Object Removal Attacks on LiDAR-based 3D Object Detectors

Zhongyuan Hau[§], Kenneth T. Co[§], Soteris Demetriou, Emil C. Lupu
Imperial College London
{zy.hau17, k.co, s.demetriou, e.c.lupu}@imperial.ac.uk

***Abstract*—LiDARs play a critical role in Autonomous Vehicles' (AVs) perception and their safe operations. Recent works have demonstrated that it is possible to spoof LiDAR return signals to elicit fake objects. In this work we demonstrate how the same physical capabilities can be used to mount a new, even more dangerous class of attacks, namely *Object Removal Attacks* (ORAs). ORAs aim to force 3D object detectors to fail. We leverage the default setting of LiDARs that record a single return signal per direction to perturb point clouds in the region of interest (RoI) of 3D objects. By injecting illegitimate points behind the target object, we effectively shift points away from the target objects' RoIs. Our initial results using a simple random point selection strategy show that the attack is effective in degrading the performance of commonly used 3D object detection models.**

I. Introduction

We are currently undergoing a revolution in transportation and mobility. New generations of vehicles are increasingly equipped with high-precision depth sensors to better perceive their environment and offer unprecedented levels of driver assistance and driving autonomy. Such vehicles commonly rely on LiDAR sensors, which collect high definition depth measurements stored in 3D point clouds. Reliably detecting objects from such point clouds is vital to the safety of the autonomous vehicle, its users and passengers.

LiDAR spoofing attacks. Recent studies have shown that it is possible to attack LiDAR-based perception systems by spoofing LiDAR return signals [6], [1], [10], [5]. Petit *et al.* first demonstrated this with physical attacks that can inject up to 10 fake 3D points in a point cloud. Cao *et al.* and Sun *et al.* progressively improved on the physical capabilities of the LiDAR spoofing adversary showing that one could reliably inject up to 60 and 200 fake points respectively. More importantly, Cao *et al.* developed a white-box model-level digitally simulated LiDAR spoofing attack that can introduce front-near fake measurements in a scene, which are then detected as objects by an end-to-end autonomous vehicle (AV) system. Sun *et al.* then demonstrated both white-box and black-box attacks that spoof vehicles in front-near locations by exploiting patterns of occluded and distant vehicles. Xiang *et al.* [13] examined the vulnerability of point-cloud based object detectors and proposed white-box approaches for point shifting and point injection to craft adversarial point-clouds. Zhao *et al.* [14] proposed a class of point cloud perturbation attacks that minimizes the number of points perturbed to flip the results of point-cloud based object detectors. They use gradient-based and genetic algorithm approaches to generate adversarial point clouds with perturbations of up to 150 points to subvert object detection with a 95% success rate.

Object hiding. Tu *et al.* [12] proposed both white-box and black-box methods to generate adversarial objects that when placed above a target vehicle, would evade point-cloud based object detectors with a success rate of 80%. For the white-box attack, the adversarial object is generated using a gradient-based approach to minimize the confidence score of the target object (vehicle). A black-box attack was also demonstrated, where the adversarial objects are chosen using a genetic algorithm approach to iterate and improve adversarial object meshes. Object-hiding attacks are considered more dangerous than spoofing objects. Whilst detecting a spoofed object can bring the ego-vehicle to a full stop, failing to detect an object has a higher chance of leading to a fatal collision.

Our work. We leverage the demonstrated state of the art capabilities of the physical LiDAR spoofing adversary [6], [10], [1] to design a new model-level *object removal attack* (ORA) that aims to hide objects from 3D object detectors. Compared to prior work with spoofed objects [6], [10], [1] ORAs have a different goal: the adversary aims not to introduce a ghost object but force mis-detection of a genuine object which can have more severe consequences. Moreover, in contrast with related work on 3D object hiding [12], we introduce a new technique that does not aim to introduce patterns on top of genuine objects, but rather to spoof points within a genuine object's bounding box such that they appear away from their original position and cause object mis-detection. ORAs are stealthier since they do not require placing adversarial objects on the target, are easier to mount and have high success rates. We conduct digital ORA attacks, emulating the physics of LiDAR operation. Their effectiveness is evaluated against popular 3D object detectors (PointRCNN [8] and Point-GNN [9]).

We found that an adversary with the ability to inject $\leq$ 200 points can reduce their *recall* to less than 25% for *Pedestrian* and *Cyclist* object classes on both models with a *random* point selection strategy. Our work demonstrates the feasibility of ORAs and we hope to inspire future work on more sophisticated ORA strategies that can lead to a better understanding of the LiDAR spoofing adversary model.

[§] Equal contribution

Workshop on Automotive and Autonomous Vehicle Security (AutoSec) 2021
25 February 2021, Virtual
ISBN 1-891562-68-1
https://dx.doi.org/10.14722/autosec.2021.23016
www.ndss-symposium.org

TABLE I. Average Precision (AP) of 3D object detection for different classes under the ORA random attack.

Model	Attack Budget	Car AP (IoU = 0.7)			Pedestrian AP (IoU = 0.5)			Cyclist AP (IoU = 0.5)		
		Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
PointRCNN [8]	0 (Clean)	88.86	78.61	77.75	62.87	54.88	48.95	73.08	56.33	52.36
	10	79.30	65.09	58.11	50.69	42.96	38.15	61.03	39.04	35.58
	20	78.85	59.57	51.00	48.06	40.38	35.42	54.66	31.67	30.12
	40	77.02	54.36	46.13	43.66	35.84	31.96	41.81	24.36	22.96
	60	72.97	48.01	40.42	38.70	31.97	27.82	33.54	19.14	18.64
	100	64.69	40.97	33.47	35.99	28.46	23.20	24.48	15.75	15.77
	150	55.74	34.05	28.33	29.04	22.79	19.93	17.01	11.95	11.33
	200	47.32	31.05	25.00	25.62	19.02	16.92	13.07	9.43	9.42
Point-GNN [9]	0 (Clean)	89.89	88.82	87.75	73.74	70.34	63.57	85.69	64.44	62.25
	10	89.62	78.33	69.31	73.74	70.25	63.49	85.69	64.17	62.01
	20	89.34	71.38	62.06	71.89	64.01	61.19	76.79	52.78	48.32
	40	86.40	62.50	53.08	63.18	55.57	48.28	57.00	36.70	34.85
	60	80.26	53.71	44.60	56.27	48.10	44.42	43.45	26.98	26.04
	100	70.47	43.89	35.36	47.15	39.24	32.50	25.63	16.06	15.86
	150	57.15	34.61	29.17	39.23	31.65	27.72	10.86	8.14	7.40
	200	45.58	28.01	22.01	31.99	26.64	23.81	4.76	3.66	3.85

II. Object Removal Attack

We introduce a new class of attacks, namely object removal attacks (ORAs). ORAs are black-box, model-level attacks that can be launched by a physical LiDAR spoofing adversary. The goal of ORAs is to displace the original depth measurements within a genuine object's bounding box to appear outside that bounding box such that the target object is not detected by 3D object detectors. Below we elaborate on our threat model and a preliminary strategy to demonstrate the feasibility of ORAs.

A. Threat Model

Physical Capabilities. We assume an adversary $\mathcal{A}$, who can spoof the LiDAR return signals of a target AV [1], [6], [10], [11]. The adversary can achieve this by deploying a device within the line of sight of a victim vehicle's LiDAR sensor. The adversarial device can capture LiDAR signals, alter them and emit them toward the victim sensor with a controlled delay. By controlling the return signal, $\mathcal{A}$ can manipulate the resulting 3D measurements reported in a 3D point cloud by the victim sensor. We assume $\mathcal{A}$ has state of the art sensor spoofing capabilities and can inject ≤ 200 points in a 3D scene within a horizontal angle of $10°$ [11]. We also assume that $\mathcal{A}$ can spoof resulting measurements to appear further away from the target vehicle than they actually are [10].

Digital Capabilities. Since $\mathcal{A}$ is in physical proximity, we assume it can also sense the environment and detect objects within the vicinity of the target vehicle. Using basic transformations, $\mathcal{A}$ can change the coordinate system of a 3D scene from the reference point of $\mathcal{A}$ to that of the target vehicle.

B. Object Removal Attack (ORA) Rationale

LiDAR spoofing has been previously demonstrated [1], [10], [11]. These attacks exploit the fact that LiDARs deployed in AVs operate under the Strongest Return Mode setting, where only a single measurement can be recorded per ray direction. Therefore, the injection of a spoofed point results in the original corresponding point in the direction of the ray (from the LiDAR to the spoofed point) to be displaced. We also leverage this phenomenon, but to achieve a different goal. Instead of aiming to introducing objects, we highlight a new class of attacks, namely *Object Removal Attacks (ORAs)* that aim to remove objects. Our proof of concept hides an object from AV perception by displacing points from a target object's point cloud with LiDAR point injection behind that object.

Algorithm 1 Obtaining Attack Trace (ORA-Random)

```
Input: list($obj\_pts\_coords$) ▷ Target Object's Point Cloud
$candidate\_pts\_coords$ =[]
$attack\_trace\_pts\_coords$ = []
for each pt in $obj\_pts\_coords$ do
    if pt within spoofing_horizontal_angle then
        $candidate\_pts\_coords \leftarrow$ pt
    end if
end for
$attack\_pts \leftarrow$ random($candidate\_pts\_coords$, $\mathcal{A}_{budget}$)
$attack\_trace\_pts\_coords \leftarrow$ ($pts$ in $obj\_pts\_coords$ $\wedge$ not in $attack\_pts$)
for each pt in $attack\_pt$ do
    $attack\_pt\_coords \leftarrow$ dist_increment_along_ray(pt)
    $attack\_trace\_pts\_coords$.append($attack\_pt\_coords$)
end for
return $attack\_trace\_pts\_coords$
```

C. ORA Operating Mechanism

An ORA exploits the LiDAR's default mode for recording the measurements of return signals–where a single return signal per ray direction is recorded. This enables an adversary to perform point injections that can remove points from a target object's original point-cloud by spoofing another signal at a different location in the same direction of the rays that are incident on an object. The resulting perturbed point-cloud would cause point-cloud based object detectors to miss detecting the target object, and thus evading object detection.

The process of ORA first starts with the adversary identifying the target object's location and the region of interest (RoI) where points are to be selected from and removed. Here, we assume that the adversary has knowledge of the 3D scene and is able to obtain bounding boxes of target objects and the coordinates of the points in the bounding boxes (i.e. object points). This can be achieved by finding a translation matrix that changes the coordinates system from the reference point of the attacker to the ego-vehicle [3]. Next the adversary obtains a set of points from the object points that are within its spoofing horizontal angle (i.e. candidate points); one way of achieving

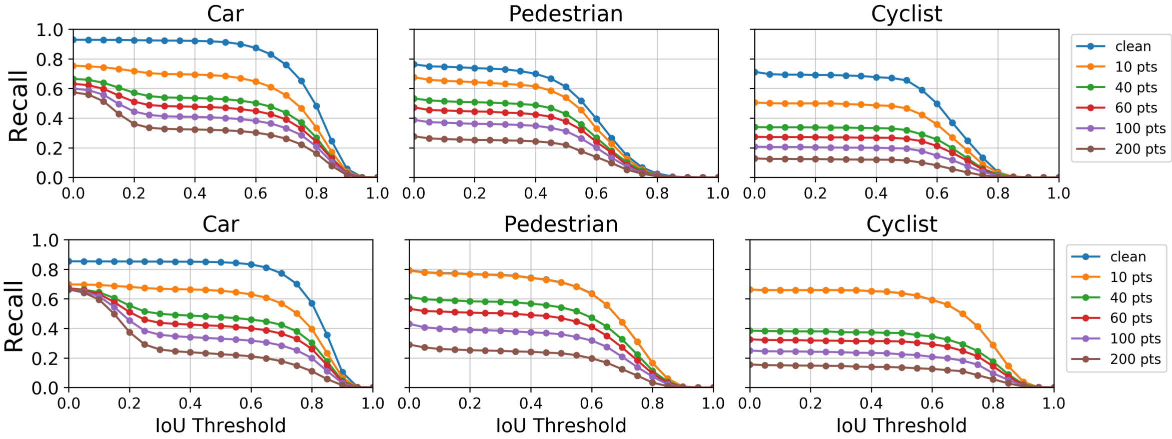

Fig. 1. Recall-IoU curves for (top) PointRCNN and (bottom) Point-GNN under the ORA random attack with different IoU thresholds.

this is to segment the object bounding box by the spoofing angle and only use the points within the segment. In our case, we used the left-most coordinates of the bounding box as an anchor point to calculate the points that are within the horizontal spoofing angle. Lastly from the subset of candidate points within the spoofing angle, the adversary picks points within its budget to perform point injection, spoofing points at a random distance behind the original points' location (in the direction of the ray). ORA is modular and can be used with various point selection strategies. We demonstrate ORA with a random point selection (*ORA-Random*) from the candidate points. *ORA-Random* is detailed in Algo. 1.

III. Experiments & Results

Models & Datasets. The proposed attack was conducted on the validation set (3769 out of 7481 scenes) of the KITTI dataset [4]. Objects in these scenes are subjected to *ORA-Random* and the resulting perturbed point clouds of the scenes are subsequently passed to popular 3D object detectors to evaluate the performance of the attack. We perform the attacks on three object types, *Cars*, *Pedestrians* and *Cyclists* as these objects are commonly encountered in AV scenarios. We perform our attack evaluations on widely-used models for 3D point cloud object detection that rely solely on LiDAR data, Point-GNN [9] and PointRCNN [8]. The two models differ in how feature extraction is performed for the object detection task. Point-GNN uses a graph neural network that encodes the point-cloud directly as a graph representation for object detection. Whereas, PointRCNN uses a multi-layer perceptron learning approach on point sets to obtain point feature vectors of the point cloud(PointNet++ [7]), which are then further processed for object detection.

Performance Metrics. For all models and scenarios, we measure the 3D AP and Recall-IOU curves of the models under attack. The 3D AP (average precision) captures the ratio of true positive predictions over all positive predictions and is the primary measure for overall performance of 3D object detectors. The Recall-IOU curve measures the recall of the detector for various IOU thresholds. The goal of the attacker would be to hide real objects from the model, so the measurement of recall is relevant in our case since it captures the ability of the detector to **not miss** objects. Therefore, $\mathcal{A}$'s goal would be to lower recall scores for target detectors.

Evaluation Scenarios. We consider two scenarios for evaluating the models' performance when under attack. The first is on the performance of the attack applied on the entire KITTI validation dataset. The second evaluation focuses on the impact of ORAs on the detection of *front-near* objects. For both scenarios, we use *ORA-Random* to perturb the point cloud of each individual type of target object found in the scenes with various point-perturbation budgets (10, 20, 40, 60, 100, 150, 200 points) that are within $\mathcal{A}$'s capabilities.

A. Attack Performance Evaluation

Table I shows the AP of the 2 models for clean (no point perturbation) and for the various attack budgets used to perturb the 3 object types. The evaluation criteria follows that of the KITTI 3D object detection benchmark (where the detection difficulty levels are determined by the size and occlusion of object). We observed that the AP decreases for increasing $\mathcal{A}_{budget}$. The effect of *ORA-Random* is most significant for *Cyclist* objects and then followed by *Pedestrian* and then *Car* objects. One reason could be that Cyclists are not as common in the dataset compared to the other two classes, resulting in poorer performance. *Pedestrian* and *Cyclist* objects are smaller objects and have significantly fewer points in their point clouds.

From Fig. 1, we observed that when increasing the point budget, the recall falls. For Cars at IoU $\geq$ 0.7 and for *Pedestrians* and *Cyclists* at IoU $\geq$ 0.5, we observe a significant decline in recall for both clean and attack, with recall falling below 0.5 for most of the attacks, with the exception of the 10-point attack on Point-GNN for pedestrian and cyclist. Our analysis on the KITTI validation set shows that the *ORA-Random attack is very effective in degrading the object detector's performance and hiding a target object.*

B. Attacking Front-Near Objects

We further investigate whether ORAs can mask front-near objects (objects in close proximity to the ego-vehicle), where accurate detection is critical to the safe operation of the autonomous vehicle. Our results are summarized in Fig. 2. We observe a general trend where objects further away from the

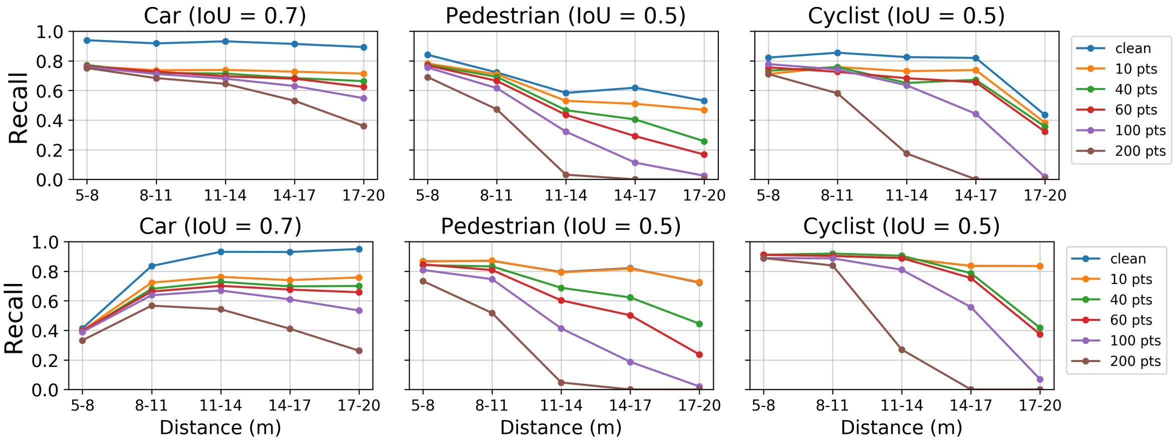

Fig. 2. Recall metrics by distance (in metres away from the LiDAR) for different IoU configurations of (top) PointRCNN and (bottom) Point-GNN.

LiDAR have lower recall but with the effects being visible even for objects $\leq 11m$. The extent of the drop in recall (w.r.t clean) is also correlated to the increase in $\mathcal{A}$'s budget. Noticeably for smaller objects such as *Pedestrian* and *Cyclist*, we observe a higher decrease in recall when increasing $\mathcal{A}_{budget}$ and distance. This is due to the smaller objects' inherent low number of points and its decreasing point density as distance increases. This provides an opportunity for the adversary to use its limited $\mathcal{A}_{budget}$ to perturb a larger proportion of points in the object's point cloud–increasing its success rate of evading detection.

C. Discussion & Implications

Although *ORA-Random* does not drastically damage the recall for front-near objects, it is still able to significantly lower recall and AP when considering the more general validation dataset. The detection of further away objects remains a critical function especially during the high-speed operation of vehicles. Thus being able to damage model recall in the general case with a random point selection strategy raises grave security concerns. Additionally, the random point selection can be improved upon with more optimized strategies that use methods such as genetic, evolutionary, or Bayesian algorithms [2], [12], [14] to create effective adversarial attacks within a point budget. Overall *ORA-Random* demonstrates that object removal attacks are a real concern which we plan to investigate in depth in future work.

IV. Conclusion

In this paper, we provide preliminary evidence that with a simple approach of shifting 3D points from a RoI, a LiDAR spoofing adversary is able to effectively perturb the point cloud of a target object to render it undetectable. We performed a sensitivity analysis and found that for smaller objects, the attacks are highly effective at a distance beyond 11m. This poses a safety concern as failure to detect such objects could have life-threatening consequences. In future work, we plan to implement optimization-based point selection ORA strategies, verify the feasibility of ORAs in the physical domain and study the effect of ORA at various distances and driving speeds on AV driving decisions with an AV simulator. As this new class of attack targets a single sensor modality, we are exploring defenses using multi-sensor fusion with RGB cameras.

References

[1] 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.

[2] Kenneth T. Co, Luis Muñoz González, Sixte de Maupeou, and Emil C. Lupu. Procedural noise adversarial examples for black-box attacks on deep convolutional networks. In *ACM Conference on Computer and Communications Security*, CCS '19, pages 275–289, 2019.

[3] Sundaram Ganapathy. Decomposition of transformation matrices for robot vision. *Pattern Recognition Letters*, 2(6):401–412, 1984.

[4] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[5] Zhongyuan Hau, Soteris Demetriou, Luis Muñoz-González, and Emil C. Lupu. Shadow-catcher: Looking into shadows to detect ghost objects in autonomous vehicle 3d sensing. *arXiv preprint arXiv:2008.12008*, 2021.

[6] Jonathan Petit, Bas Stottelaar, Michael Feiri, and Frank Kargl. Remote attacks on automated vehicles sensors: Experiments on camera and lidar. *Black Hat Europe*, 11:2015, 2015.

[7] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, pages 5099–5108, 2017.

[8] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–779, 2019.

[9] Weijing Shi and Raj Rajkumar. Point-gnn: Graph neural network for 3d object detection in a point cloud. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1711–1719, 2020.

[10] Hocheol Shin, Dohyun Kim, Yujin Kwon, and Yongdae Kim. Illusion and dazzle: Adversarial optical channel exploits against lidars for automotive applications. In *International Conference on Cryptographic Hardware and Embedded Systems*, pages 445–467. Springer, 2017.

[11] 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. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 877–894. USENIX Association, August 2020.

[12] James Tu, Mengye Ren, Siva Manivasagam, Ming Liang, Bin Yang, Richard Du, Frank Cheng, and Raquel Urtasun. Physically realizable adversarial examples for lidar object detection. *arXiv preprint arXiv:2004.00543*, 2020.

[13] Chong Xiang, Charles R Qi, and Bo Li. Generating 3d adversarial point clouds. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9136–9144, 2019.

[14] Yiren Zhao, Ilia Shumailov, Robert Mullins, and Ross Anderson. Nudge attacks on point-cloud dnns. *arXiv preprint arXiv:2011.11637*, 2020.