] has limitations in generating deployable adversarial patches, as they fail to consider the semantics of the input scene. Hence, a problem remains open: **② How to design single-modal attack to effectively subvert fusion models?**

In response to ②, we propose a novel attack framework against camera-LiDAR fusion models through the less significant camera modality and leave the LiDAR data benign. We utilize adversarial patches as the attack vector, aiming to compromise fusion models and cause false negatives in object detection. As illustrated in Figure 1, our attack employs a two-stage approach to generate an optimal adversarial patch for the target fusion model. In the first stage (2^{nd} row of Figure 1), we identify vulnerable regions in the image input using our novel sensitive region recognition algorithm. This optimization-based method employs a trainable mask to simultaneously identify the sensitivity of different image areas under adversarial attacks, providing a comprehensive inspection of the vulnerabilities and regions of interest in the camera modality for a specific fusion model. Based on the identified vulnerable regions, we then classify fusion models as either object-sensitive or globally sensitive, enabling tailored attack strategies for each type of model. In the second stage (3^{rd} row of Figure 1), we implement different attack strategies for the two types of models to maximize our attack performance. For globally sensitive models, we devise scene-oriented attacks, wherein adversarial patches can be placed on background environments (e.g., roads or walls) to compromise the detection of arbitrary nearby objects (see undetected pedestrians in the red circle at the bottom of Figure 1). For object-sensitive models, we design object-oriented attacks that can compromise the detection of a specific target object by attaching the patch to it (see the

undetected vehicle in the red circle of Figure 1). Compared to the prior patch optimization technique [30], the patches generated by our proposed framework offer a significant advantage by being both physically deployable and effective. We shall open our code and data right after acceptance. In summary, this work makes the following contributions:

- We present the first attack against advanced camera-LiDAR fusion models leveraging only the camera modality, thereby further exposing the security issues of MSF-based AV perception and challenging the security assumption that sensor fusion improves robustness.
- We develop an algorithm for identifying the distribution of vulnerable regions in images, offering a comprehensive assessment of areas susceptible to adversarial attacks, which provides valuable insights into the regions of interest within the visual data for fusion models.
- We introduce a framework for attacking fusion models with adversarial patches, in which we employ a two-stage approach and different attack strategies to either maximize or customize the attack effect based on the recognized sensitivity heatmap of the target model.
- We evaluate our attack using five state-of-the-art fusion models on a real-world dataset collected from industrial-grade AV sensor arrays [22]. Results show that our attack framework successfully compromises all models. The object-oriented attacks are effective on all models, reducing the detection score of a target object from 0.727 to 0.151 on average. The scene-oriented attacks are effective for two globally sensitive models, significantly decreasing the mean average precision (mAP) of detection performance from 0.824 to 0.353.

2. Background and Related Work

AV perception. AVs are equipped with an array of sensors to facilitate interaction with intricate environments. These sensors include cameras, LiDARs, radars, IMUs, GPS, among others, with the perception module being the first component in the AV system to process the sensor data. Perception tasks encompass road lane detection [27], [37], traffic sign/light recognition [44], [76], 3D object detection [19], [47], [48], [69], etc. Many tasks rely on camera sensors and utilize neural networks to extract semantic information from image inputs. However, the 3D object detection is a more challenging and security-critical task, as it requires understanding the three-dimensional nature of the world and identifying obstacles. Hence, sensor fusion techniques are introduced in 3D object detection, leveraging camera and LiDAR sensors for enhanced accuracy in detection outcomes. Some production-grade AVs utilize fully vision-based perception algorithms (e.g., Tesla), while most others adopt camera-LiDAR fusion-based techniques (e.g., Baidu Apollo [4] and Google Waymo [16]). The output of 3D object detection models generally consists of 3D bounding boxes, object categories, and detection scores for objects within the scene. Detection results and other perception outputs are utilized by subsequent modules in AV systems

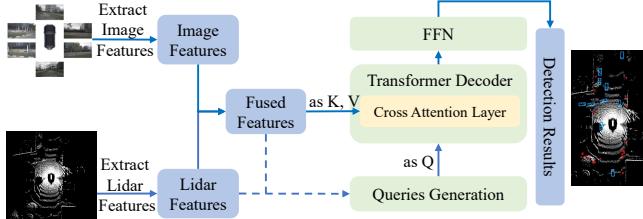


Figure 2: A **general architecture** of state-of-the-art camera-LiDAR fusion models for 3D object detection. FFN represents Feed-Forward Network.

for object tracking, trajectory prediction, route planning, and driving control.

Camera-LiDAR fusion. AVs are equipped with multiple cameras around the vehicle, providing a comprehensive 360-degree view. LiDAR sensors are typically mounted centrally on top of the vehicle, enabling a 360-degree scan of the surrounding environment with laser beams, resulting in a 3D point cloud. Images and point clouds represent distinct modalities, and numerous prior works have investigated methods to effectively fuse them for improved object detection performance. Specifically, [39] categorizes fusion strategies into three groups based on the stage of fusion: 1) cascaded semantic-level fusion, which employs a single-modal detection result to augment the input of the other modality [52], [65]; 2) integrated semantic-level fusion, which conducts independent perception for each modality and subsequently fuses the semantic outputs [4]; and 3) feature-level fusion, which combines low-level machine-learned features from each modality to yield unified detection results. Feature-level fusion can be further divided into alignment-based and non-alignment-based fusion. Alignment-based fusion entails aligning camera and LiDAR features through dimension projection at the point level [29], [46], [62], the voxel level [41], [47], the proposal level [28], [43], or the bird’s eye view [48], [50] before concatenation. For non-alignment-based fusion, cross-attention mechanisms in the transformer architecture are employed for combining different modalities [19], [69]. Contemporary fusion models with a high detection accuracy are predominantly using feature-level fusion. Our primary focus is to investigate and target models utilizing this fusion strategy.

Figure 2 illustrates the general architecture of cutting-edge camera-LiDAR fusion models using feature-level fusion. From left to right, multi-view images obtained from cameras and the point cloud data from LiDAR sensors are initially processed independently by neural networks to extract image and LiDAR features. The networks responsible for feature extraction are commonly referred to as “*backbones*”, which encompass ResNet50 [40], ResNet101 [40], SwinTransformer [49] for images, and SECOND [68], PointNet [53], VoxelNet [75] for point cloud data, among others. Subsequently, the extracted features from each modality are fused together employing either alignment-based or non-alignment-based designs, which vary from model to model. After fusion, a detection head is em-

ployed to generate the final predictions. The 3D object detection head generally adopts a transformer decoder-based architecture in cutting-edge fusion models, as the efficacy of transformers in object detection has been substantiated by DETR [26]. In the transformer-based detection head, the input consists of three sequences of feature vectors named queries (Q), keys (K) and values (V). Each input query vector corresponds to an output vector, representing detection results for an object, including bounding box, object category, and detection score. Input keys and values, derived from fused features, provide scene-specific semantic information. The initial queries generation in various models exhibits distinct design characteristics. For instance, UVTR [47] uses learnable parameters as queries, DeepInteraction [69] and TransFusion [19] employ LiDAR features sampled from fused features, while BEVFusion-MIT [50] and BEVFusion-PKU [48] utilize bird’s eye view fused features. The decoder output is subsequently processed by a Feed-Forward Network (FFN) for final regression and classification results. In this work, we study camera-LiDAR fusion models using feature-level fusion and our attack is general to both alignment-based and non-alignment-based fusion approaches regardless of the design of detection head.

AV perception attacks. AV perception models predominantly rely on Deep Neural Networks (DNNs), which are known to be susceptible to adversarial attacks. Consequently, various attacks have been developed against AV perception tasks, including road lane detection [54], traffic sign/light recognition [56], [58], [73], [74], monocular depth estimation [30], and 3D object detection [17], [20], [23]–[25], [39], [57], [59], [60], [70]. 3D object detection models can be classified into three categories: camera-based, LiDAR-based, and fusion-based models. Attacks targeting each category have been proposed in the context of AV systems. For camera-based models, adversaries typically employ adversarial textures to manipulate the pixels captured by AV cameras [20], [70]. This approach is cost-effective and can be easily implemented through printing and pasting. Recent studies have concentrated on enhancing the stealthiness of the adversarial patterns [30], [34]. In the case of LiDAR-based models, some attackers utilize auxiliary equipment, such as photodiodes and laser diodes, to intercept and relay the laser beams emitted by AV LiDAR systems, thereby generating malicious points in the acquired point cloud for the attack [23], [25], [57]. Alternatively, others employ malicious physical objects with engineered shapes to introduce adversarial points for the attack [17], [60]. Regarding camera-LiDAR fusion models, multi-modal attacks have been developed that perturb both camera and LiDAR input either separately [59] or concurrently [24], using the previously mentioned attack vectors. Additionally, single-modal attacks on LiDAR input have been conducted in a black-box manner [39]. To the best of our knowledge, our study is the first to explore single-modal attacks on fusion models through the camera modality, and we successfully compromise state-of-the-art fusion models using deployable adversarial patches.

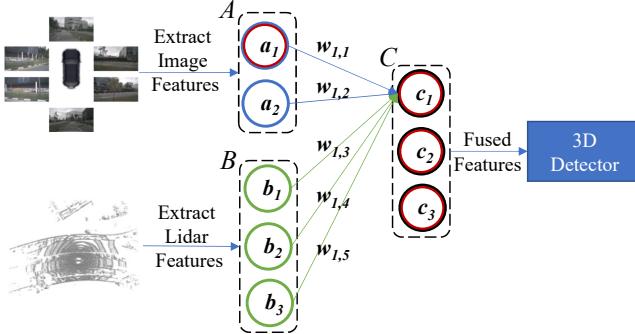


Figure 3: **Simplified illustration** of single-modal attacks against fusion models for 3D object detection. a_i denotes image features, b_i Lidar features and c_i fused features. Nodes marked in red denote affected features by perturbations on image.

3. Attack Goal and Threat Model

Attack goal. The primary goal of our attack is to undermine the object detection capabilities of sensor fusion models employed in autonomous driving systems. We examine the feasibility of exploiting single-modal attacks to deceive multi-modal sensor fusion models, leading to false negative detection results. The inability to detect objects (e.g., pedestrians, traffic barriers, and vehicles) can have disastrous consequences, putting human lives at risk. Although single-modal attacks are relatively easier to deploy than multi-modal attacks, they present additional challenges. Previous studies have investigated the potential of defeating sensor fusion systems by targeting the LiDAR input [39] or both the camera and LiDAR inputs [24], [59]. However, achieving this objective by attacking only the camera input remains an open research problem.

Threat model. Our attack assumes that the attacker has complete knowledge of the camera-LiDAR fusion model used by the target autonomous driving vehicle. Therefore, our attack model is considered to be in a white-box setting. This assumption is consistent with similar works in the literature that implement adversarial attacks on autonomous driving systems [24], [25], [36], [54], [56], [71], [73]. This assumption can be realized through reverse engineering the perception system of the victim vehicle, which has been demonstrated in systems like Tesla Autopilot [9], [10], or through the utilization of open-sourced systems such as Baidu Apollo [4] and Autoware.ai [2]. Secondly, we assume that the attacker has the capability to deploy an adversarial patch onto the ground or a target object, which is practical in the physical world with the patch generated by our framework. We employ Estimation of Transformation (EoT) [18] to enhance the physical-world performance, similar to approaches in [24], [54]. The effectiveness of our patch, considering various angles and distances from the victim vehicle in dynamic backgrounds with arbitrary objects, is evaluated in Section 6. As in [30], [34], [73], the attacker can print the adversarial patch with a home printer and affix it onto an object or draw the pattern on the ground in the form of street paintings [1]. The adversarial pattern can also be concealed

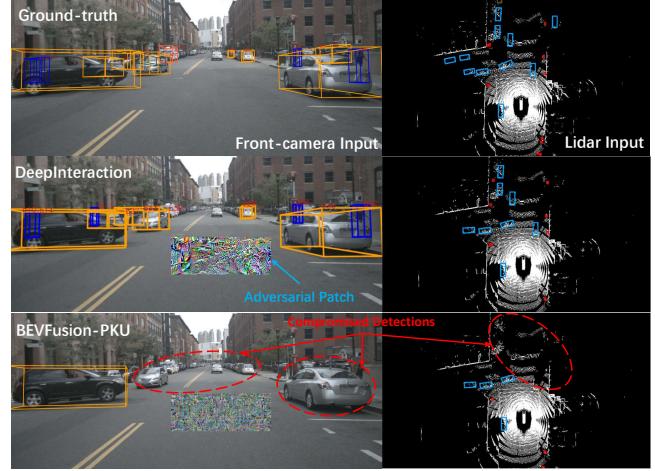


Figure 4: **Motivating example** of adversarial patch attack on camera input against camera-LiDAR fusion models.

within natural textures (e.g., dirt or rust) or artistic designs, utilizing existing camouflage techniques [30], [34] to remain stealthy and persistent, avoiding detection and removal.

4. Challenges and Motivation

4.1. Challenges of Single-Modal Attacks

Single-modal attacks against fusion models are more desirable for adversaries because it reduces the effort required for an attack, yet they present additional challenges:

Fusion models may exhibit increased robustness. Sensor fusion aims to capitalize on the advantages of each modality while mitigating their limitations. Fusing these modalities has been proven to outperform single-modal models in tasks such as 3D object detection [48], tracking [64], and map segmentation [50]. Given the vulnerability of single-modal models to adversarial attacks, researchers seek to improve robustness through fusion techniques, as additional modalities can provide supplementary information for verifying detection results. Traditional fusion methods, such as the Kalman Filter [66], have demonstrated resilience against attacks like sensor spoofing [32], [77]. Thus, multi-modal models may render single-modal attacks more challenging.

Modalities may exhibit varying significance. Since camera and LiDAR sensors provide distinct information for 3D object detection (i.e., texture from cameras and depth from LiDAR), fusion models may assign different weights based on each modality's contribution to the final decision. LiDAR is often considered more critical for 3D object detection due to its rich 3D information. Furthermore, state-of-the-art single-modal 3D object detection models and ablation studies of fusion models with single modality demonstrate that LiDAR-based models outperform camera-based models significantly [11], [33], [48], [50], further validating the varying importance of these modalities. Consequently, attacking through a less important modality (e.g., the camera) may be challenging.

Attack strategy may depend on fusion approaches. Multi-modal 3D object detection models employ various fusion approaches to integrate features from different modalities. In [39], fusion approaches are classified into cascaded semantic-level fusion, integrated semantic-level fusion, and feature-level fusion, while in [48], they are categorized into point-level fusion, feature-level fusion, and bird-eye-view fusion. Different fusion models may result in disparate vulnerabilities, when only a single modality is attacked. For example, when the adversary tries to attack through the camera input, susceptible input regions may differ among these vulnerabilities. Consequently, different attack strategies may be required to maximize attack performance, increasing the complexity of attack design.

4.2. Motivation

In light of the challenges delineated in Section 4.1, we first analyze the feasibility of single-modal attacks on fusion models. Since modern camera-LiDAR fusion models are all based on DNNs, we start with a simplified DNN-based fusion model. As shown in Figure 3, we use $A^\top = [a_1, a_2]$ to represent the extracted image feature vector and use $B^\top = [b_1, b_2, b_3]$ to denote the LiDAR feature vector. Suppose these features are concatenated to a unified vector during fusion and used to calculate the next layer of features $C^\top = [c_1, c_2, c_3]$ with weight parameters $W \in R^{3 \times 5}$. Hence we have:

$$C^\top = W \cdot \text{vstack}(A, B), \quad (1)$$

where `vstack()` is a concatenation operation. Specifically, the elements of C are calculated as follows:

$$c_i = \sum_{j=1}^2 w_{i,j} \cdot a_j + \sum_{j=1}^3 w_{i,2+j} \cdot b_j \quad (i = 1, 2, 3). \quad (2)$$

Now, suppose adversarial perturbations are applied to an area of the input image and some image features (i.e., a_1) are affected while LiDAR features remain benign. Let $A'^\top = [a_1 + \Delta_1, a_2]$ be the adversarial image features. Then the fused features are $C'^\top = [c'_1, c'_2, c'_3]$ calculated as follows:

$$\begin{aligned} c'_i &= \sum_{j=1}^2 w_{i,j} \cdot a_j + \sum_{j=1}^3 w_{i,2+j} \cdot b_j + w_{i,1}\Delta_1 \\ &= c_i + w_{i,1}\Delta_1 \quad (i = 1, 2, 3). \end{aligned} \quad (3)$$

As we show, *every fused feature is tainted by the adversarial features and the effect will finally propagate to the 3D object detection results, making single-modal attacks on prediction results possible*. The degree of effect on the result depends on the weights of the image features (e.g., $w_{i,1}$) in the model. Larger weights could have severe consequence. In addition, the fusion approach could also affect the result. For example, if we only concatenate the second element (a_2) of A with LiDAR features to calculate C in the model, the previous adversarial attack on image cannot work anymore.

Motivating example. The above analysis of simplified DNN-based fusion model provides insights into the feasibility of attacking fusion models through a single modality.

However, the effectiveness of such attacks on complex and practical fusion models remains unknown. Thus, we present an example to illustrate our examination of state-of-the-art fusion models using real-world data. We select a frame from a scene in the Nuscenes dataset [22] containing both camera and LiDAR data (refer to the first row of Figure 4). In this scene, the ego-vehicle navigates a road with multiple cars and pedestrians in the vicinity. State-of-the-art camera-LiDAR fusion models, namely DeepInteraction [69] and BEVFusion-PKU [48], can detect the positions, dimensions and classes of surrounding objects correctly. We then perform a traditional adversarial patch attack [21] and define a patch on the road ahead of the victim vehicle (i.e., ego-vehicle). We optimize the patch content to cause false negative detection of objects, while keeping LiDAR data unaltered. The results for DeepInteraction and BEVFusion-PKU are depicted in the second and third rows of Figure 4, respectively. As illustrated in the second row, DeepInteraction’s object detection remains unaffected by the patch. However, our attack successfully compromises BEVFusion-PKU, causing it to fail in detecting objects near the patch, as indicated by the red circles in the third row of Figure 4. The success in attacking BEVFusion-PKU confirms the feasibility of attacking a fusion model through manipulating a single modality input, even if it is a less significant one. Moreover, the failure to attack DeepInteraction reveals that a uniform attack strategy and location cannot prevail across all fusion models, as different models may exhibit unique vulnerabilities such as distinct susceptible areas.

Our two-stage approach. To automatically identify the most susceptible area in the input image under adversarial attacks and maximize the attack effect on camera-LiDAR fusion models, we propose a two-stage approach. In the first stage, we aim to identify vulnerable regions of the subject model for adversarial attacks on the camera-modality. In the second stage, we generate an adversarial patch based on the identified vulnerable region to maximize the attacking effect. To characterize the vulnerable regions, we introduce the concept of “*sensitivity*” as a property of areas in input images. Sensitivity measures the degree to which specific area of an image impacts adversarial goals when perturbations are introduced. An area with high sensitivity means perturbations there have large influence and can achieve good attack performance. Hence, sensitive regions are more vulnerable to adversarial attacks than other regions. Formally, the sensitivity S_A of an area A is as follows:

$$\begin{aligned} S_A &\propto \max_p \{L_{adv}(x, l) - L_{adv}(x', l)\}, \\ \text{where } x' &= x \odot (1 - A) + p \odot A, \\ x, p &\in [0, 1]^{3 \times h \times w}, A \in \{0, 1\}^{3 \times h \times w}. \end{aligned} \quad (4)$$

Here, x is the input image with height h and width w , l is the LiDAR point cloud and x' is the adversarial image with perturbations p in region A . L_{adv} denotes the adversarial loss defined by adversarial goals. Examining the sensitivity of each area on the image through individual patch optimization is very time consuming. Moreover, it

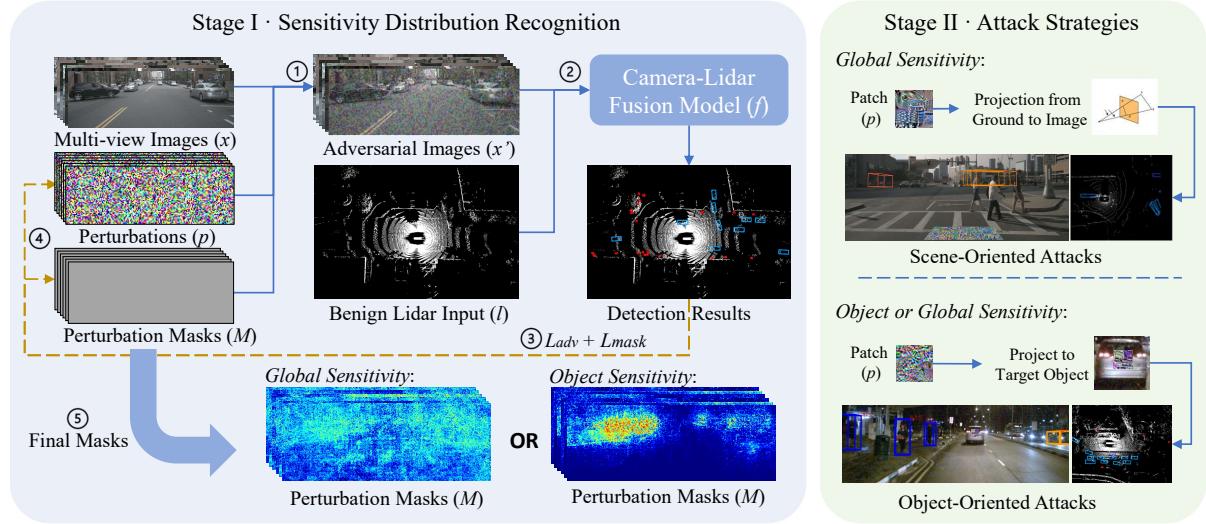


Figure 5: Framework of our single-modal attacks against camera-LiDAR fusion model with adversarial patches.

becomes increasingly unaffordable as the granularity of the considered unit area increases. Despite the availability of numerous decision interpretation methods, such as Grad-CAM [55] and ScoreCAM [63], which generate heatmaps to illustrate areas of attention within images, these techniques are not designed for complex fusion models, thus rendering their direct application challenging. Furthermore, it is essential to differentiate between model decision interpretation and sensitivity recognition, as the motivating example provided demonstrates the road as a susceptible region for adversarial attacks in some models, whereas the primary focus of object detection should be directed towards the objects themselves, as an interpretation method would show. Therefore, to recognize the sensitivity distribution on input images efficiently, we propose a novel optimization-based method in the first stage. Based on this, we classify the distribution into two types: object sensitivity and global sensitivity. In the second stage, we launch adversarial patch attacks on sensitive areas and leverage different attack strategies to either customize the attack effect (e.g., false negative detection) to a specific object or maximize the effect to arbitrary surrounding objects. Our evaluation with five state-of-the-art camera-LiDAR fusion models published in 2022 validates the efficacy of our attack framework and each model with unique sensitivity distributions can be compromised to produce false negative detection results via highly-practical adversarial patches. Regarding the one-stage regional patch optimization technique [30], it could complement our two-stage approach by integrating into the second stage to refine the patch area after vulnerable regions have been identified.

5. Attack Design

Figure 5 presents the framework of our single-modal adversarial attack on camera-LiDAR fusion models using an adversarial patch, employing a two-stage approach. Ini-

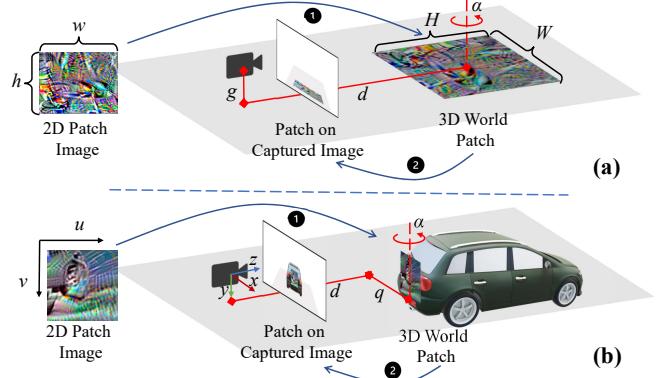


Figure 6: The projection in (a) scene-oriented attacks and (b) object-oriented attacks.

tionally, we identify the sensitivity distribution of the subject network, and subsequently, we launch an attack based on the identified sensitivity type. During the first stage, to recognize the sensitivity distribution, we define perturbations and universal perturbation masks with dimensions identical to the multi-view image input. We then compose the adversarial input by applying the patch and mask to images of a scene sampled from the dataset (step ①). After feeding the adversarial input images and corresponding benign LiDAR data to the subject fusion model, we obtain object detection results (step ②). We calculate the adversarial loss based on the detection scores of objects in the input scene (step ③) and utilize backpropagation and gradient descent algorithm to update masks and perturbations, aiming to minimize adversarial loss and mask loss (step ④). We repeat this process for thousands of iterations until convergence is achieved, and then visualize the final mask as a heatmap to determine the sensitivity type (step ⑤). The heatmap's high-intensity regions signify areas more susceptible to adversarial attacks.

Based on the distribution of sensitive areas, we classify the heatmap into two types: *global sensitivity* and *object sensitivity*. Global sensitivity refers to the distribution of sensitive areas covering the entire scene, including objects and non-object background. Object sensitivity, on the other hand, indicates that only object areas are sensitive, while non-object parts exhibit lower intensity on the heatmap.

In the second stage of our attack framework, we adopt different attack strategies based on the identified sensitivity heatmap type. For global sensitivity, we implement scene-oriented attacks. By placing a patch on the static non-object background (e.g., roads and walls), we deceive the fusion model on the victim vehicle and compromise the detection of untargeted objects surrounding the patch within the scene. For both object sensitivity and global sensitivity, we can employ object-oriented attacks. In this approach, we attach a patch to the target object, causing the object detection model on the victim vehicle to fail in detecting the target object while leaving the detection of other objects unaltered. Since adversarial patches, optimized as 2D images, should be placed in a 3D environment during attacks, directly pasting these patches onto scene images results in unrealistic synthesis and a greater disparity between physical scenarios and experimental conditions, thereby reducing practicality. Consequently, we employ projections proposed in [31] to account for the structure of the physical world while synthesizing the adversarial scene image. Figure 6 depicts the projections in both attack strategies. As shown, pixels of the 2D patch image are first projected to 3D coordinates in the physical world (see step ①) and then projected back to pixels on the captured scene image (step ②), connecting the patch image with the patch area in the scene image. Specifically, in the scene-oriented attacks, the patch can be horizontal on the ground, and we define the longitudinal distance d , the lateral distances q and viewing angle α to control its location. In object-oriented attacks, the patch is vertical, and d , q and α depend on the target object's position in the scene, which can be extracted from the bounding box of the object that is predicted in the benign case. Details of the two stages are discussed in the following subsections.

5.1. Sensitivity Distribution Recognition

Models employing different fusion strategies may exhibit varying distributions of vulnerable regions since the model structure influences the training process and weight assignment. As a fusion model may have millions of weight parameters, identifying sensitive areas by analyzing weight parameters is infeasible. Consequently, we propose an automatic method to recognize the sensitivity distribution of a fusion model on a single-modal input.

The main idea is to leverage the gradients of input data with respect to the adversarial loss. Larger gradients in an input region indicate that small perturbations have a higher impact on the adversarial goal, making it a more "sensitive" area. Taking Figure 3 as an example, through back-propagation, the calculation of an input pixel x_i 's gradient with respect to adversarial loss L_{adv} is shown in

Equation 5.

$$\begin{aligned} \nabla_{x_i} = \frac{\partial L_{adv}}{\partial x_i} &= \sum_{j=1}^2 \left(\frac{\partial L_{adv}}{\partial a_j} \frac{\partial a_j}{\partial x_i} \right) \\ &= \sum_{j=1}^2 \left[\left(\sum_{k=1}^3 \frac{\partial L_{adv}}{\partial c_k} \frac{\partial c_k}{\partial a_j} \right) \frac{\partial a_j}{\partial x_i} \right] \\ &= \sum_{j=1}^2 \left[\left(\sum_{k=1}^3 \frac{\partial L_{adv}}{\partial c_k} w_{k,j} \right) \frac{\partial a_j}{\partial x_i} \right] \end{aligned} \quad (5)$$

As demonstrated, the weights of the single modality in fusion (i.e., $w_{k,j}$ ($k = 1, 2, 3$; $j = 1, 2$)) are involved in the gradient calculation, and higher weights can lead to larger gradients. Therefore, we leverage the gradients as an overall indicator to understand the significance or vulnerability of different areas within the single-modal input.

In a formal setting, the proposed methodology for recognizing sensitivity distribution can be articulated as an optimization problem. The primary objective is to concurrently minimize an adversarial loss and a mask loss, which can be mathematically represented as follows:

$$\arg \min_{p,m} L_{adv} + \lambda L_{mask} \quad (6)$$

$$\text{where } L_{adv} = MSE(f_{scores}(x', l), 0) \quad (7)$$

$$L_{mask} = MSE(M, 0) \quad (8)$$

$$x' = x \odot (1 - M) + p \odot M \quad (9)$$

$$M[i, j] = \frac{1}{2} \times \tanh(\gamma \cdot m[\lfloor \frac{i}{s} \rfloor, \lfloor \frac{j}{s} \rfloor]) + \frac{1}{2} \quad (10)$$

$$\text{s.t. } p \in [0, 1]^{3 \times h \times w}, m \in R^{1 \times \lfloor \frac{h}{s} \rfloor \times \lfloor \frac{w}{s} \rfloor}, \quad (11)$$

Here, x is the image input, which is normalized and characterized by dimensions h (height) and w (width), such that $x \in [0, 1]^{3 \times h \times w}$. The variables l , p , m , and λ represent the LiDAR input, the perturbations on image with dimensions equal to x , the initial mask parameters, and the mask loss weight hyperparameter, respectively. The desired sensitivity heatmap corresponds to the perturbation mask M . Visualization of variables can be found in Figure 5.

Initially, the mask parameters $m \in R^{1 \times \lfloor \frac{h}{s} \rfloor \times \lfloor \frac{w}{s} \rfloor}$ are transformed into the perturbation mask $M \in [0, 1]^{1 \times h \times w}$ using Equation 10. The $\tanh()$ function maps values in m into the $[0, 1]$ range, and its long-tail effect encourages the mask M values to gravitate towards either 0 or 1. The hyperparameters γ and s modulate the convergence speed and heatmap granularity, respectively. Subsequently, the perturbation mask M is utilized to apply the perturbation p to the input image x , resulting in the adversarial image x' , as shown in Equation 9, where \odot denotes element-wise multiplication. Adversarial image x' and benign LiDAR data l are then feed to the fusion model. Since our attack goals are inducing false negative detection results, one objective of our optimization is to minimize the detected object scores. Hence, we use the mean square error (MSE) between the scores and zero as the adversarial loss L_{adv} (Equation 7). In this context, f_{scores} refers to the camera-LiDAR fusion model, and the output consists of the detected object scores.

The optimization's secondary objective is to minimize the perturbation mask values, achieved by incorporating a mask loss L_{mask} (Equation 8).

The optimization of these two losses is a dual process. Minimizing the adversarial loss (i.e., maximizing attack performance) necessitates a higher magnitude of perturbations on the input. Conversely, minimizing the mask loss indicates a lower magnitude of perturbations. As a result, the dual optimization process converges on applying higher magnitude perturbations on sensitive areas (to improve attack performance) and lower magnitudes for insensitive parts (to minimize mask loss). Upon analyzing the optimization process from a gradient perspective, it becomes evident that the gradients of M regarding the mask loss steer towards the direction of minimizing mask values. In contrast, the gradients with respect to the adversarial loss exhibit an opposite direction. As indicated in Equation 5, areas with higher sensitivity and greater weights in fusion possess larger gradients regarding L_{adv} , resulting in areas with higher intensity on the mask following optimization. Consequently, the mask M serves as a good representation of the sensitivity distribution, and visualizing M allows for the attainment of the sensitivity heatmap. Then we can further classify the fusion model into object sensitivity or global sensitivity by comparing the expectation of the average intensity of object areas with non-object background in each scene as follows:

$$S(f) = \begin{cases} Object, & \mathbf{E}_x \left[\frac{\sum(M \odot A_o)}{\sum A_o} \right] > \beta \mathbf{E}_x \left[\frac{\sum(M \odot (1 - A_o))}{\sum(1 - A_o)} \right] \\ Global, & otherwise \end{cases} \quad (12)$$

Here, $S(f)$ represents the sensitivity type of fusion model f , and A_o is a mask with values 1 denoting the object areas and 0 denoting the non-object areas in scene x . A_o is obtained through utilizing the pixels covered by the bounding boxes of objects that are detected in benign cases. M refers to the recognized sensitivity heatmap of x . β is the coefficient of the classification threshold and is set to 3 in our experiments.

5.2. Attack Strategies

In the second stage of our attack framework, we introduce two attack strategies based on the fusion model's sensitivity type: scene-oriented attacks and object-oriented attacks. Both strategies employ optimization-based adversarial patch generation methods. The optimization problem can be formally represented as:

$$\min_p \quad \mathbf{E}_{(x,l) \sim D} [MSE(f_s(x',l), 0)] \quad (13)$$

$$\text{where } x' = x \odot (1 - M_x) + p_x \odot M_x \quad (14)$$

$$M_x = \text{proj}_x(M), p_x = \text{proj}_x(p) \quad (15)$$

$$\text{s.t. } p \in [0, 1]^{3 \times h \times w}, M \in \{0, 1\}^{1 \times h \times w}. \quad (16)$$

Here, multi-view images x and the corresponding LiDAR data l are randomly sampled from the training set D . The initial patch image p and patch mask M possess the same width (w) and height (h) as x . The mask M represents a patch area for cropping the patch image, with values equal to 1 inside the patch area and 0 elsewhere. Unlike Equation 6,

M contains discrete values and is not optimizable. $proj_x()$ denotes a linear perspective projection function to project the initial patch and patch mask onto a specific region of x . The projection learns from [31] and the target area is contingent upon the attack strategy. The adversarial input x' , obtained by applying p_x and M_x to the original image input x (Equation 14), is subsequently fed into the fusion model f_s along with benign LiDAR data l . The output of f_s consists of detected object scores, which vary in scope depending on specific attack strategies. We minimize the MSE between detected object scores and zero to achieve false negative detection results, and we leverage the Expectation of Transformation (EoT) [18] across all training samples to enhance the robustness and generality of our attack.

Specifically, for *scene-oriented attacks*, the goal is to compromise the detection of arbitrary objects near an adversarial patch attached to the environment (e.g., road or wall) of a target scene. The optimization goal in Equation 13 for scene-oriented attacks can be expressed as:

$$\min_p \mathbf{E}_{(x,l) \sim D_s} [MSE(f_{s_all}(x',l), 0)] . \quad (17)$$

In this scenario, the training set D_s is composed of the target scene in which the ego-vehicle is stationary (e.g., at an intersection or parking lot). The categories and locations of objects surrounding the ego-vehicle in the scene can change dynamically. The output of the fusion model f_{s_all} during optimization is the detection score of *all* detected objects in the target scene. The function $proj_x$ projects the patch image and mask onto a certain area of the background environment (e.g., the road) and then maps it back to the scene image, as shown in Figure 6a. This process can be expressed formally with Equation 18 and 21, where (u^p, v^p) denotes a pixel on the patch image, (x^p, y^p, z^p) the corresponding 3D coordinates on the physical patch in the camera's coordinate system, (u^s, v^s) the corresponding pixel on the scene image, W and H the physical width and height of the patch, g the height of the front camera on the ego-vehicle and K the camera intrinsic.

$$\begin{bmatrix} x^p \\ y^p \\ z^p \\ 1 \end{bmatrix} = B \cdot \begin{bmatrix} W/w & 0 & -W/2 \\ 0 & 0 & g \\ 0 & -H/h & H/2 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} u^p \\ v^p \\ 1 \end{bmatrix}, \quad (18)$$

$$\begin{bmatrix} x^p \\ y^p \\ z^p \\ 1 \end{bmatrix} = B \cdot \begin{bmatrix} W/w & 0 & -W/2 \\ 0 & H/h & -H/2 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} u^p \\ v^p \\ 1 \end{bmatrix}, \quad (19)$$

$$B = \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha & q \\ 0 & 1 & 0 & 0 \\ \sin \alpha & 0 & \cos \alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (20)$$

$$\begin{bmatrix} u^s & v^s & 1 \end{bmatrix}^\top = 1/z^p \cdot K \cdot \begin{bmatrix} x^p & y^p & z^p & 1 \end{bmatrix}^\top \quad (21)$$

For *object-oriented attacks*, the goal is to compromise the detection of the target object with an attached adversarial

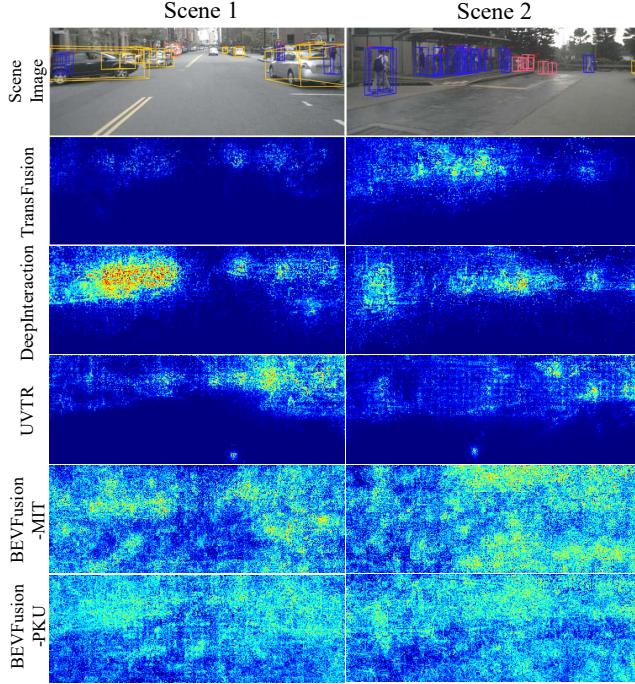


Figure 7: The **sensitivity heatmap** of five camera-LiDAR fusion models on two scenes.

patch while keeping other objects unaffected. The optimization goal can be represented as:

$$\min_p \mathbf{E}_{(x,l) \sim D_t} [MSE(f_{s_target}(x', l), 0)]. \quad (22)$$

In this case, the training set D_t contains a scene in which the target object appears in each data frame. The ego-vehicle may drive following the target object where the background changes dynamically. The output of the fusion model f_{s_target} during optimization is the detection score of the target object exclusively. The function $proj_x$ projects the patch image and mask onto the target object in the scene image using Equation 19 and Equation 21. Unlike the scene-oriented attack in which location of the patch is defined by us using longitudinal distance d , lateral distances q and viewing angle α , in object-oriented attacks (Figure 6b), these parameters depend on the position of the target object in the scene and can be extracted from the predicted 3D bounding box of the target object in benign cases.

6. Evaluation

6.1. Evaluation Methodology and Setup

Model selection In our evaluation, we use five state-of-the-art camera-LiDAR fusion-based 3D object detection models that are published in 2022. These models include Transfusion [19], DeepInteraction [69], UVTR [47], BEVFusion-MIT [50] and BEVFusion-PKU [48]. These models cover a diverse range of feature-level fusion approaches, including alignment-based fusion, non-alignment-based fusion, and various detection head designs. Detailed selection criteria

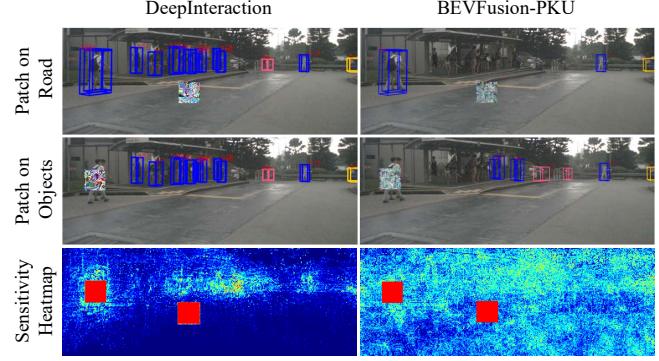
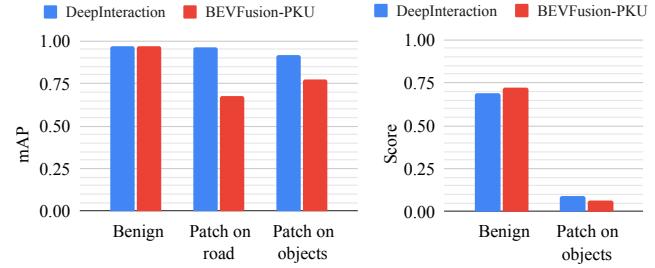


Figure 8: Property validation of the sensitivity heatmap using image-specific adversarial patches.



(a) mAP of all objects in the scene. (b) Score of patched objects.

Figure 9: **Performance of image-specific adversarial patch in validating properties of sensitivity heatmap.**

regarding representativeness, practicality and accessibility can be found in Appendix A.

Evaluation Scene Selection. Our evaluation scenes are selected from the Nuscenes dataset [22]. This dataset contains real-world multi-view images and point cloud data collected from industrial-grade sensor array, and they are derived from hundreds of driving clips. The selected scenes for testing in our evaluation contains 375 data frames, encompass diverse road types, surrounding objects and time-of-day situations. By leveraging this rich dataset, our evaluation framework benefits from a scientifically-rigorous and accurate representation of real-world driving scenarios.

6.2. Sensitivity Distribution Recognition

This section reports on the evaluation of our sensitivity distribution recognition method. Firstly, we present the qualitative results of the sensitivity heatmap generated by our method. Subsequently, we use image-specific adversarial patches on regions with varying degrees of sensitivity to validate the property of our sensitivity heatmap in presenting the vulnerable image areas susceptible to adversarial attacks.

Sensitivity heatmaps. We utilize Equation 6 to generate the sensitivity heatmap for the five fusion models, using two different scenes, each with varying proportions of vehicles and pedestrians. During the optimization, we set the hyperparameters λ to 1, s to 2, and γ to 1. We adopt an Adam [42] optimizer with a learning rate of 0.001 and conduct 2000 iterations of optimization. Due to the unique size of the input images x for each model, we scale and crop the generated

TABLE 1: Attack performance of the **scene-oriented adversarial patch attack** against 3D object detection.

Models		mAP	CR	TK	BS	TR	BR	PD	BI
BF-PKU	Ben.	0.824	0.453	0.448	1.000	0.991	0.898	0.990	0.989
	Adv.	0.353	0.136	0.116	0.524	0.239	0.611	0.242	0.604
	Diff.	-0.47	-0.32	-0.33	-0.48	-0.75	-0.29	-0.75	-0.39
BF-MIT	Ben.	0.886	0.538	0.939	0.858	0.992	0.895	0.989	0.990
	Adv.	0.553	0.279	0.652	0.720	0.488	0.623	0.337	0.772
	Diff.	-0.33	-0.26	-0.29	-0.14	-0.50	-0.27	-0.65	-0.22
TF	Ben.	0.758	0.493	0.451	0.700	0.991	0.692	0.989	0.990
	Adv.	0.759	0.494	0.452	0.706	0.992	0.693	0.989	0.989
	Diff.	0.001	0.001	0.001	0.006	0.001	0.001	0.000	-0.00
DI	Ben.	0.807	0.459	0.522	0.947	0.990	0.750	0.989	0.989
	Adv.	0.808	0.460	0.529	0.947	0.990	0.751	0.989	0.989
	Diff.	0.001	0.001	0.007	0.000	0.000	0.001	0.000	0.000
UVTR	Ben.	0.850	0.557	0.989	0.704	0.990	0.736	0.982	0.989
	Adv.	0.862	0.558	0.989	0.786	0.990	0.741	0.982	0.989
	Diff.	0.013	0.001	0.000	0.082	0.000	0.005	0.000	0.000

Abbreviations. BF-PKU: BEVFusion-PKU [48], BF-MIT: BEVFusion-MIT [50], TF: TransFusion [19], DI: DeepInteraction [69], UVTR [47], CR: Car, TK: Truck, BS: Bus, TR: Trailer, BR: Barrier, PD: Pedestrian, BI: Bicycle.

sensitivity heatmap of different models to 256×704 for better visualization. This size matches the smallest input size of BEVFusion-MIT [50]. Figure 7 depicts the generated sensitivity heatmaps. The first row displays the scene images captured by the front camera of the ego vehicle while the subsequent rows exhibit the sensitivity distributions, i.e., sensitivity heatmaps, of the corresponding scene image using different models. The brightness or warmth of colors in the heatmap corresponds to the sensitivity of a particular region to adversarial attacks. Higher brightness areas signify higher susceptibility to attacks, while lower brightness denotes more robustness. Observe that the sensitive regions for the initial three models, namely Transfusion [19], DeepInteraction [69] and UVTR [47], primarily lie on objects like vehicles and pedestrians. This suggests that attacks on objects could prove to be more effective, whereas non-object areas such as the road and walls are more resistant. The last two models (BEVFusion-MIT [50] and BEVFusion-PKU [48]) demonstrate high sensitivities throughout the entire scene, irrespective of objects or background regions. This indicates their vulnerability at a global level. Since different sensitivity types demonstrate distinct level of vulnerabilities, we discuss the reason behind in our defence discussion (Section 7).

Property validation. To validate the utility of sensitivity heatmaps in identifying vulnerable regions to adversarial attacks, we employ traditional image-specific adversarial patch on regions with distinct levels of sensitivity and evaluate the adversarial performance on two fusion models (i.e., DeepInteraction and BEVFusion-PKU) with distinct sensitivity types. We define a patch on both object and non-object areas to compare models’ vulnerabilities since the two areas demonstrate different brightness on the sensitivity heatmap of DeepInteraction. More specifically, we define a patch area of size 50×50 on objects or roads within Scene 2, and generate the patch to minimize all object scores in the



Figure 10: Comparison between the benign and adversarial cases of **scene-oriented attacks**.

scene. We utilize the Adam optimizer with a learning rate of 0.001 and execute the optimization for 5000 iterations. Our experimental results are shown in Figure 8 and Figure 9.

In Figure 8, the first and second rows illustrate the patch on road and objects, respectively. The third row shows the sensitivity heatmaps with selected patch areas designated in red for reference purposes. Each column represents a distinct model. Our findings demonstrate that the effect of adversarial patches on the detection capability of DeepInteraction is negligible for the patch on the road; however, the patch on the object leads to compromised detection of the patched objects. These results align with the sensitivity heatmap, where the object area shows greater intensity compared to the road area, meaning the object area is more vulnerable. Contrarily, BEVFusion-PKU is globally sensitive, as shown by the sensitivity heatmap. It is observed that patches at both locations can cause false negative detection of surrounding objects, which is consistent with the heatmap and thus validates its accuracy.

Quantitative results are presented in Figure 9. We show in Figure 9a the mean average precision (mAP) of the detection results of all objects in the current scene, and we show in Figure 9b the average detection score of the two pedestrians covered by the patch in the patch-on-objects case. Our findings reveal that the patch on road only affects the BEVFusion-PKU model, and has no significant impact on DeepInteraction, as indicated by the unchanged mAP. However, the patch on object is demonstrated to be effective for both models, as the detection scores for the

TABLE 2: Attack performance of the **object-oriented adversarial patch attack**.

Models	Targeted object			Other objects		
	Ben. Score	Adv. Score	Diff.	Ben. mAP	Adv. mAP	Diff.
TransFusion	0.655	0.070	-0.584	0.921	0.923	0.003
DeepInteraction	0.658	0.110	-0.549	0.964	0.965	0.001
UVTR	0.894	0.189	-0.705	0.963	0.963	0.000
BEVFusion-MIT	0.714	0.219	-0.495	0.965	0.968	0.003
BEVFusion-PKU	0.712	0.168	-0.544	0.956	0.958	0.001
Average	0.727	0.151	-0.575	0.954	0.955	0.002

patched objects decrease substantially in Figure 9b. Notably, the impact of the adversarial patch on DeepInteraction is confined mainly to the patched object, with only a minor effect on the mAP of the scene (see the minor decrease in the third blue bar of Figure 9a). In contrast, the impact on BEVFusion-PKU is more comprehensive, affecting not only the patched object but also surrounding objects, leading to a greater decrease in mAP than in DeepInteraction (see the third red bar of Figure 9a).

In summary, our findings substantiate the reliability of sensitivity heatmaps as an effective metric to identify susceptible regions and determine optimal targets for adversarial patch attacks. Our study has additionally revealed that models with global sensitivity are more susceptible to such attacks since both object and non-object areas can be targeted, while only object areas can be exploited in models with object sensitivity. Consequently, we have devised distinct attack strategies for models with different sensitivity distributions, which are described and assessed in Section 6.3 and Section 6.4.

6.3. Scene-oriented Attacks

Scene-oriented attacks are primarily aimed at fusion models with global sensitivity. Such models are vulnerable to adversarial patches placed upon non-object background scenes, as evidenced in Section 6.2. In contrast to the regional patch-on-road attack that we previously assessed, scene-oriented attacks are not restricted to the per-frame level but can affect the detection of arbitrary objects in a given scene, even those that were not initially present during the patch generation. Therefore, this type of attack poses a more significant and practical threat in real-world scenarios as attackers can effortlessly paste generated adversarial patches onto the ground, rendering victim vehicles in close proximity blind. This could pose a significant risk to pedestrians and surrounding vehicles.

Experimental setup. We select one scene from the Nuscenes [22] dataset in which the ego-vehicle is stationary at a traffic light (see Figure 10). Note that a Nuscenes scene represents a driving clip that lasts about one minute. This scene includes 490 frames of multi-modal data (multi-view images, 360-degree LiDAR point clouds, and object annotations), and the object types and locations vary across different frames. We split the scene into two subsets: the first 245 frames are used as the “training set” to generate

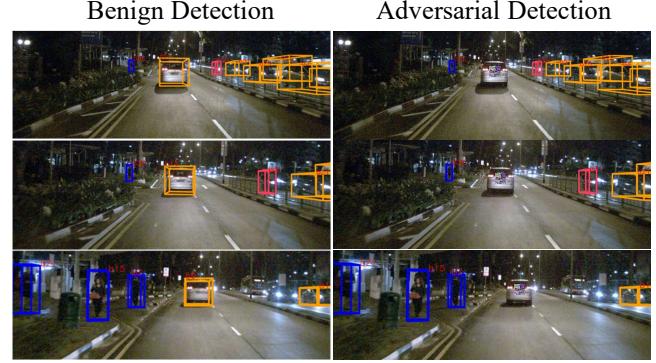


Figure 11: The benign and adversarial cases of **object-oriented attacks**.

an adversarial patch on the ground in front of the victim vehicle, using Equation 17. The remaining 245 frames are used as the “test set” to measure the attack performance. We use mAP as the overall metric and average precision (AP) for seven object categories as the specific metrics. The dimensions of the designated patch are 2 m × 2 m, situated at a distance of 7 m from the ego vehicle on the ground. We map the patch region in the physical world onto the front-camera image with the LiDAR-to-image projection matrix obtained from the dataset. The optimization process utilizes the Adam algorithm, incorporating a batch size of 5 and a learning rate of 0.01, executed over 1000 iterations.

Results. Table 1 and Figure 10 present the quantitative and qualitative results of our evaluation. In Table 1, the first column shows various models, the third column presents the mAP of object detection results in the test set, and the subsequent columns denote the average precision (AP) of different objects categories. We report the benign performance (no attack), adversarial performance (patch attack) and their difference (attack performance) for each model. Our findings indicate that the model performance of the two globally sensitive models (i.e., BEVFusion-PKU [48] and BEVFusion-MIT [50]) has considerably decreased, for all object categories. The mAP of their detection results decreased more than 35%. However, the other three models with object sensitivity remain unaffected. These results align with our conclusion in Section 6.2 and further reveal the vulnerability of globally sensitive models to more practical scene-oriented attacks. Additionally, our experiment confirms the robustness of object-sensitive models against attacks in non-object background areas. Figure 10 demonstrates the benign and adversarial scenarios in the test set and the corresponding object detection outcomes of BEVFusion-PKU [48]. It is evident from the right column that the majority of objects in proximity to the adversarial patch are undetected, encompassing cars, buses, pedestrians, and other entities, which highlights a considerable safety concern for both the ego-vehicle and individuals at the intersection.

6.4. Object-oriented Attacks

Object-oriented attacks target object-sensitive models that are more robust to attacks in non-object background

TABLE 3: Attack performance with **various patch distances**.

Distance	mAP	Automotive	Barrier	Pedestrain	Bicycle
7.0m	-0.208	-0.221	-0.629	-0.476	-0.990
7.5m	-0.337	-0.377	-0.408	-0.609	-0.842
8.0m	-0.428	-0.586	-0.443	-0.750	-0.742
8.5m	-0.261	-0.023	-0.884	-0.651	-0.990
9.0m	-0.211	0.001	-0.615	-0.512	-0.990
9.5m	-0.001	0.000	0.000	-0.010	0.000
10.0m	-0.027	0.014	-0.151	-0.169	0.000
Average	-0.210	-0.170	-0.447	-0.454	-0.651

areas (i.e., scene-oriented attacks). As discussed in Section 6.2, the influence of attacks on objects for object-sensitive models is more localized, as opposed to the more widespread effects in globally sensitive models. Consequently, we devise object-oriented attacks to undermine the detection of specific objects by attaching an adversarial patch. In contrast to scene-oriented attacks, where a static patch is positioned on the ground, affecting arbitrary near-by objects, object-oriented attacks concentrate their impact on a specific target object, leaving the detection of other objects unaltered. The adversarial patch moves in conjunction with the attached object. This approach offers a higher degree of customization for attackers, enabling them to manipulate the impact at the object level rather than the entire scene.

Experimental setup. We select an additional scene from the Nuscenes dataset, featuring a target vehicle driving in close proximity ahead of the ego-vehicle (see Figure 11). This scene comprises 260 sample frames, exhibiting dynamic variations in object types, positions, and background scenarios. Similar to the scene-oriented attacks, we utilize the first half of the frames as the "training set" and the latter half as the "testing set." We utilize the object-oriented projection (see Figure 6b) to map the patch image onto an area of the target object in the scene image, with physical location parameters d , q and α extracted from the ground-truth 3D bounding boxes of the object. This area covers approximately one-ninth of the target vehicle's rear area and maintains a consistent location across all frames. Using Equation 22, the optimization process employs the Adam optimizer with a batch size of 5 and a learning rate of 0.01, executed for 1000 iterations. During testing, we measure and report the target object's average detection score and the mAP of detection results for other objects in the scene.

Results. Our evaluation results are presented in Table 2 and Figure 11. In Table 2, the first column represents various fusion models, the second to fourth columns display the average detection score of the target object, and the fifth to seventh columns indicate the mAP of other objects. The results demonstrate a substantial decrease in the target object's detection scores, from 0.727 to 0.151 on average, thus validating the efficacy of our object-oriented adversarial attacks across all models. Furthermore, the detection results of other objects in the scene remain virtually unaffected, as evidenced by the negligible change in mAP. Consequently, object-oriented attacks can be more customized and target-specific. It is noteworthy that object-oriented attacks are

TABLE 4: Attack performance with **various patch angles**.

Angles	mAP	Automotive	Barrier	Pedestrain	Bicycle
-15°	-0.274	-0.317	-0.430	-0.761	-0.277
-10°	-0.258	-0.047	-0.795	-0.604	-0.990
-5°	-0.270	-0.312	-0.447	-0.757	-0.247
0°	-0.428	-0.586	-0.443	-0.750	-0.742
5°	-0.236	-0.006	-0.804	-0.548	-0.990
10°	-0.264	-0.305	-0.421	-0.757	-0.248
15°	-0.252	-0.015	-0.741	-0.729	-0.990
Average	-0.283	-0.227	-0.583	-0.701	-0.641

effective not only for object-sensitive models but for all models, in contrast to scene-oriented attacks, which are exclusively effective for globally sensitive models. Figure 11 demonstrates the patched target vehicle in multiple frames and the failure of detecting it with DeepInteraction as the subject fusion model.

6.5. Ablation Studies

Varying distance and viewing angles. In order to conduct a comprehensive evaluation of our adversarial attacks, we modify the distance and viewing angles of the adversarial patch and assess the scene-oriented attack performance on the BEVFusion-PKU [48] model. During the distance evaluation, we position a patch with dimensions of 3m × 5m on the ground in front of the ego-vehicle, varying the distance from 7 meters to 10 meters. In the viewing angle assessment, we place the patch 8 meters from the ego-vehicle and rotate it around the z-axis (i.e., the vertical axis perpendicular to the ground) from -15 degrees to 15 degrees. All other experimental settings remain consistent with Section 6.3. The performance degradation caused by our attack is reported in Table 3 and Table 4. The first column represents various distances and viewing angles, while the second column exhibits the mean Average Precision (mAP) degradation of detection results. Subsequent columns display the average precision degradation for individual object categories. The automotive group encompasses cars, trucks, buses, and trailers. As illustrated in Table 3, the adversarial patch demonstrates strong attack performance within the range of 7 meters to 9 meters, with diminishing effectiveness beyond 9 meters. This decrease in efficacy may be attributed to the patch appearing smaller in the camera's field of view as the distance increases, resulting in fewer perturbed pixels and a consequent decline in attack performance. In real-world scenarios, distances less than 9 meters are practical for initiating scene-oriented attacks. For example, street paint [1] at intersections is typically less than 9 meters to the leading vehicle. Observations from Table 4 indicate that the patch maintains robust attack performance across varying angles, with optimal performance occurring at no rotation (0°). Both distance and angle results reveal that attack performance is better for foreground objects (e.g., pedestrians and bicycles) situated closer to the adversarial patch and ego-vehicle. Another ablation study about the granularity of sensitivity heatmap can be found in Appendix B.

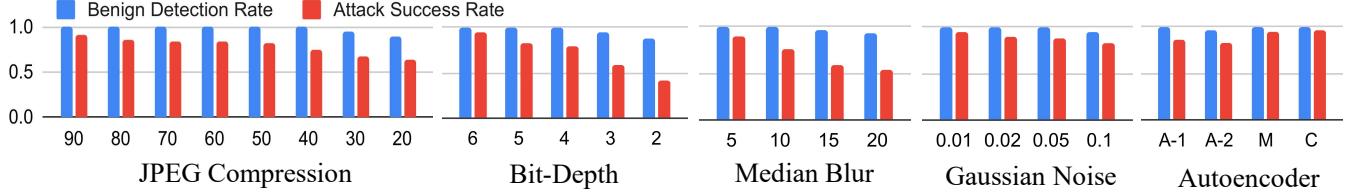


Figure 12: Five directly-applicable **defense methods**.

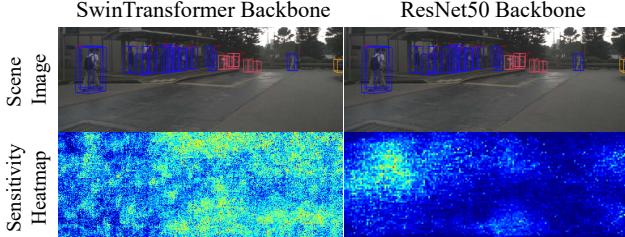


Figure 13: Sensitivity heatmap of BEVFusion-MIT [50] using **different image backbones**.

7. Defence Discussion and Limitations

Architecture level defence. Our analysis reveals that camera-LiDAR fusion models exhibit two types of sensitivity to adversarial attacks: global and object sensitivity. Globally sensitive models are more vulnerable as they are susceptible to both scene-oriented and object-oriented attacks. In contrast, object-sensitive models are more robust due to their smaller sensitive regions and resistance to non-object area attacks. Both model types, however, perform similarly in benign object detection. We investigate the architectural designs to understand the cause of different sensitivity types. We find that object-sensitive models (DeepInteraction [69], UVTR [47], and Transfusion [19]) employ ResNet50 [40] as their image backbone, while globally sensitive models (BEVFusion-PKU [48] and BEVFusion-MIT [50]) use SwinTransformer [49]. To further investigate, we retrain BEVFusion-MIT [50] with ResNet50 and compare the sensitivity heatmap to the original SwinTransformer model, as shown in Figure 13. The results indicate that sensitive regions are more focused on objects when using ResNet50, suggesting that the image backbone significantly impacts model vulnerability. An explanation is that the CNN-based ResNet50 focuses more on local features due to its small convolutional kernels, while the transformer-based SwinTransformer captures more global information through self-attention. Consequently, adversarial patches distant from objects can still affect detection in transformer-based backbones. To enhance the model’s security against such attacks, incorporating ResNet50 as the image backbone is a preferable architectural choice.

DNN level defence. Despite numerous defense strategies proposed for adversarial attacks, to the best of our knowledge, none specifically target camera-LiDAR fusion models employed in 3D object detection. To assess the efficacy of our attack under various defenses, we apply five widely-used defense techniques that perform input transformations without necessitating the retraining of the victim network:

JPEG compression [35], bit-depth reduction [67], median blurring [67], Gaussian noise addition [72], and autoencoder reformation [51]. Comprehensive configurations of these methods can be found in [54]. We execute object-oriented attacks, as detailed in Section 6.4, and implement the defense techniques on input images. Utilizing DeepInteraction [69] as the fusion model, we report both the benign detection rate of the target object without attacks and the attack success rate when patches are applied. An optimal defense should simultaneously maximize the benign detection rate and minimize the attack success rate. Our results are depicted in Figure 12. For each defense technique, we modify the parameters to regulate defense strength. Although the attack success rate decreases as the defense strength increases, our method consistently achieves high success rates (over 70%) across scenarios where the benign performance is not significantly impacted by the defense technique. This outcome may be attributed to the fact that the defenses primarily disrupt digital-space, human-imperceptible perturbations, and are therefore less effective against physically optimized adversarial patch attacks with unbounded perturbations [54]. These findings indicate that directly applicable defense methods are insufficient in thwarting our attack without compromising benign performance. Consequently, novel adaptations of advanced defenses or the development of new defense techniques tailored to sensor fusion models are necessary, a direction we propose for future research.

Limitations While our results demonstrate successful attacks against state-of-the-art camera-LiDAR fusion models using the camera modality, we have not conducted an end-to-end evaluation on an actual AV to illustrate the catastrophic attack outcomes (e.g., collisions or sudden stops). This limitation stems from cost and safety concerns and is shared by other studies in AV security research [24], [25], [54]. It should be noted that the most advanced fusion models examined in this work have not yet been implemented in production-grade autonomous driving systems. Publicly available systems, such as Baidu Apollo [4] and Autoware.ai [2], employ integrated semantic-level fusion rather than the feature-level fusion investigated here. As a result, we did not demonstrate an end-to-end attack in simulation. However, feature-level fusion is gaining attraction in both academia [48], [50], [69] and industry [45], driven by advancements in network designs and enhanced performance. Our attack is applicable to all feature-level fusion models. Furthermore, our evaluations using a real-world dataset and with industrial-grade AV sensor array underscore the practicality of our attack in real-world scenarios.

8. Conclusion

We challenge the security assumption that fusion models are more robust to attacks through single modal attacks against camera-LiDAR fusion models. We leverage the affordable adversarial patch to attack the less significant camera modality in 3D object detection. The proposed optimization-based two-stage attack framework can provide a comprehensive assessment of image areas susceptible to adversarial attacks through a sensitivity heatmap, and can successfully attack five state-of-the-art camera-LiDAR fusion models on a real-world dataset with customized attack strategies. Results show that the adversarial patch generated by our attack can effectively decrease the mAP of detection performance from 0.824 to 0.353 or reduce the detection score of a target object from 0.727 to 0.151 on average.

References

- [1] Art painted on crosswalks makes streets safer. <https://www.washingtonpost.com/lifestyle/2022/06/08/crosswalk-art-safety-bloomberg/>.
- [2] Autoware. <https://www.autoware.org/>.
- [3] AutoX Opens Real Robotaxi Service In China To The General Public. <https://www.forbes.com/sites/bradtempleton/2021/01/27/autox-opens-real-robotaxi-service-in-china-to-the-general-public/>.
- [4] Baidu Apollo. <https://apollo.auto/index.html>.
- [5] Baidu fully opens Apollo Go Robotaxi services in Beijing. <https://www.globaltimes.cn/content/1203174.shtml>.
- [6] BEVFusion-MIT Project Page. <https://github.com/mit-han-lab/bevfusion>.
- [7] BEVFusion-PKU Project Page. <https://github.com/ADLab-AutoDrive/BEVFusion>.
- [8] DeepInteraction Project Page. <https://github.com/fudan-zvg/DeepInteraction>.
- [9] Experimental Security Research of Tesla Autopilot. https://keenlab.tencent.com/en/whitepapers/Experimental_Security_Research_of_Tesla_Autopilot.pdf.
- [10] Hacker shows what Tesla Full Self-Driving's vision depth perception neural net can see. <https://electrek.co/2021/07/07/hacker-tesla-full-self-drivings-vision-depth-perception-neural-net-can-see/>.
- [11] Nusenes Object Detection Leaderboard. <https://www.nusenes.org/object-detection?externalData=all&mapData=all&modalities=Any>.
- [12] Tesla Autopilot Uses Transformer. <https://youtu.be/j0z4FweCy4M?t=3621>.
- [13] Transfusion Project Page. <https://github.com/XuyangBai/TransFusion>.
- [14] UVTR Project Page. <https://github.com/dvlab-research/UVTR>.
- [15] Waymo launches robotaxi service in San Francisco. <https://techcrunch.com/2021/08/24/waymo-launches-robotaxi-service-in-san-francisco/>.
- [16] Waymo Tech. <https://waymo.com/tech/>.
- [17] Mazen Abdelfattah, Kaiwen Yuan, Z Jane Wang, and Rabab Ward. Adversarial attacks on camera-lidar models for 3d car detection. In *IROS*, 2021.
- [18] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *ICML*, 2018.
- [19] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *CVPR*, 2022.
- [20] Adith Boloor, Karthik Garimella, Xin He, Christopher Gill, Yevgeniy Vorobeychik, and Xuan Zhang. Attacking vision-based perception in end-to-end autonomous driving models. *Journal of Systems Architecture*, 2020.
- [21] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.
- [22] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liang, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020.
- [23] Yulong Cao, S. Hrushikesh Bhupathiraju, Pirouz Naghavi, Takeshi Sugawara, Z. Morley Mao, and Sara Rampazzi. You can't see me: Physical removal attacks on lidar-based autonomous vehicles driving frameworks. In *USENIX Security*, 2023.
- [24] 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 *S&P*, 2021.
- [25] 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 *CCS*, 2019.
- [26] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.
- [27] Li Chen, Chonghao Sima, Yang Li, Zehan Zheng, Jiajie Xu, Xiangwei Geng, Hongyang Li, Conghui He, Jianping Shi, Yu Qiao, et al. Persformer: 3d lane detection via perspective transformer and the openlane benchmark. In *ECCV*, 2022.
- [28] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, 2017.
- [29] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. Futr3d: A unified sensor fusion framework for 3d detection. *arXiv preprint arXiv:2203.10642*, 2022.
- [30] Zhiyuan Cheng, James Liang, Hongjun Choi, Guanhong Tao, Zhiwei Cao, Dongfang Liu, and Xiangyu Zhang. Physical attack on monocular depth estimation with optimal adversarial patches. In *ECCV*, 2022.
- [31] Zhiyuan Cheng, James Chenhao Liang, Guanhong Tao, Dongfang Liu, and Xiangyu Zhang. Adversarial training of self-supervised monocular depth estimation against physical-world attacks. In *ICLR*, 2023.
- [32] Hongjun Choi, Zhiyuan Cheng, and Xiangyu Zhang. Rvplayer: Robotic vehicle forensics by replay with what-if reasoning. In *NDSS*, 2022.
- [33] Florian Drews, Di Feng, Florian Faion, Lars Rosenbaum, Michael Ulrich, and Claudius Gläser. Deepfusion: A robust and modular 3d object detector for lidars, cameras and radars. In *IROS*, 2022.
- [34] Ranjie Duan, Xingjun Ma, Yisen Wang, James Bailey, A Kai Qin, and Yun Yang. Adversarial camouflage: Hiding physical-world attacks with natural styles. In *CVPR*, 2020.
- [35] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv preprint arXiv:1608.00853*, 2016.
- [36] Ivan Evtimov, Kevin Eykholt, Earlene Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, and Dawn Song. Robust physical-world attacks on machine learning models. *arXiv preprint arXiv:1707.08945*, 2017.
- [37] Noa Garnett, Rafi Cohen, Tomer Pe'er, Roee Lahav, and Dan Levi. 3d-lanenet: end-to-end 3d multiple lane detection. In *CVPR*, 2019.
- [38] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

- [39] R Spencer Hallyburton, Yupei Liu, Yulong Cao, Z Morley Mao, and Miroslav Pajic. Security analysis of {Camera-LiDAR} fusion against {Black-Box} attacks on autonomous vehicles. In *USENIX Security*, 2022.
- [40] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [41] Yang Jiao, Zequn Jie, Shaoliang Chen, Jingjing Chen, Xiaolin Wei, Lin Ma, and Yu-Gang Jiang. Msmdfusion: Fusing lidar and camera at multiple scales with multi-depth seeds for 3d object detection. *arXiv preprint arXiv:2209.03102*, 2022.
- [42] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [43] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In *IROS*, 2018.
- [44] Ruturaj Kulkarni, Shruti Dhavalikar, and Sonal Bangar. Traffic light detection and recognition for self driving cars using deep learning. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018.
- [45] Hongyang Li, Chonghao Sima, Jifeng Dai, Wenhui Wang, Lewei Lu, Huijie Wang, Enze Xie, Zhiqi Li, Hamming Deng, Hao Tian, Xizhou Zhu, Li Chen, Yulu Gao, Xiangwei Geng, Jia Zeng, Yang Li, Jiazhi Yang, Xiaosong Jia, Bohan Yu, Yu Qiao, Dahua Lin, Si Liu, Junchi Yan, Jianping Shi, and Ping Luo. Delving into the devils of bird's-eye-view perception: A review, evaluation and recipe. 2022.
- [46] Ruihui Li, Xianzhi Li, Pheng-Ann Heng, and Chi-Wing Fu. Pointaugment: an auto-augmentation framework for point cloud classification. In *CVPR*, 2020.
- [47] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. In *NeurIPS*, 2022.
- [48] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. BEVFusion: A Simple and Robust LiDAR-Camera Fusion Framework. In *NeurIPS*, 2022.
- [49] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.
- [50] Zhijian Liu, Haotian Tang, Alexander Amini, Xingyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *ICRA*, 2023.
- [51] Dongyu Meng and Hao Chen. Magnet: a two-pronged defense against adversarial examples. In *CCS*, 2017.
- [52] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *CVPR*, 2018.
- [53] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017.
- [54] Takami Sato, Junjie Shen, Ningfei Wang, Yunhan Jia, Xue Lin, and Qi Alfred Chen. Dirty road can attack: Security of deep learning based automated lane centering under physical-world attack. In *USENIX Security 21*, 2021.
- [55] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017.
- [56] Dawn Song, Kevin Eykholt, Ivan Evtimov, Earlene Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, and Tadayoshi Kohno. Physical adversarial examples for object detectors. In *12th {USENIX} Workshop on Offensive Technologies ({WOOT})*, 2018.
- [57] Jiachen Sun Sun, Yulong Cao Cao, Qi Alfred Chen, and Z Morley Mao. Towards robust lidar-based perception in autonomous driving: General black-box adversarial sensor attack and countermeasures. In *USENIX Security*, 2020.
- [58] Kanglan Tang, Junjie Shen, and Qi Alfred Chen. Fooling perception via location: a case of region-of-interest attacks on traffic light detection in autonomous driving. In *NDSS Workshop on Automotive and Autonomous Vehicle Security (AutoSec)*, 2021.
- [59] 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. *arXiv preprint arXiv:2101.06784*, 2021.
- [60] 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 *CVPR*, 2020.
- [61] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017.
- [62] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *CVPR*, 2020.
- [63] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *CVPR Workshop*, 2020.
- [64] Li Wang, Xinyu Zhang, Wenyuan Qin, Xiaoyu Li, Lei Yang, Zhiwei Li, Lei Zhu, Hong Wang, Jun Li, and Huaping Liu. Camo-mot: Combined appearance-motion optimization for 3d multi-object tracking with camera-lidar fusion. *arXiv preprint arXiv:2209.02540*, 2022.
- [65] Zhixin Wang and Kui Jia. Frustum convnet: Sliding frustums to aggregate local point-wise features for amodal 3d object detection. In *IROS*, 2019.
- [66] Greg Welch, Gary Bishop, et al. An introduction to the kalman filter. 1995.
- [67] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.
- [68] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 2018.
- [69] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. In *NeurIPS*, 2022.
- [70] Jindi Zhang, Yang Lou, Jianping Wang, Kui Wu, Kejie Lu, and Xiaohua Jia. Evaluating adversarial attacks on driving safety in vision-based autonomous vehicles. *IEEE Internet of Things Journal*, 2021.
- [71] Qingzhao Zhang, Shengtuo Hu, Jiachen Sun, Qi Alfred Chen, and Z Morley Mao. On adversarial robustness of trajectory prediction for autonomous vehicles. In *CVPR*, 2022.
- [72] Yuchen Zhang and Percy Liang. Defending against whitebox adversarial attacks via randomized discretization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 2019.
- [73] Yue Zhao, Hong Zhu, Ruigang Liang, Qintao Shen, Shengzhi Zhang, and Kai Chen. Seeing isn't believing: Towards more robust adversarial attack against real world object detectors. In *CCS*, 2019.
- [74] Yiqi Zhong, Xianming Liu, Deming Zhai, Junjun Jiang, and Xiangyang Ji. Shadows can be dangerous: Stealthy and effective physical-world adversarial attack by natural phenomenon. In *CVPR*, 2022.
- [75] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *CVPR*, 2018.
- [76] Zhe Zhu, Dun Liang, Songhai Zhang, Xiaolei Huang, Baoli Li, and Shimin Hu. Traffic-sign detection and classification in the wild. In *CVPR*, 2016.
- [77] Mattia Zorzi. Robust kalman filtering under model perturbations. *IEEE Transactions on Automatic Control*, 2016.

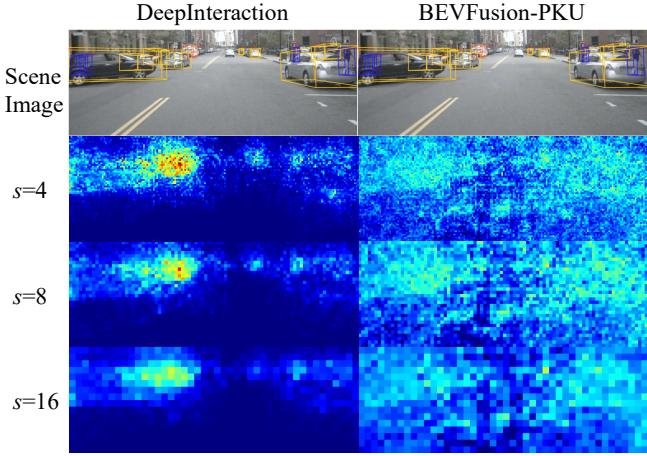


Figure 14: Sensitivity heatmap with **different granularity**.

Appendix

A. Model Selection Criteria

1) Representativeness: The selected models are the latest and most advanced fusion-based models for 3D object detection. Published in 2022, each model’s performance ranked top in the Nuscenes 3D object detection leaderboard [11]. Additionally, each model employs the Transformer architecture [61] as the detection head, which is widely recognized as a cutting-edge design in object detection models and has been adopted in Tesla Autopilot [12].

2) Practicality: The inputs to these models are multi-view images captured by six cameras surrounding a vehicle and the corresponding LiDAR point cloud collected by a 360-degree LiDAR sensor positioned on the vehicle’s roof. This configuration of sensors is representative of practical autonomous driving systems and provides a more comprehensive sensing capability when compared to models that rely solely on front cameras (e.g., KITTI dataset [38]).

3) Accessibility: All models are publicly available, ensuring that they can be easily accessed by researchers. The best-performing version of each model that utilizes camera-LiDAR fusion was selected and utilized in our experiments and can be found on their respective project repositories on GitHub [6]–[8], [13], [14].

Overall, these five models were selected as they provide a comprehensive and representative assessment of the latest advancements in camera-LiDAR fusion-based 3D object detection and were deemed practical, accessible and relevant in the context of autonomous driving applications.

B. Varying Granularity of Sensitivity heatmap.

Granularity of sensitivity heatmaps. Our sensitivity recognition algorithm employs the hyper-parameter s in Equation 6 to regulate the granularity of the mask M applied to the perturbation, which denotes the size of a unit area. In order to assess whether the distribution of sensitivity would be

influenced by variations in mask granularity, we perform experiments with diverse settings of s . By default, we establish s as 2, and the corresponding outcomes are illustrated in Figure 7. Outcomes derived from different granularity settings are depicted in Figure 14. We assign values of 4, 8, and 16 to s , and generate the sensitivity heatmap for DeepInteraction and BEVFusion-PKU. The first row in Figure 14 exhibits the original scene image and ground-truth objects. As the results indicate, the sensitivity distribution for a given scene remains generally consistent across differing granularity of masks, and the two types of sensitivity continue to exhibit unique attributes. DeepInteraction retains its object-sensitive nature, whereas BEVFusion-PKU persistently demonstrates global sensitivity.