Crafting Adversarial Examples on 3D Object Detection Sensor Fusion Models

Won Park
University of Michigan
University of Michigan, Ann Arbor, MI, 48109, USA
wonpark@umich.edu

Qi Alfred Chen
UC Irvine
UC Irvine, Irvine, CA, 92697
alfchen@uci.edu

Z. Morley Mao
University of Michigan
University of Michigan, Ann Arbor, MI, 48109, USA
zmao@umich.edu

Abstract

A critical aspect of autonomous vehicles is the object detection stage, which is increasingly being performed with what are called sensor fusion models: 3D object detection models which take in both 2D RGB image data and 3D depth data (like from a LIDAR sensor) as inputs. However, while there has been lots of work on the performance of these models, their security, particularly against adversarial examples, has not yet been explored.

In this work, we perform the first preliminary study to analyze the robustness of a popular sensor fusion model architecture towards adversarial attacks. We find that despite the use of the 3D data, simply modifying the image via our raw-pixel attack is enough to fool the model and cause objects to disappear. We picked 28 random samples with 119 vehicles from the KITTI dataset and show that our raw pixel disappearance attack is able to generate successful adversarial examples against 133 of those images. We extend this attack and develop a modified algorithm to create generalizable adversarial patches that can fool multiple vehicles. To better understand this performance against adversarial examples, we run experiments that show the model learns to rely on the LIDAR input more than the image input, suggesting the image input can prove to be an "Achilles' heel" against adversarial examples.

1. Introduction

Autonomous vehicle manufacturers often use *sensor fusion* models to help the vehicles detect the environment around them. These types of models are 3D object detection models that take in two types of inputs: a 2D image from a camera and 3D depth data usually from a LIDAR sensor. With the growing proliferation of autonomous vehicles, their security is becoming more paramount, especially against adversarial examples.

It has long been known in the community that machine learning models are vulnerable to adversarial examples, maliciously crafted inputs designed to intentionally fool the model into outputting an erroneous result. Adversarial machine learning techniques have been applied extensively to create attacks on image classification and 2D object detection models. However, it is unclear how robust a 3D object detection model is to such techniques.

While recent work [6] has shown theoretically that models that take in multiple inputs are still vulnerable to potential perturbations in a single input, no one has actively explored the robustness and crafted adversarial examples against sensor fusion models. We design new techniques to craft adversarial examples on sensor fusion models. We then investigate some defenses and attempt to explain why the model is susceptible to these attacks.

The model we chose for our study is AVOD, [7] an open-source 3D object detection model that performs well on the KITTI benchmark. Furthermore, its two stage detector network architecture is one that is typical of sensor fusion models, making it an ideal candidate for our study. Our key contributions are as follows:

- We perform the first study of adversarial examples on sensor fusion models. We show that existing techniques for attacks on image classification and 2D object detection models are not directly transferable. We modify these attacks to show that sensor fusion models are vulnerable to adversarial attacks that modify just the image input. These attacks include the *raw pixel disappearance attack* which achieves a 94.92% accuracy in causing objects to disappear. We show that these adversarial examples are able to resist basic defenses.

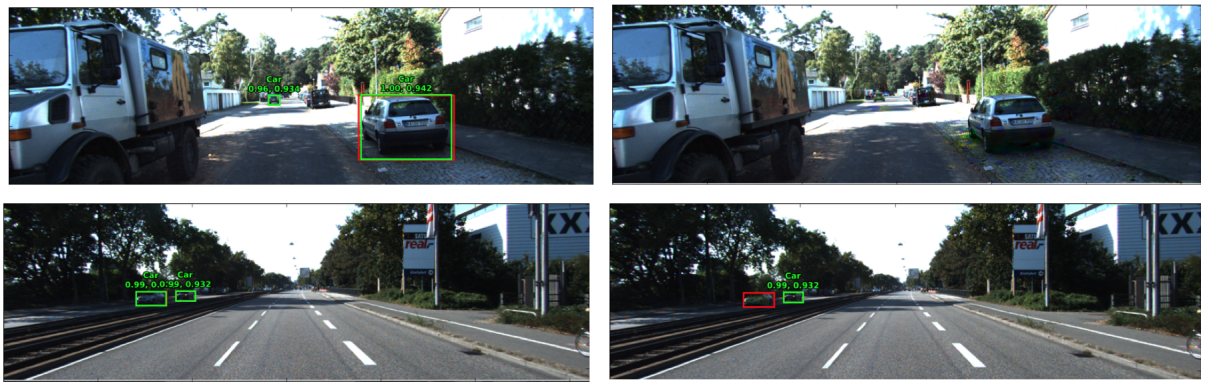

Figure 1. Results of some of our raw-pixel attacks. The left images are outputs for normal, benign images. The first value corresponds to the classification confidence and the second value corresponds to the IOU with the ground truth bounding box. The right images show the corresponding adversarial images. Note that our attack works in deleting any number of objects, whether they are in the foreground or background.

- We show that despite the symmetric architecture, the model frequently leans heavily on the LIDAR input to detect obstacles .

2. Related Work

2.1. Attacks on 2D Object Detection

There has been lots of work involved in 2D object detection models. These range from attacks in the raw pixel space [9] to launching these attacks in the physical world [4, 5, 11].

We draw inspiration from these techniques when attacking the 3D object detection model. However, some of this work is not directly transferable and so we modify the attacks to fit this model.

2.2. Attacks on 3D Object Detection

Existing work attacking 3D object detection models have largely been aimed at attacking models that solely use point cloud data [2, 8]. However, in the physical space it is more difficult to launch an attack to fool the LIDAR sensor. For this reason, we are motivated to look into attacks that solely modify the image, as as this type of attack is much easier for an adversary to perform.

3. Threat Model

We will assume that the adversary is a white-box attacker, having full access to the model. Despite this, we limit the adversary to modification of just the image for two reasons. First, we are better able to leverage existing work in the adversarial image space. Second, an attack through modification of just the image is much easier to carry out in the real world than an attack that requires modifications to the LIDAR sensor. Thus, by restricting attacks to just images, we are assuming a less powerful and more realistic attacker. Finally, we add another restriction that the adversary will not be able to modify the model in any way, including any post-processing steps, like non-maximum suppression (NMS).

4. Disappearance Attack

The first attack we developed, which we call the *raw-pixel disappearance attack*, tries to fool the model into not detecting an object it had previously detected. As stated in the threat model, we limit the adversary to modifying just the image. In order to make our attack more realistic, we have also disallowed the adversary from looking at any values before any post-processing steps.

To cause the desired object to disappear, we must remove all potential bounding boxes around said object. Removing a bounding box can be done by forcing the output softmax probability of an object to fall below the detection threshold.

We will call the set of all potential boxes that we need to attack B. More concretely, suppose we have image w and add adversarial noise δ. For ease of notation, suppose $C(w, b) \in R^c$ denotes the output classification logits of bounding box b on image w and $C(w)$ outputs the logits for all the potential bounding boxes of image w in decreasing order according to softmax score - in other words, the first element of $C(w)$ is the bounding box with the highest confidence. To minimize the score, we will then try to find

$$\text{argmin}_\delta \sum_{b \in B} C(w + \delta, b)$$

Since we wish to make the perturbation to the image as small as possible, we add another element as suggested by

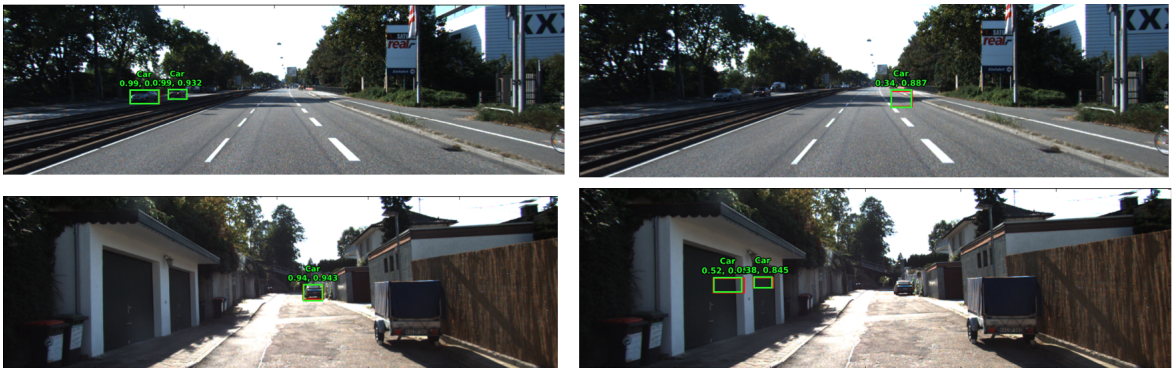

Figure 2. Output of experiment in which we switched LIDAR and image inputs. The two images on the left show the normal output for benign inputs. The two right images show the output when the LIDAR for one is switched for the other (and vice versa). Note that the model output follows the LIDAR more than the image.

the CW attack [3]: $D(w+\delta, w)$ which measures the L_2 norm between the adversarial image and the regular image.

Thus, the final loss function $L(.)$ that we are trying to minimize becomes:

$$L(w+\delta, B) = \epsilon * \sum_{b \in B} F(C(w+\delta, b)) + D(w+\delta, w)$$

$F(.)$ represents use of the softmax and ϵ is used to weigh one value versus the other. The optimal value of ϵ is found through binary search.

There lies an additional challenge in the fact that due to NMS and the restrictions we set on the adversary, not all of the elements of B will be visible. In other words, for some bounding boxes b, $C(w, b)$ is not existent and the logits will not be visible.

Algorithm 1: Raw-pixel attack

input : Raw image w, k
output: Adversarial noise δ
begin
  $\delta \leftarrow 0$
  while *Object is still detected* **do**
    $B' \leftarrow C(w+\delta)[0, ..., k]$;
    $\delta \leftarrow argmin_\delta L(w+\delta)$
  end
  return δ
end

To overcome this, we modify the algorithm to greedily attack the top confidence bounding box. The reasoning behind this algorithm is that as we keep trying to lower the confidence of the bounding box with the highest score, one of two outcomes will happen. In one, the object in question will no longer be detected, in which case our attack goal is accomplished. In the other case, the bounding box in question will be removed via NMS and the next top-score bounding box will appear and the process can be repeated.

This naive process will remove all objects present in an image, but an adversary can selectively remove certain objects by applying a mask. In this case, the objective function needs to be modified to attack the top k bounding boxes simultaneously. The full algorithm is shown as Algorithm 1.

4.1. Evaluation and Results

To test the attack, we utilized an instance of AVOD that identifies vehicles and trained to match the results stated in the original paper. We then choose 27 random samples containing a total of 119 detected objects and tried constructing adversarial examples using the method stated above. We are able to cause 113 objects to disappear, resulting in a 94.92% success rate.

To investigate if objects at different distances differ in the amount of distortion required, we create categories based on how far the vehicles are from the camera. Vehicles with an area of fewer than 3000 pixels are considered in the background, those containing between 3000 and 40000 pixels are considered in the middle ground, and any larger vehicles are considered foreground vehicles. Unsurprisingly, we find that vehicles in the foreground require more distortion than vehicles in the background. To normalize this, we divide the L2 norm by the area of the vehicle we are targeting. We find that there is not a statistically significant difference in the amount of distortion in this normalized L2 norm space needed to mount a successful attack for the different categories.

4.2. Towards Physical Realization

After the success of the raw pixel attack, we aimed to create an attack that is generalizable over many inputs. We drew inspiration from the expectation over transformation (EOT) algorithm [1]. In the case of the KITTI dataset how-

Experiment	mAP score (Std. Deviation)
Baseline	72.76 (0.61)
Mask Img (0)	59.52 (0.58)
Mask Img (125)	40.08 (0.51)
Mask Img (255)	25.11 (0.92)
Mask LIDAR (0)	0.002 (0.001)

Table 1. The mean AP scores on "moderate" difficult based on masking different inputs. Note that masking the LIDAR input has a much greater effect than masking the image.

ever, it is very difficult to apply any transformation to an image and also properly modify the corresponding LIDAR data. Furthermore, we also have to overcome the model's lack of support for batching. Therefore, we decided to use different object samples available in KITTI instead. We also modified the algorithm to run sequentially rather than in parallel. Preliminary results are promising; however, we are still running experiments to ensure the technique works for different images.

5. Defenses

In this section, we analyze the use of possible defenses against our attack. We consider in this section feature squeezing as a potential defense. This technique was first introduced by Xu et al [10]. In short, they propose a defense to "squeeze" the features of the image into a low-fidelity version. One of the methods proposed is bit reduction, in which the number of bits used to encode the image is reduced. We chose this defense because it is a simple add-on defense that does not harm accuracy too much. While the original model has a 3D AP score of 73, we find that the extreme act of squeezing the image input into just 2 bits drops the score to 67.65. A similar result was found when we trained two new versions of the model. We believe the reason for this is that the model has learned to use LIDAR input more heavily than the image input. We explore this further in Section 6.

To consider if this defense is effective, we applied this defense on all the adversarial examples constructed in Section 4. We consider the adversarial example to be stopped if the vehicle is identified with over a 10% confidence and with an IOU of over 0.5 with the ground truth. We find that the defense is able to recover only 46 out of 81 adversarial examples (a 56.79% rate). If the objects are recovered however, the outputted bounding box is often correctly located.

6. Analysis of Sensor Input

Motivated by the results of our experiments on the attacks and defenses, we suspect that the model architecture, while symmetrical, heavily utilizes the LIDAR sensor input over the image. To test this, we ran an ablation study on five models in which we masked out one input and then another. To mask the image, we ran each instance of the model three times, filling the image with a value of 0, 125, or 255. To mask the LIDAR, we utilized a value of 0. The mean AP scores for the various experiments are shown in Table 1. The LIDAR drops performance of the model far more than masking the image, suggesting that the LIDAR is utilized more heavily than the image.

6.1. Switching Inputs

For the second set of experiments, we randomly used the LIDAR from one sample and the image for another. This was done for 30 random samples, swapping the image of one and the LIDAR of another. This experiment helps give an insight of how the model performs when the image and the LIDAR are at odds with each other. Some of the results are shown in Figure 2.

For the sake of simplicity, we considered a sample as "correct" if the bounding box was correctly drawn according to the LIDAR sensor. We find that out of 107 potential objects, 81 were identified correctly despite having conflicting images. Among the 30 samples, there were 7 additional bounding boxes that did not correspond to any LIDAR bounding box. However, among these, only 2 were drawn correctly around the vehicle in the image space.

All this suggests that the use of image in this architecture proves to be an "Achilles' heel". While most of the detection of an object is done using the LIDAR input, a well-crafted image input can override this, thus providing an avenue for adversaries to attack and fool the model.

7. Conclusion

In this work we explore sensor fusion models' security against adversarial examples. We pick a popular architecture and craft adversarial examples on the image input that can cause objects to disappear. We also show that the 3D depth data input is more heavily used in the model than the image input.

This also helps explain our preliminary findings that suggest making an obstacle appear is more difficult than causing it to disappear. For the future, we are currently evaluating all results on different iterations of the model to strengthen our findings, including testing for how well our adversarial examples transfer from one model to another. We are also investigating algorithms to create more robust adversarial examples that are able to fool different orientations of cars and can lead to physical real-world attacks. Finally, we are designing techniques to attempt to attack a black-box model in which no internal workings of the model are known to the adversary.

References

[1] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples, 2017.

[2] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Zhuoqing Morley Mao. Adversarial Sensor Attack on LiDAR-based Perception in Autonomous Driving. In *Proceedings of the 26th ACM Conference on Computer and Communications Security (CCS'19)*, London, UK, November 2019.

[3] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. *2017 IEEE Symposium on Security and Privacy (SP)*, May 2017.

[4] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramèr, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Physical adversarial examples for object detectors. In *Proceedings of the 12th USENIX Conference on Offensive Technologies*, WOOT'18, page 1, USA, 2018. USENIX Association.

[5] Lifeng Huang, Chengying Gao, Yuyin Zhou, Changqing Zou, Cihang Xie, Alan Yuille, and Ning Liu. Upc: Learning universal physical camouflage attacks on object detectors, 2019.

[6] Taewan Kim and Joydeep Ghosh. On single source robustness in deep fusion models. In *NeurIPS*, 2019.

[7] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven Lake Waslander. Joint 3d proposal generation and object detection from view aggregation. *CoRR*, abs/1712.02294, 2017.

[8] Yuxin Wen, Jiehong Lin, Ke Chen, and Kui Jia. Geometry-aware generation of adversarial and cooperative point clouds. *CoRR*, abs/1912.11171, 2019.

[9] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan Yuille. Adversarial examples for semantic segmentation and object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[10] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*, 2018.

[11] 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 *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, page 1989–2004, New York, NY, USA, 2019. Association for Computing Machinery.