Autonomous Cars Project
Literature Review Report

Daria Zahaleanu, Class of 2020
Prof. Hoda AlKhzaimi
Summer Internship 2018

I. Introduction

Autonomous driving cars are rapidly evolving from sci-fi to reality with the support of many companies including Mercedes-Benz, General Motors, Nissan, Tesla Motors, Audi, and Google. Some of the self-driving cars have been already actively testing fully functioning prototypes on actual roads. Designing and developing partial to fully autonomous cars has been attracting a significant growing interests in cars industry and is likely to continue growing at a rapid rate over the next few decades. Deep learning has received increased attention in both academic and industrial communities and derived algorithms that are core of many vision, robotics, speech recognition, and natural language applications. The research and development in self-driving cars is an active ongoing efforts that involved many partners, which help self-driving cars understand the environment, such as traffic signs and surrounding objects, using the images taken from cameras on the car, or even provide end-to-end control for the car. Recently, in Singapore, we've seen autonomous vehicles taken onto the streets as taxis, and it is expected that more autonomous vehicles (AVs) will increase in popularity in many other countries in the near future. [1] However, as self-driving vehicles are equipped with more electronic sensors such as radars, sonars, LiDARs that are commanded remotely using software and network connectivities, they number of security vulnerabilities and cyber attacks has increased. [1]

Adversaries today become increasingly skillful at sending cyber attacks using low-cost devices that break a car's security system or by sending adversarial samples into the car's network. In an unfortunate hijacked car scenario, the car accelerates when it comes across a traffic sign, instead of slowing down and it may hit pedestrians or other cars in its way, resulting to imminent human and material losses. A common misperception is that the adversaries can only attack the car by attaching a physical device to a certain part of the vehicle – in reality, most of the cyber attacks are launched wirelessly into a car's network. The fact that personal data is stored in or transmitted through vehicle networks becomes even more problematic, as the attackers can now steal a person's identity and manipulate the protocols of biometric authentication. Any cyber security incident is a problem for any automaker in the world and ultimately, it's a matter of public safety that needs to be addressed by developing more advanced attack and defense trainings that would enable researchers and automakers to counteract potential attacks.

Questions such as *What role should the government play in keeping self-driving cars safe from the hackers?* or *How can startups and automakers implement more effective and safe AV components?* bring up real concerns about AVs regulation, which is a vague topic that authorities haven't tackled yet. As a matter of fact, in a recent survey, 30% of the Americans cited safety concerns when asked if they would like to ride in a self-driving car. Autonomous cars safety becomes a social concern that more and more people are aware of, along with the increasing number of hijacked AVs becomes a social issue that needs to be both prevented and countered.

II. Problem Statement

The purpose of this report is to organize and discuss the various methods of attacking and defending AVs, by simulating the autonomous vehicle as an active agent and train it to explore both attacks and defenses in its immediate surrounding environment, through a deep understanding of the security issue of deep neural networks. This report looks into the background of autonomous vehicles and its challenges, by presenting specific examples of attacks: defensive distillation, robust physical perturbations and black-box attacks.

III. Background and Related Work

One of the most important aspects of the AVs' vulnerability to malicious attacks is understanding the reasons that make them susceptible to adversarial attacks. *Thing et al.* accounted for the increase in the communication channels and the vulnerable security testing approaches. The communication channels have a reportedly increased communication between AVs and the external environment that occurs via several ad-hoc, inter-vehicular networks (VANETs), that allows information sharing among nearby autonomous vehicles so that each vehicle is aware of its rapidly-changing surroundings [1]. Therefore, once a malicious attack spreads to an AV, it will spread out to the entire network. Similarly, the intra-vehicular controller area network (CAN) acts as a central network of communication that spreads the malicious thread to all its nodes. Moreover, the fact that CAN packets don't have any authentication factor or source identifier field makes the CAN bus even more vulnerable. [1] Also, the presence of different sensors that are remotely commanded and can be easily accessed by attacking the wireless network account for the sensitivity of the self-driving cars to cyber attacks.

The multitude of potential threats and vulnerabilities that an AV may encounter requires a rigorous classification of these, based on the type of attackers, the attack vector, target, motive and potential consequences. Therefore, while the physical attacks may be more familiar to the readers, the remote-access ones are the most prevalent and harmful nowadays (Figure 1). The physical non-invasive attacks, also known as side-channel attacks represent the most trivial and outdated attacks, that collect data about power consumption, electromagnetic leaks, acoustic signal analysis and data remanence that reveal valuable information about the performance of the vehicle. On the other side, the physical invasive attacks include more refined methods of attacks, such as code modification; code injection where malicious payloads infiltrate AVs; packet sniffing where the target is represented by packets of data are transmitted over the Internet; packet fuzzing, where invalid data is sent to the target system; in-vehicle spoofing, often represented by spam emails. [1]

The remote-access attacks use the enhanced connectivity to collect or inject information from/to LiDARs, cameras or GPS. The most harmful external signal spoofing is the GPS spoofing that occurs when erroneous, deceiving information is fed into the GPS of an AV, by drifting off the path of the car. Jamming attacks are attacks against the wireless medium and external facing sensors, such as LiDARs or cameras, where a jammer device blocks the sensors from receiving signals. [1]

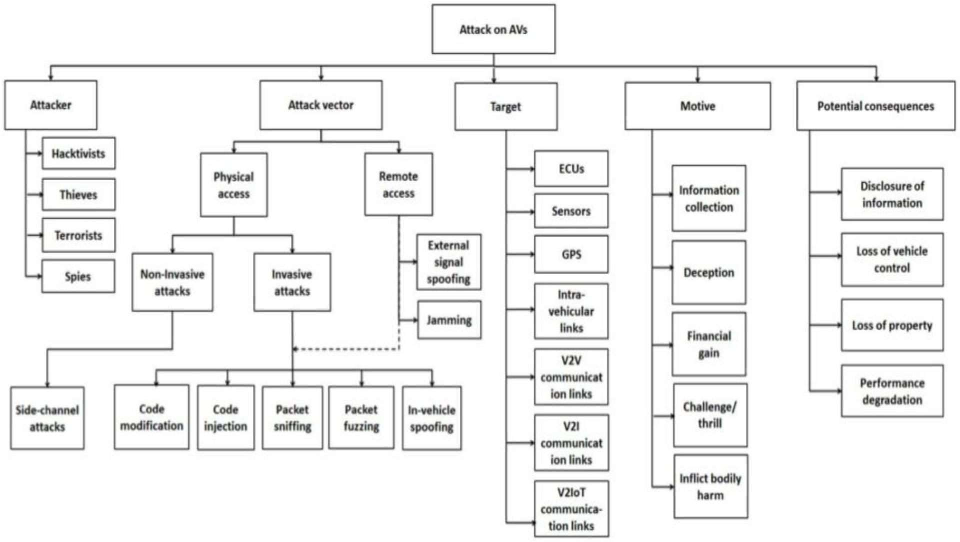

Figure 1. Autonomous Vehicle Attack Taxonomy [1]

Regarding the defense taxonomy, there are four main categories: preventive, passive, active and collaborative defence (Figure 2). In preventive defense, the approach focuses on attempting to stop an attack from happening by ensure more secure ways of communication, in-vehicle device authentication, user authentication or by creating a firewall, a network security system that controls the incoming and outgoing traffic based on a set of rules [1]. In passive defence, adversaries have the intent and capability to harm the AVs; however, these can be stopped by adding another layer of defence, represented by intrusion detection, anti-malware, nullification and isolation of an attack, and the possibility to recover from attacks. By active countering the attacks, users have the option to continuous monitor the AV infrastructure (it provides snapshots of the security status) or to use adaptive security, set of measures that generate deceptions tactics. Collaborative defence is represented by cloud computing. [1]

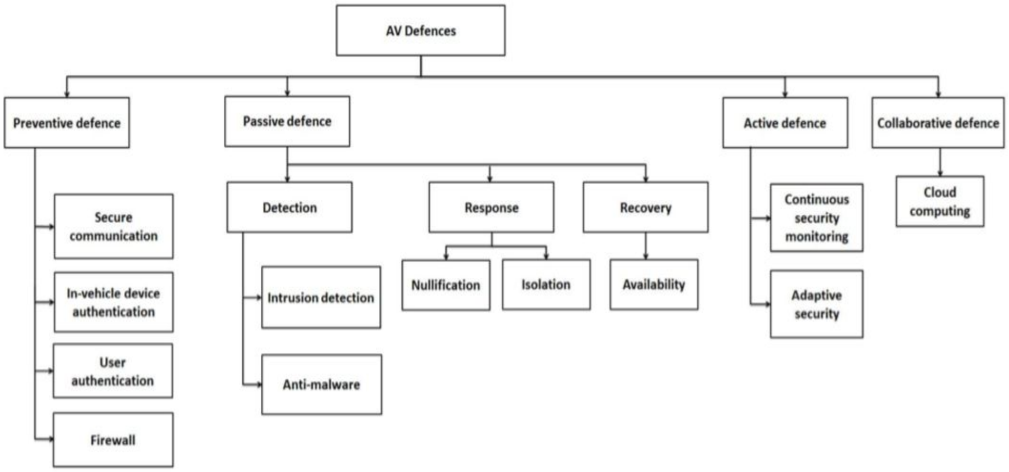

Figure 2. Autonomous Vehicle Defense Taxonomy [1]

While the classification of malicious threats against AVs is clear to us, the detection and the combat of these threats remains an open problem that can only be solved by means of deep learning (DL) models. According to *Papernot et al.*, deep learning takes advantage of large datasets to achieve accuracy rates higher than previous classification techniques [2]. Nowadays, DL models are able to multitask in domains like speech recognition, language processing, vision etc., thanks to a new organization into *deep neural networks* (DNN). DNN are large neural networks organized into layers of neurons, corresponding to successive representations of the input data [2]. Neurons are modeled with different weights and biases, parameters used for information storage. Network training is done by gradient descent using *backpropagation*, which is a method used to calculate the gradient that corresponds to the weights that are to be used in the network. DNNs are modeled using a layer of neurons, which are computing units applying an activation function (sigmoid, ReLU, tanh) to the previous layer's weighted representation of the input to generate a new representation [4]. DNN defines and computes:

$$F(x) = fn\,(\theta n, fn-1\,(\theta n-1, \ldots f2\,(\theta 2, f1\,(\theta 1, x)))) \,, \qquad (1)$$

where θ is a weight vector impacting on the neuron's activation.

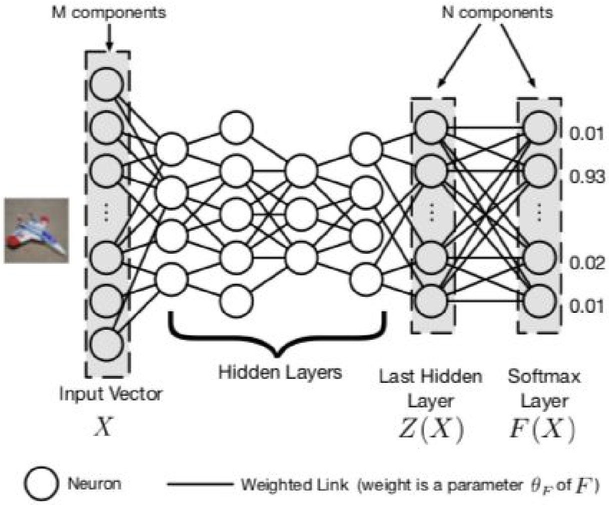

Figure 3. DNN architecture [4]

In the learning phase of a DNN, function F, the output, learns values by being fed in large sets of input-output pairs $(\vec{x}, \vec{y})$ and it adjust the weight parameters to reduce the prediction error between the prediction $F(\vec{x})$ and the output $\vec{y}$. During the test phase, the DNN is fed a set of parameters θ to make predictions on inputs unseen during training [4].

However, the increased use of DL and the access to large data sets comes with the disadvantage of having more classified inputs, that are the result of adversarial samples. *Adversarial samples* are malicious inputs modified to yield erroneous model outputs that appears unmodified to human observers. In other words, the inputs are altered by adding imperceptible perturbations to force the classifier to misclassify the adversarial inputs, while it remains correctly classified by the human observer. A classifier is a machine learning model that learns a mapping between inputs and a set of classes [2]. For example, in Figure 4, the classifier would normally classify the samples as numbers from 0 to 9, but because of the distortion that is added to input samples, the DNN that takes the output from the classifier misclassifies them. Adversarial samples solve the following optimization problem:

$$\vec{x}^* = x + argmin\{\vec{z} : \tilde{O}(\vec{x} + \vec{z}) = \tilde{O}(\vec{x})\} = \vec{x} + \delta_{\vec{x}} \quad (2)$$

where O represents the oracle, the entity that queries for the label for a particular data point; x, z are input vectors and x* is the adversarial sample, such that $\tilde{O}(\vec{x}^*) \neq \tilde{O}(\vec{x})$.

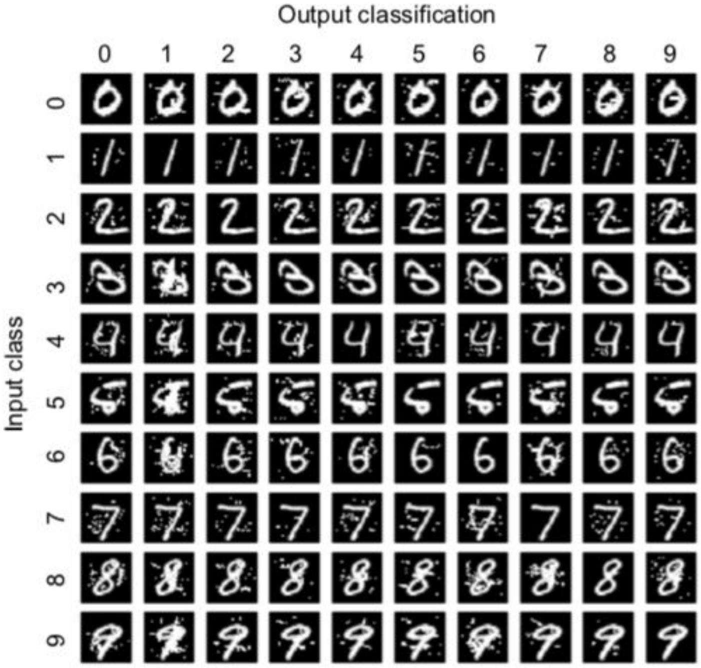

Figure 3. Adversarial sample generation

IV. **Specific Sections related to the problem statement**

A. Defensive Distillation

Distillation is a training method designed to train a DNN using knowledge transferred from a different DNN. In the context of defense to adversarial perturbations against DNNs, defensive distillation is a type of distillation that uses the knowledge from a DNN to improve its resilience to adversarial samples. At the same time, it extracts additional knowledge about training points as class probability vectors, which is fed back to the regimen [3]. Overall, defensive distillation reduces the effectiveness of adversarial samples from 95% to 0.5% (*Papernot et al.*) and it smoothes out classifier models, by reducing the sensitivity of a DNN to input perturbations by a factor of 10^{30} [3].

The general framework that describes the adversarial sample crafting contains two folds: direction sensitivity estimation and perturbation selection. In this adversarial model, it is assumed that the adversary has the capability to access the θ parameter. In the direction sensitivity estimation step, DNN identifies directions in the data around sample X in which model F is most likely to result in a class change. In other words, the goal of this step is to find the dimensions of the input X that will produce the expected adversarial behavior with the smallest perturbation. In the perturbation selection step, the adversary uses the knowledge from the previous step to select a perturbation affecting sample X's classification. Figure 4 describes the whole process: if no perturbation X* is identified, then the process starts over again with the direction sensitivity estimation, whereas if a perturbation X* is identified, then the value of X is replaced by the value of X*.

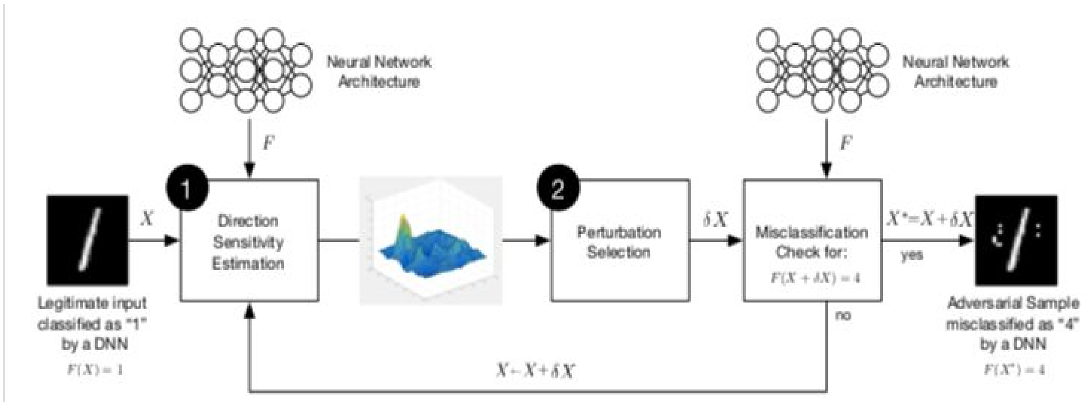
Neural Network Architecture
Neural Network Architecture
F
F
X
1
Direction Sensitivity Estimation
2
Perturbation Selection
δX
Misclassification Check for:
F(X + δX) = 4
X*=X+δX
yes
no
Legitimate input classified as "1" by a DNN
F(X) = 1
Adversarial Sample misclassified as "4" by a DNN
F(X*) = 4
X←X+δX

Figure 4. Adversarial Crafting Framework [3]

B. Robust Physical Perturbations

Eykholt et al. evaluated the effectiveness of perturbations on physical objects and showed that adversaries can physically modify objects using low-cost techniques to reliably cause misclassifications [3]. The robust physical perturbation attacks generated random perturbations by adding stickers and graffiti that would lead to the misclassification of object by DNN, without arousing suspicion in human operators. Therefore, the goal of this type of attack is to effectively create adversarial samples where the object itself is physically perturbed by placing stickers on it. The physical world challenges for this attack include different physical conditions such as the environment, the spatial constraints, the physical limits on imperceptibility, fabrication errors and nonetheless the angle and distance from the camera of the generated perturbations.

Based on the above description, two types of tests have been run on two different types of classifiers (LISA-CNN and GTSRB-CNN): stationary tests and drive-by tests, where the researchers used a camera in a vehicle approaching a sign at different angles and distances. The subtle posters added as a mask on top of the real sign, as well as different types of camouflage (graffiti, stickers) registered different types of success rates that reflect the proportion of adversarial samples misclassified by the substitute DNN. These results are summarized in Figure 5 and 6 below, where, for each image the top two labels and their associated confidence values are shown.

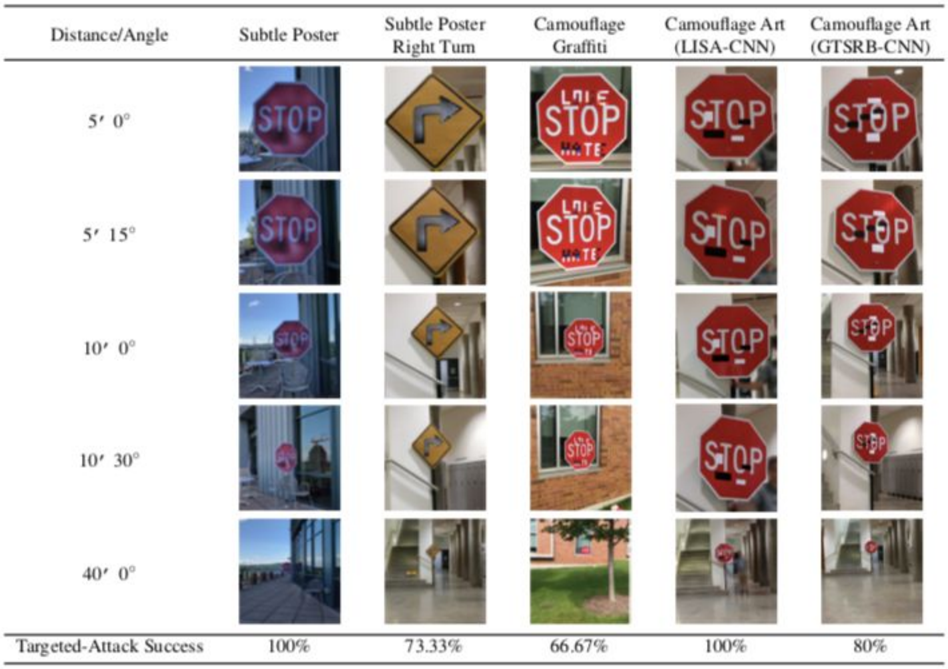

Figure 5. Stationary testing summary for the two types of classifiers (LISA-CNN and GTSRB-CNN)

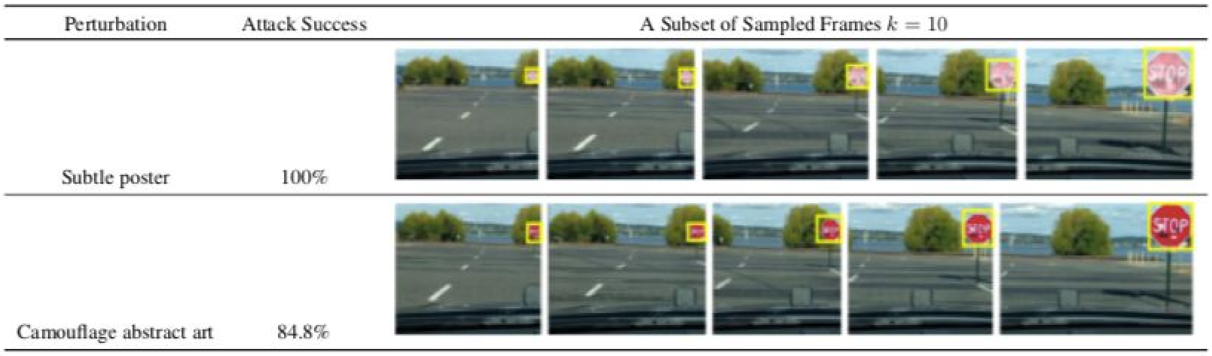

Figure 6. Drive-by testing summary for LISA-CNN.

C. Black - Box Attacks

Unlike the other two type of attacks, classified as white-box attacks, as the adversary has full access to the internal structure of the classifier, in black-box attacks, the adversary doesn't have any knowledge about the model, except for the the labels, the allow him/her access to the output only. The goal is as we previously saw, to produce a minimal perturbation to input X, sufficient to determine the DNN to misclassify it, but imperceptible enough for the human eye. The approach to this type of attack is as following: use DNN as an orale to synthesized data and generate a synthetic data set S0 to build a model F that approximates the oracle's decision. Then using the new architecture F to craft adversarial samples, the oracle outputs a misclassified probability vector, corresponding to input X*. This step, called substitute DNN training algorithm, ends with the Jacobian-based dataset augmentation, which generates a new set of data S (Figure 7). Once the adversary training a substitute DNN, it uses it to craft adversarial samples, by using two types of algorithms: Goodfellow, or the fast gradient sign method and Papernot.

Therefore, the black-box attack generalizes to machine learning models that are not necessarily DNNs, regardless of the differentiability of the functions.

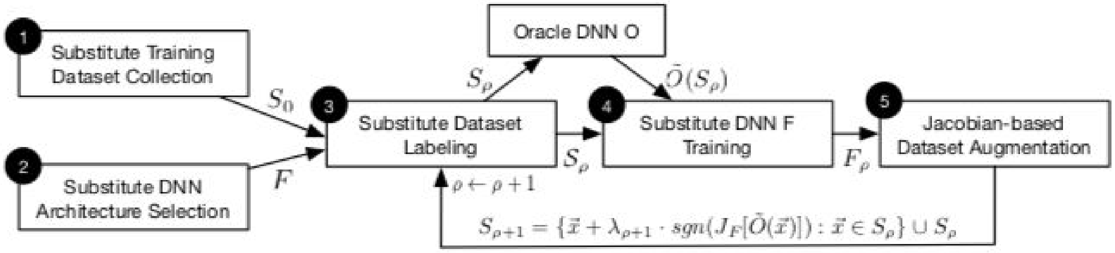
1
Substitute Training Dataset Collection
2
Substitute DNN Architecture Selection
S_0
F
3
Substitute Dataset Labeling
Oracle DNN O
S_ρ
$\tilde{O}(S_\rho)$
4
Substitute DNN F Training
S_ρ
F_ρ
5
Jacobian-based Dataset Augmentation
$\rho \leftarrow \rho + 1$
$S_{\rho+1} = \{\vec{x} + \lambda_{\rho+1} \cdot sgn(J_F[\tilde{O}(\vec{x})]) : \vec{x} \in S_\rho\} \cup S_\rho$

Figure 7. DNN Substitute Training [4]

V. **Conclusion and Future Work**

This report paper broadly explored possible adversarial attacks in deep learning systems, by presenting three of the most used adversary algorithms: defensive distillation, robust physical perturbations and black-box attacks.

This paper sets the basis for future work which will simulate the autonomous vehicle as an active agent and train it, in order to to explore both attacks and defenses in its immediate surrounding environment, through a deep understanding of the security issue of deep neural networks. The proposed project aims to develop an adversarial deep learning model that offers a balanced defensed under a reasonable threat so that mitigate the attacks to security sensitive application of autonomous vehicles

VI. **References**

1. L.L. Thing, Vrizlynn & Wu, Jiaxi. (2016). Autonomous Vehicle Security: A Taxonomy of Attacks and Defences. 164-170. 10.1109/iThings-GreenCom-CPSCom-SmartData.2016.52.
2. Papernot, Nicolas & McDaniel, Patrick & Jha, Somesh & Fredrikson, Matt & Berkay Celik, Z & Swami, Ananthram. (2015). The Limitations of Deep Learning in Adversarial Settings.
3. Papernot, Nicolas & McDaniel, Patrick & Wu, Xi & Jha, Somesh & Swami, Ananthram. (2016). Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks. 582-597. 10.1109/SP.2016.41.
4. Papernot, Nicolas & McDaniel, Patrick & Goodfellow, Ian & Jha, Somesh & Berkay Celik, Z. & Swami, Ananthram. (2016). Practical Black-Box Attacks against Machine Learning. eprint arXiv:1602.02697.
5. Szegedy, Christian & Zaremba, Wojciech & Sutskever, Ilya & Bruna, Joan & Erhan, Dumitru & Goodfellow, Ian & Fergus, Rob. (2013). Intriguing properties of neural networks.
6. Evtimov, Ivan & Eykholt, Kevin & Fernandes, Earlence & Kohno, Tadayoshi & Li, Bo & Prakash, Atul & Rahmati, Amir & Song, Dawn. (2017). Robust Physical-World Attacks on Machine Learning Models.