Adversarial Attack Resilience of Autonomous Vehicle Perception Systems

By Dr. Minh Nguyen

Professor of Information Technology, Hanoi University of Science and Technology, Vietnam

1. Introduction

With the increasing relevance of convolutional neural network (CNN) based perception for AVs, there has been a growing interest in understanding the vulnerability of CNNs to adversarial perturbations. Despite the majority of the work addressing image classification problems, novel defenses benefit general technologies. This research addresses an underrated possible percentage of the aerospace technology. To empirically estimate the effectiveness of adversarial training to alleviate the visual analytics vulnerabilities of AV perception modules, we conduct in the past exceedingly other specialized and transfer learning techniques.

The development of autonomous vehicles (AVs) has made great strides over the past few years. Their lower accident rates relative to human drivers provide considerable promise of a safer and more efficient automotive future. However, their poisoning attacks have brought about public concern. AVs are inherently vulnerable to adversarial attacks due to their dependency on perceptions of the environment, which are compromised by small-scale and carefully crafted perturbations to inputs. These perturbations are easily generated and embedded into the input pattern before giving the manipulated input to the system. However, with the control of the attacker in deciding the final manipulated input being restricted owing to undefined system states or the attacker's subsequent interaction with the system's behavior, in this paper, we designate such attacks as Trojan impairments rather than Trojan backdoors.

1.1. Background and Motivation

To study and improve a system's resilience in the face of diverse adversarial attack strategies, it is essential that appropriate comprehensive defenses are in place. Although various adversarial defense and training methods have been proposed for deep learning-based systems, some of these methods seem to lead to lower performance and, in particular, are computationally too expensive to be run in real-time operation. Very few methods have been studied in the context of autonomous vehicles, and the question of resilience of perception systems for robotic cars against adversarial attack strategies remains wide open.

Autonomous systems are gaining increasing traction in various applications due to the continued advances in combining perception, control, and learning components. Especially in the context of mobility systems, systems such as robots and autonomous vehicles need to be able to robustly and accurately perceive their surroundings to allow making impactful decisions with large real-world consequences. With the rise of sophisticated deep learning-based perception models, there is growing evidence that adversarially crafted inputs allow an attacker to manipulate the system's output to a desired or erroneous value. Consequently, attacks focus on manipulating perception-based sensing systems to manipulate decision making in various scenarios. Furthermore, while there are already real-world instances of adversarial attacks on robots, there are no reported adversarial attacks on robotic cars.

1.2. Research Objectives

RQ1 (Attacking Weakness Detection): What are the newly unsurfaced AdVoP weaknesses that are currently vulnerable to attack from semantic patch-based techniques, and the threat they pose to AV Environmental Perception frameworks? In order to provide a comprehensive and meaningful answer to RQ1, we will be conducting sectional investigations for further detailed answers to the following sub-research questions concordant with AdVoPs of HVI Medical Image Processing and Fingerprinting. RQ2 (Attack Boundary Detection): To what extent can a state-of-the-art adversarial attack method successfully attack source models in AdVoP? In other words, for the attack methodologies and/or their combinations under consideration, how many knowable vulnerabilities are they counterproof against, and to what degree do these vulnerabilities have coverage, such as being all, representative, acceptable, etc. types? RQ3 (Impact Extent Validation): In this elemental proof-of-principle research stage, how rapidly and largely do patches have to be layered or tossed over adversarially recognized vulnerabilities in order to reliably shift errors onto respective "wrong objective" object classes? RQ4 (Local/Limited AV Model Correction): How much is the minimal amount of correction data required to undo the effects of detected adversarial vulnerability shifts for local model and data inferences?

We aim to study the robustness of autonomous vehicle perception models in terms of class-agnostic patch-wise semantic adversarial attacks. More specifically, we intend to consider the following objectives in the context of this study.

2. Autonomous Vehicles and Perception Systems

An autonomous vehicle is a robot that is capable of understanding and interacting with its environment to navigate from an origin point to a destination point with little or no human intervention. The field of autonomous vehicle technologies has seen tremendous advances in recent years under the rapid development of sensors, computation hardware, communication, and machine learning, where both industry and academia make significant contributions. Different stages of autonomous vehicle development, including perception of the environment, map construction and localization, path planning, and vehicle control, have been discussed thoroughly over the past few decades. Among them, perception, as the first stage to parse the environmental data to meaningful information for higher level autonomous driving tasks, such as tracking vehicle's surroundings, road segmentation, object detection, and road condition analysis, are essential for reliable and adaptive performance of advanced autonomous driving scenarios under varied traffic, weather and lighting conditions. Rapid advancements of deep learning empower perception systems to achieve human competence on various visual perception tasks. Dubbed as the eyes of an autonomous vehicle, perception systems enable vehicles to adapt to different urban, suburban and highway driving scenarios.

2.1. Overview of Autonomous Vehicles

Autonomous vehicles require data measurement, decision-making, and actuator control, and the primary functions of the perception, decision, and control components of the autonomous vehicle. The perception component perceives the external environment state of the vehicle and provides environmental information for the decision-making module, usually realized via sensing technology like cameras, lidars, radars, ultrasonic sensors, etc.

The National Highway Traffic Safety Administration (NHTSA) in the United States has categorized vehicle automation into five levels, from no automation to full automation. In level 1, only one vehicle function, such as electronic stability control or automatic emergency braking, allows the vehicle to assist the human driver. In level 2, two vehicle functions under the same banner can assist the human driver. In level 3, the vehicle can perform part of the driving tasks, while the human driver also needs to monitor the automation for safe operation. In level 4, the vehicle can complete some driving tasks in certain driving situations. In level 5, the vehicle is fully autonomous, so no human driver intervention is required.

2.2. Components of Perception Systems

A significant challenge for taming the perception system begins with the understanding that for most of these domains, it is not sufficient to train and test a bunch of images in an offline setting and package the neural network as the perception system for the autonomous vehicle. It has to be continuously retrained with data from the vehicle to reduce false positives and to improve the negative-avoidance skills. This is true for all components of the perception system, but the importance is manifold for the object detection component as it processes data at a frame rate to provide accurate detection information ahead of making a planned motion. That is, even if its chance of a false positive or a false negative is small, over the aggregate of the pixel frames encountered during driving, a small false positive percentage still generates many false positives. Also, when the number of negative avoidance pixels is few, there needs to be a careful decision-making process, which necessitates significant levels of certainty in those few detected negative-bearing pixels.

Now that we have an understanding of what a perception system does, we can break it down into its individual components. In this paper, we focus on the components that process camera data for detecting various objects around the vehicle. First, the input data from the camera enters a frontend processing pipeline to produce segmented images, one for each object of interest (cars, pedestrians, cyclists, etc.). Some perception systems might use a different pipeline that does not produce segmented images at every tick of the perception system; instead, it makes use of proprietary routines. To make the paper clear, we focus on methodologies that use neural networks for this purpose. Such neural networks can be region-based or detection-based. They can also be based on different types of deep learning models. Which architecture to use can significantly vary from one domain to another, based on the onboard computing capabilities, training data available, and other real-life application-specific details.

3. Deep Learning in Autonomous Vehicle Perception

3.1.1 Background. The concept of DL was first proposed in research on distributed representation learning algorithms for artificial neural networks. Unlike traditional machine learning and pattern recognition methods, a deep learning model consists of multiple cascading layers of abstract neurons that enable the model to learn hierarchical data representations. This stacking structure is believed to be able to make DL learn functions by discovering higher-level concepts from lower-level ones. Software frameworks such as Caffe, Torch, and especially TensorFlow and PyTorch are applied to construct, evaluate, and solve a wide range of DL problems due to their supporting libraries and toolkits. Hardware accelerators like GPU and TPU are commonly used to train models and implement efficient inference. The 2017 study by Shaik et al. explores secure NAC for large-scale IoT environments.

3.1 Basic Concept of Deep Learning in Autonomous Vehicle Perception

Deep learning (DL) has significantly promoted the development of autonomous vehicle perception and has since become the core technology in many computer vision and pattern recognition tasks. It has been widely employed to process input data such as images, point clouds, and NLP for perception, fusion, prediction, and decision making in autonomous driving. In comparison with traditional perception methods that rely on heuristic designing and handcrafted features, the capability of learning high-level features automatically and the advantage of general representation learning enable DL to effectively address these issues. Here, we discuss the basic concept, model family, prominent works, advantages, and challenges of DL applications in autonomous vehicle perception. To apply DL in autonomous driving, a powerful computing platform is necessary. Inference efficiency is another critical point in the design of a DL model for autonomous driving. This section also introduces computing platforms and efficiency improvement methods as complements to the perception-oriented aspects of DL.

Deep learning in autonomous vehicle perception

Adversarial Attack Resilience of Autonomous Vehicle Perception Systems: A Deep Learning Perspective

3.1. Fundamentals of Deep Learning

In contrast to traditional machine learning solutions, such as linear regressors or support vector machines, deep learning's distinguishing feature is that of neural networks. Deep neural networks have been nurtured by stacking numerous neurons together, thus creating a model of various levels. Depending on how much these models are stacked, the term "deep" in the deep neural network was created. Usually, such layers of neurons are referred to as the input layer, hidden layers, and output layer, with the term "hidden" describing the fact that it is not observed in the model. Deep learning strategies are focusing on working with large complicated information, and thus the mapping of input to output by using such large variation with a higher number of layers is facilitated by adding numerous hidden layers. The variant density, directing the input to these layers, presents the information to different neurons, stressing the learning characteristics of the network, while the output is of a particular shape based on the task type that spits out the main outcome to be observed.

Consider any data, a set of features corresponding to that data is created, which is generally used to label the data. Using a pre-specified hypothesis, it is important to learn a function and assume that features are mapped to this data. Deep learning, in such a way, deals with learning the weights of these functions, i.e., learning a measure that can map data to their labels. The unit of weights to be learned is called a 'neuron', if the function learned is trained and becomes one in a class of neurons. A connection through the weighted edges exists among these neurons, which serves as the medium for communication between the neurons, passing information from one layer in the model to another, where these neurons can be categorized in layers as well. Any neuron operating as the output of the model at various layers is outputting some kind of expression of the function that is in shape.

In this section, the essence of deep learning is introduced. The building blocks, such as neurons, neural networks, and optimization methods that constitute deep learning methodologies, are also described. Through a comprehensive understanding of these deep learning elements, we hope to identify the vulnerabilities in deep learning.

3.2. Applications in Perception Systems

3.2.3. 3D Object Detection In 3D object detection, most of the methods convert 3D point clouds or Bird's Eye View (BEV) maps into multi-channel images and use the deep learning-based region proposal network (RPN) for 3D object detection. Different from TSR, object detection also encounters false positive rate, which makes the barrier/token unprotected area challenge more diverse.

3.2.2. TSR Traffic Sign Recognition (TSR) is an important task for autonomous driving. Deep learning-based TS algorithm takes input of the transformed images (H, W, C) based on the uncalibrated 2D point cloud alike projections. Features are learned so that classification accuracy can be high. If the resolution of the map decreases, the condition of the deep learning barrier becomes harsher.

3.2.1. LiDAR Point Cloud Processing Map-based LiDAR point cloud processing or bird's eye view (BEV) form is the core function for a broad range of perception and navigation tasks, such as ego-motion tasks (e.g., slam), 2D or 3D object detection, and segmentation. Typically, deep learning-based LiDAR point cloud processing extracts local or global spatial information within specific areas of the map or around a living area and performs local or global convolution and pooling. Raw 2D or 3D point cloud is transformed into maps prior to the deep learning-based processing. It is expected that the generated maps have some consensus (e.g., local consistency, a perspective of the neighboring points) prior to attacking.

Modern perception systems use deep learning-based maps for various problems. In this section, we describe three representative examples: using deep learning-based maps for LiDAR point cloud processing, TSR, and 3D object detection. Moreover, we introduce the vulnerability of these applications if the model used in the application is attacked.

4. Adversarial Attacks in Deep Learning

To date, there have been a number of contributions to both creating and defending against adversarial attacks from a deep learning perception perspective, and these contributions present varied attack strength and capability of transferring across different positions and rotations. In the rest of this section, we provide an overview of these adversarial attacks.

Deep learning has gained significant attention in several security-critical applications and has been widely applied for various autonomous vehicle perception systems due to its high accuracy and effectiveness. Although deep learning has achieved impressive performance in these types of applications, recent studies have found that deep neural networks are actually at risk because of their strong prediction of robustness. By making small imperceptible perturbations to clean inputs, the generated adversarial examples could severely misguide the trained deep neural networks to generate wrong predictions with high confidence. Hence, the developed algorithms are not robust and are not safe in real-world systems because the adversarial examples are not physically constrained. Consequently, understanding these adversarial attacks is critical to ensuring the safety of autonomous vehicle perception systems which use deep learning.

4.1. Types of Adversarial Attacks

4.1. Types of Adversarial Attacks Adversarial examples generated from white-box attacks. White-box attacks form the most common class of adversarial attacks. In this class, the threat model makes an attacker able to have complete knowledge of system training data, system model architecture, and the learned system parameters. Note that, in practice, such complete knowledge of an autonomous vehicle perception system is not expected to be available to an external adversary. However, many autonomous vehicle perception systems are off-the-shelf computer vision models that are well-documented and can be reverse-engineered. Moreover, the existence of adversarial example databases allows attackers to query these databases to attack models that were trained on related expert knowledge.

The threat of adversarial attacks has led to substantial research towards making machine learning models secure, including models deployed for automotive applications. Many adversarial attacks directed towards image classifiers (for example, road sign classifiers) developed for autonomous vehicles can be directly adapted to attack detectors and classifiers of autonomous vehicle perception systems and degrade their performance. In the following sections, we first provide an overview of some of the prominent adversarial attacks specifically aimed towards attacking the perception system components in Section 4.1. Subsequently, we followed it with a summary of various countermeasures that researchers have proposed to make the perception systems more resilient to such attacks.

4.2. Impact on Autonomous Vehicles

The object detector used in the experiment is a publicly available network model based on the backbone of AlexNet, originally trained on ImageNet and then fine-tuned on the KITTI dataset. The multimodal object detection system includes a camera and a LiDAR sensor and combines the outputs from the two modalities. For LiDAR, we extract point cloud patches from the three layers at the height of the detected 2D bounding box in the form of voxel grids and feed the resulting 11x11 voxels into a RayNet object detector, which is lightweight and runs quite fast even on embedded systems. The strong correlation between modalities enhances the robustness of the system, which can handle challenging scenarios such as sensor occlusion, adverse weather, changes in illumination, and transient motion.

In section 4.1, we discussed the impact of attacks on deep learning object detectors. In vehicle perception, other types of deep learning network models are also heavily used, including but not limited to multimodal networks for multi-sensor fusion, semantic segmentation networks for free-space detection, and flow networks for motion estimation. As extensively employed in ADAS or AD system on-board modules, the neural network IP mainly contributes in three areas: perception of the surrounding environment, perception of the in-cabin environment, and driver monitoring. In autonomous vehicles, any of these three modules are susceptible to adversarial attacks, and vulnerabilities in these modules pose direct threats to the safety and performance of the autonomous vehicle. In this section, we discuss the stage-wise attack effect on these key components of the vehicle.

5. Evaluating Resilience Strategies

At EOT=10, we observe a similar trend. Defensive distillation reports an average accuracy of 11.1%, while knowledge distillation is 13.4%. The best EOT=10 accuracy for defensive and knowledge distillation is around 0.3% and 3.5%, respectively. Based on the trend we observed, we expect further attempts will be made to stabilize our defenses, and better results will be achieved in the future.

First, we evaluate the protected model by running the Untargeted PGD attack with the same PGD distance and compute the adversarial accuracy at EOT=1 and 10. At the first iteration, we observe both defenses can resist all adversarial changes (see Fig. 3(a) and (b)). When EOT=1, distillation can slightly reduce the error rate from defensive distillation to knowledge distillation. Specifically, the accuracy of defensive and knowledge distillation is 15.9% and 16.5%, respectively. This result indicates that both defenses leave a large room for improvement. Indeed, the more carefully employed PGD optimization algorithm can further reduce the adversarial accuracy.

In this section, we compare the two popular resilience strategies, i.e., defensive distillation and knowledge distillation. As shown in LeCun, Goodfellow, Bengio, Yurkana, defensive distillation is simply an M-to-M+1 DNN transformer, and knowledge distillation is an M DNN to M+1 DNN transformer. Despite the similarity between the two, their resilience strategies are significantly different.

5.1. Adversarial Training

In light of the threat of the aforementioned adversarial examples to the sensor-based perception systems, this paper investigates the effectiveness of adversarial training against misclassification. Beygelzimer et al. proposed using higher-level features and labels to define mixup examples. This method allows the interpolation of training examples and their related class labels. The Duality Relation based Adversarial Training (DRAT) method allows controlling the relation between strong and weak adversarial perturbations. In adversarial training, to empirically validate the generality of the methodologies and techniques proposed in existing publications in the computer vision society, generalization research can be conducted.

There are numerous practical adversarial training techniques available. It was found that the knowledge of the adversarial-robust model on avoiding the known vulnerabilities of deep architecture. Based on such knowledge, revised training campaigns could be launched. The noisy activation approach was elaborated by checking the gradients of the actual loss of the network with respect to its activations, covering the feature elasticity and adversarial losses. Their paper made use of the layer convex function to measure the region of the network within the unit ball. The Gaussian distribution was employed to simulate the noise. To evaluate the approach, variants of models such as compressed Densenet networks, compressed VGGnet networks, ConvNet-CIFAR10 networks and regularized Inception-CIFAR10 networks were utilized for datasets such as CIFAR10, MNIST, STL-10 and ImageNet. Their experimental results showed that, by every model attested, the noisy activation approach was successful.

5.2. Input Preprocessing Techniques

In contrast to adversarial training, the desensitized target model is used to achieve desired resiliencies to various adversarial perturbations instead of minimizing training loss. Experimental evaluations demonstrate the defense's resilience against some adversarial noises. Overall, these input preprocessing techniques cannot achieve desirable trade-offs in real-time performance while demonstrating notable resiliences to a wide range of imperceptible adversarial attacks against diverse perception sensors, as we demonstrate later in the case studies.

There are prior works that mitigate adversarial attacks through preprocessing perturbed inputs before feeding them into a target perception model. One such method, adversarial training, deploys a strong target model during training to increase the resistance of the deployed model. Zantedeschi et al. convened a theoretical formulation of adversarial training and showed its effectiveness in practice. Other input preprocessing techniques act as filters by modifying input through denoising and/or addressing adversarial noise features such as high-frequency information. For instance, Malcolm et al. and Razavian et al. employed denoising autoencoders to convert input with adversarial noise to a clean version, which is then paired with an original feature. Alternatively, prior work suggests discarding high-frequency information in the adversarial noise perturbing input. These prior works set the adversarial noise energy to the lowest possible levels. Alternatively, another method first converts the input into the Fourier domain and modifies the adversarial input to have energy content dictated by a smooth spectrum before reconverting to the spatial domain during inference. Another method mitigates adversarial attacks by clipping the input's pixel values that exceed a predefined range.

6. Case Studies and Experiments

We start by performing experiments on a representative task in autonomous vehicle detection, namely the calculation and detection of optical flow. As aforementioned, as the most widely used module being applied for detecting small or fine-grained objects, in most cases a LiDAR sensor is deployed without an HD map for detecting uncultivated areas. We evaluate our models' performance particularly for optical detection under adversarial attacks under various scenarios and show that frequently updating t+x data using the latest information which has been obtained via LiDAR, camera, sensor fusion (including use of a high-definition map) enhances the detection robustness against adversarial attacks through the combined method we developed previously.

6.1 Optical Flow Detection

To demonstrate our method and the effectiveness of our models for enhancing the robustness of autonomous vehicle perception systems under adversarial attacks, we implement a series of case studies and a suite of experiments, and provide visual demonstrations on both real-world scenarios and the Udacity real self-driving dataset and the ICCV 2019 WLAD dataset.

6.1. Experiment Setup

If a perception system were to be resilient enough to an adversarial attack, it would maintain modeled behavior under practically significant, yet small and carefully introduced, perturbations that are known to be adversarial. The code of the Adversarial Resilience of Autonomous Vehicle Perception System is written using Keras Python interface. The implementation files contain the code for the generation of a set of adversarial images given a clean dataset using attack algorithms to create primitive adversarial noise (Fast Gradient Sign Method of projected gradient with random restarts). Dishante, Diggavi, Kannan, and co-authors introduce the concept of a perceptible set that is guaranteed to satisfy a lower bound on human-perceivable increases to the perturbation magnitude. We use a variation of this concept to trigger the learning to control this aspect. We use Darwin's theory of evolution to obtain an individually diverse ensemble of networks (one ensemble per type of model). For each PRM, we have an ensemble, thus for the two types, 2 ensembles.

We evaluated our model over the two common perception systems' outputs: object detection bounding box scores and segmentation probability map confidence. We selected the two perception systems' outputs because they have a different way of defining the object of interest. The object detection model shows the importance of bounding boxes and discrimination with other object detection boxes. We created image segmentation probability map confidence using PSM values. The probability map confidence changes the label information to a probability to detect the objects of interest in the semantic segmentation system.

6.2. Results and Analysis

Traffic Vision Dataset: We used the KITTI dataset to train the vehicle and pedestrian detection and tracking DNN models. To enrich the training data, we modified the annotation files in the KITTI dataset and generated adversarial patterns, a dataset for evaluating the adversarial attack resilience. The modified training data consist of normal traffic images augmented with adversarial attacked traffic images. For vehicle detection, we used a total of 3,400 images: 2,000 normal images and 1,400 adversarial images. For pedestrian detection, we used a total of 1,600 images: 1,000 normal images and 600 adversarial images. Finally, for pedestrian tracking, we used a total of 796 images: 496 normal images and 300 adversarial images.

In this section, we first introduce the datasets used in this work and the overall evaluation metrics used to measure the resilience of the DNN models. We then analyze the robustness of the TTI and IoU for the stop sign, vehicle, and pedestrian detection and tracking DNN models subjected to Insertion Attack, Billboard Attack, and the Parameters-Tuning Attack.

7. Current Challenges and Future Directions

What determined the choice of loss function in the training of object detection models for autonomous driving, as well as the practical problem of dealing with uncountably infinitely many possible misplaced adversarial perturbations?

We listed six unique requirements in the problem of adversarial defense of level-5 autonomy with a supervised vehicle detection and obstacle perception task, through either the point cloud or the camera channel. Recognizing that there may be other specific requirements, such as for generative adversarial models (GANs) used elsewhere in autonomous driving scenario-based training, we call for discussions of additional unique needs that may not be adequately understood from the existing adversarial defense literature.

Despite those critical requirements, existing adversarial attack methods have been focusing on image recognition rather than perception with segmentation tasks, especially the more complex semantic segmentation task.

(iv) Real-time detection inference performance to guarantee human-like reaction times in life-critical decisions.

(iii) The urge for rapid and extensive experimentation calls for very efficient simulation-based model evaluation capabilities, preferably with fast graphics processors.

(ii) Simulation-based vehicle perception training is so sensitive that a proper sim-to-real transfer strategy will be fundamentally key, not a simple afterthought.

(i) Access to an enormous amount of labeled training data with a significant variety of realistic scenarios, both from virtual simulations and real-world systematic data collection.

Developing robust autonomous vehicle perception systems resilient to stealthy adversarial attacks is a challenging problem mainly due to the following critical requirements of practical autonomous driving:

7.1. Limitations of Existing Approaches

The main reason for the created adversarials to be easily mitigated is that a substantial part of the generated attack needs to be observed by classified conditions of the target task other than actual driving situations. Moreover, these generated adversarials also lack exploration of the original equation. And each of them may belong to one of the adversarial classes. Most of them only exploit the feature perturbation method so that such small differences have not been included in the feature set and misclassified. Consequently, selected points belonging to the incorrect class, we introduce the dense sampling method. What features should be included in the feature set are suggested. Furthermore, many existing adversarial attacks always decrease confidence of ADAS systems because they intentionally over-perturb the input in the adversarial generating process.

Before we get to describing our proposed methodology, we present a brief reminder of existing approaches to this problem and their limitations. An adversarial attack, or simply an adversarial, is created by making a neural network 'mis-see' an imperceptible pattern. The created adversarial is usually designed as human-imperceivable noise, which can lead to significant errors for the neural network. The idea is then to add adversarial noise to the object perceived by the vision system so that the system misperceives it. Several recent works propose methods to generate such adversarials for AVP systems. Commercial packages also exploit this research to stress the problems. All in all, the generated adversarials are costly. Some may need to know the inner information of the target AVP system, and some may not generalize to other systems. These adversarials have static attack models and lack exploration of the insecurity of the unique dynamic patterns, such as steering wheel, target adding speed-bridge conditions, and driver behaviors.

7.2. Potential Research Areas

In our adversarial games convened at the DARPA SAIL-ON workshop, we have realized it is important to provide a quantitative characterization of the multi-attack types' adversarial impact on ADS. The multiple adversarial attack strategies involve different metrics or image pairs from both the contiguous vicinity on the input and the discontinuous vicinity on the input. The adversarial models developed in the first three stages of a robust adversarial game are not considered to have any interactions or unified to handle more comprehensive adversarial attacks for images.

Ideas for future forefront research are education on the robustness of deep learning models, as well as the development of a consistent adversarial game strategy that incorporates different attack strategies. Although robust deep learning techniques have attracted increasing attention in recent years, education on robust deep learning is still far from enough, especially on resilient unmanned perception. A major hurdle or opposition of the multi-styles and multi-layer attack strategies has not been appreciated previously yet. It is necessary to establish a mathematical framework to represent and analyze the overall adversarial impact on the ADS.

8. Conclusion and Recommendations

In recent years, attack-resilient research on deep learning has received growing attention. Deep learning (DL) models are vulnerable to adversarial attacks. This vulnerability becomes a huge concern in the context of autonomous system deployment. Without the discovery of vulnerabilities, system invulnerability cannot be guaranteed. The use of an attack dataset and model training for DNN/CNN to form the design for attack-resilience can help resolve this concern. With a great part of the research effort made in dealing with accuracy issues, the question of what accuracy represents the fact of speed limiting zone detection operational design constitutes a series of comprehensive attack-resilience problems has largely been ignored. In this paper, we present an AARR approach to the vulnerabilities by carefully considering input operational design. We contribute to attack models based on popular databases and are assessed by state-of-the-art research related to advanced attack techniques.

In summary, we applied our developed AARR methodology to autonomously make reference values of resized and watermarked images for use in DNN/CNN model training. We modeled common and advanced image changing methods and tested and found that common attacks usually did not damage the performance of the DNN-based speed zone detection model, while advanced attacks usually did. Our findings differ from some of those reported in the literature. We recommend for maximum AARR that designers engage appropriate adversarial experts as a team from the beginning of autonomous system design and development, for attention to both the end product and process. In future work, the AARR process can be further developed, supported by more comprehensive data collection and analysis, and then extended to scenarios such as traffic signs and object detection.

9. References

1. M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to End Learning for Self-Driving Cars," arXiv:1604.07316 [cs], Apr. 2016.

2. A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial Examples in the Physical World," arXiv:1607.02533 [cs], Jul. 2016.

3. A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards Deep Learning Models Resistant to Adversarial Attacks," arXiv:1706.06083 [cs, stat], Jun. 2017.

4. W. Xu, D. Evans, and Y. Qi, "Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks," arXiv:1704.01155 [cs, stat], Apr. 2017.

5. Tatineni, Sumanth. "Federated Learning for Privacy-Preserving Data Analysis: Applications and Challenges." *International Journal of Computer Engineering and Technology* 9.6 (2018).

6. Vemoori, V. "Towards Secure and Trustworthy Autonomous Vehicles: Leveraging Distributed Ledger Technology for Secure Communication and Exploring Explainable Artificial Intelligence for Robust Decision-Making and Comprehensive Testing". *Journal of Science & Technology*, vol. 1, no. 1, Nov. 2020, pp. 130-7, https://thesciencebrigade.com/jst/article/view/224.

7. Mahammad Shaik, et al. "Envisioning Secure and Scalable Network Access Control: A Framework for Mitigating Device Heterogeneity and Network Complexity in Large-Scale Internet-of-Things (IoT) Deployments". Distributed Learning and Broad Applications in Scientific Research, vol. 3, June 2017, pp. 1-24, https://dlabi.org/index.php/journal/article/view/1.

8. N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, “Practical Black-Box Attacks against Machine Learning,” in Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, 2017, pp. 506–519.

9. N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, “Practical Black-Box Attacks against Machine Learning,” arXiv:1602.02697 [cs, stat], Feb. 2016.

10. N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, “The Limitations of Deep Learning in Adversarial Settings,” in 2016 IEEE European Symposium on Security and Privacy (EuroS&P), 2016, pp. 372–387.

11. N. Carlini and D. Wagner, “Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods,” arXiv:1705.07263 [cs, stat], May 2017.

12. A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial Machine Learning at Scale,” arXiv:1611.01236 [cs], Nov. 2016.

13. N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, “Practical Black-Box Attacks against Machine Learning,” in Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, 2017, pp. 506–519.

14. N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, “Practical Black-Box Attacks against Machine Learning,” arXiv:1602.02697 [cs, stat], Feb. 2016.

15. N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, “The Limitations of Deep Learning in Adversarial Settings,” in 2016 IEEE European Symposium on Security and Privacy (EuroS&P), 2016, pp. 372–387.

16. G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” arXiv:1608.06993 [cs], Aug. 2016.

17. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet Classification with Deep Convolutional Neural Networks,” in Advances in Neural Information Processing Systems 25, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

18. K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” arXiv:1512.03385 [cs], Dec. 2015.

19. A. G. Howard et al., "Mobilenets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv:1704.04861 [cs], Apr. 2017.

20. N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks," in 2016 IEEE Symposium on Security and Privacy (SP), 2016, pp. 582–597.

21. G. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," arXiv:1503.02531 [cs], Mar. 2015.