Adversarial Attack on Autonomous Vehicles

An Approach To Deceive Traffic Light Detection System

Tahsin Mullick, Sharmin Afrose and Sheik Murad Hassan Anik

Abstract

Autonomous cars are gradually occupying the streets. In the near future, they will become ubiquitous. Recent events that resulted in the loss of human lives forces us to raise questions about their security, primarily concerning their onboard learning algorithms that act as the brains of the car. In our paper, we expose one such vulnerability i.e. street sign misclassified as a traffic light. To the best of our knowledge, this vulnerability has not been explored in the existing literature. We achieve the stated vulnerability using a white box adversarial attack by creating perturbations that can fool a learning algorithm to misclassify traffic lights. We consider ImageNet dataset and our custom curated dataset to test and train our model and generate adversarial images to expose vulnerabilities in the learning algorithm.

Index Terms

Information Sec, Image Classification, Autonomous Vehicle, Traffic Light Detection, Adversarial Attack.

I. Introduction

In recent times, autonomous cars have been garnering substantial prominence. A number of autonomous cars including Tesla Autopilot, Volkswagen and Audi's Traffic Jam Pilot, Ford Argo AI, Daimler Intelligent Drive, etc are being more visible on the street. These driverless cars are designed to provide numerous options to the drivers with regards to safety as well as comfort. These autonomous cars can sense their surrounding environment and classify objects (pedestrian, traffic lights, street signs, etc) and can take appropriate actions. It is highly essential for the manufacturers to integrate model architecture that can classify the objects accurately. Otherwise, a tragic accident can occur that may endanger human lives. Therefore, it is paramount to design an accurate classifier model in autonomous cars.

It is challenging to build an accurate classifier model. On the initiation of adversarial attack, the task of accurate classification becomes more challenging. In the adversarial attack, careful modification or perturbation in an object image can cause the classifier to misclassify to another object. In recent works [1], [2], carefully crafted perturbation is done on traffic signs to misclassify it as other traffic signs. The challenges of creating adversarial images are that a minimal amount of perturbation has to be done so that the changes are not visible to the human eye, yet the classifier model of the autonomous car misclassifies the images.

In our project, we execute the adversarial attack on street signs to be misclassified as traffic lights and deceive the autonomous car. To accomplish the task, we use Inception-v3 model. Specifically, the target image is passed through a Convolutional Neural Network (CNN). The loss will be calculated from the softmax layer. The gradient descent optimizer calculates the perturbation from the loss function. The perturbation is added in the target image and gets evaluated in the CNN again. We evaluate our performance of the generated image using L2 distance and Structural Similarity Index (SSIM). Figure 1 shows the correct classifiation of the image classifier and Figure 2 shows the the result of our crafted adversarial sample on the same classification model. There is no visible dissimilarity between the orginial and adversarial image.

Our contributions are summarized as follows.

- We design a novel application case of generating adversarial examples of street sign that is misclassified as traffic light.
- Our generated perturbations are very small that are undetectable to human eye. The evaluation result of SSIM and L_2 distance shows the quality of adversarial images.
- We also perform transfer learning to create a custom classifier. We tested and trained our custom classifier on our customized dataset.

II. Motivation

Autonomous cars are an imminent future.They are also open to a wide array of security threats. Being equipped with multiple sensors that detect and sense the world around them. A failure in any aspect of the learning algorithms that perform these vital operations can lead to life threatening disasters. The project is motivated to expose these threats particularly with respect to generating and utilizing adversarial perturbations that can severely affect the learning algorithms. The hope is that exposing effectiveness of such attacks can help researchers work towards more robust learning based solutions that are not easily susceptible to these security threats.

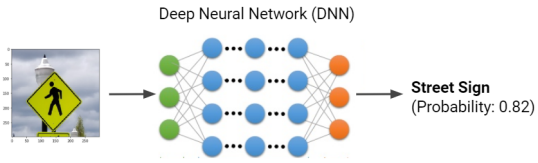

Fig. 1: Classification

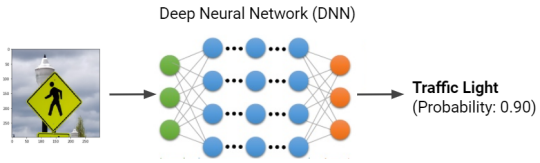

Fig. 2: Misclassification due to Adversarial Sample

III. Related Works

Generation of adversarial examples rely on two important components which include firstly detection of traffic lights and signs and secondly the generation of perturbation. Detection of street signs and lights have been studied quite extensively in literature with robust algorithms that are able to harness the advances in computer vision effectively.

Detection methods can be classified into either image processing approach or machine learning based approach. The image processing methods primarily rely on shape and color as cues for detection, work done by Linder*et al*. [3] and Franke *et al*. [4] on traffic lights focused on classification of color of each pixel and followed it with the usage of component analysis for segmentation to locate regions of interest. With respect to traffic sign detection Athrey *et al*. [5] used thresholding and blob detection with template matching. Omachi *et al.*[6] grouped pixels that exceeded a specific threshold by normalizing the color space of images. Image processing techniques offer advantages based on the fact that they are not data dependent as such do not suffer from overfitting and work well in specific scenarios. They are however prone to face challenges under slight variability.

The lack of robustness in image processing approaches to variable scenarios is a gap that can be filled by resorting to learning based models. The advantage of learning based models is that they can be trained on a broad set of traffic light or signs under different lighting conditions in various environments. ACF based detectors presented by Morten *et al.* [7] and Philipsen *et al.* [8] outperformed image processing detectors on the LISA dataset. YOLO by Redmon *et al.*[9] is an algorithm that is used to detect traffic lights which includes a separate CNN to classify states of traffic lights. Traffic sign detection research too makes use of a variety of learning based models such as multi-scale CNN by Sermanet and LeCun[10], Support Vector Machines (SVMs) that coupled with CNNs to develop finer classification as presented in Yang *et al.*[11]. Pon *et al.* [12] presents a way to enable detection of both traffic light and traffic sign using a combined data set. Their deep hierarchical architecture works with a mini-batch proposal selection mechanism. They solve overlapping issues of data set in their proposed method and are one of the first networks to perform joint detection on traffic light and traffic sign.

Once the problem of detection of traffic lights and sign are dealt with, the next step in creating adversarial examples is generation of perturbations. One of the seminal works in this field was by Szegedy *et al.* [13] where they discovered several models, including state-of-the-art neural networks being vulnerable to adversarial examples and misclassification. There method of generating perturbation involved taking advantage of discontinuity in deep neural networks input output mappings. The perturbations in their work stemmed out of maximizing the networks prediction error. They were also able to show that the same perturbation could be applicable to multiple different networks, which were trained on different subset of the dataset misclassifying the same input. This paper led to many other papers where authors have applied to adversarial examples to different applications to expose security issues. Amidst them work done, Eykholt *et al.* [2] propose a general attack algorithm, Robust Physical Perturbation (RP2) that was robust under different physical conditions. Their algorithm was tested on real-world road sign classification. In another paper by Chawin *et al.* [14] traffic sign recognition was shown to be compromised when physically printed perturbations were superimposed on traffic signs.

The work done in this report is unique in that is focuses on misclassifying street signs as traffic lights. Work of this kind has not been presented till now in literature to the best of our knowledge.

IV. Our Approach

This section outlines the approach taken by the project and serves to present a brief overview of the experimentation section which subsequently delves into extensive details. Additionally we present the threat model for the project where we discuss the considered constraints and targets of attack.

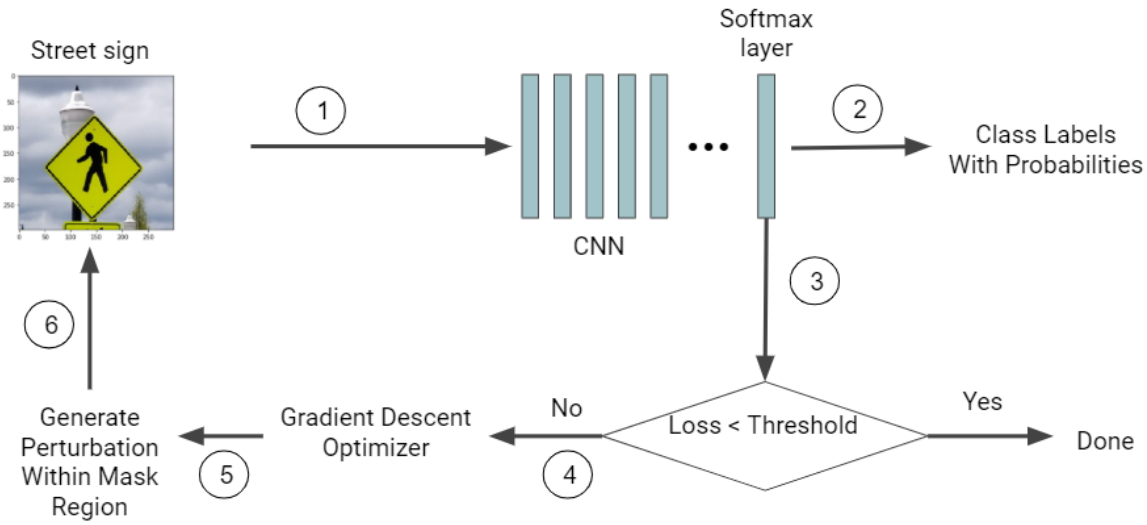

Fig. 3: System Architecture Overview

A. *Methodology*

The work of developing a model which successfully misclassifies street signs, requires a strong learning based approach which in this case is the Inception-v3. The Inception-v3 has been trained on the ImageNet dataset. It has been chosen due to its superior performance considering a large number of layers that are able to extract fine features from images. The ImageNet dataset is divided into a variety of classes. Each class has images in different lighting conditions and orientations with respect to that class.Thus they perfectly fit out our requirements for usage in generation of adversarial examples.

As shown in Figure 3 in step 1 an image from our own data set is passed on to the Inception-v3. After beign processed through the layers of the convolutional network the softmax layer outputs class labels with probabilities in step 2. These probabilities signify its belief of the class to which the input image belongs. The softmax layer also happens to be the the layer that is tapped into to extract values with regards to the perturbation generation as indicated by step 3.

If the loss is less than the set threshold, the perturbation is considered done else the values is fed into the fast gradient descent optimizer as illustrated in step 4. The gradient descent optimizer creates and outputs the perturbations as per the set parameters in step 5 . Finally step 6 superimposes the perturbations onto a given image and attempts to deceive the CNN into misclassifying the image as another class. A amplified version of the perturbation is presented in Figure 4.

B. *Threat Model*

The work presented in this report is an attempt at exposing security threats. Following is a description on our threat model and which clearly defines the type of attack carried out, the constraints set on the modification and specific targeted attack. In this section, we will discuss our threat model. For our adversarial attack, the attacker is bound by some constraints. The threat model discussion is as follows:

- White-box attack scenario: We assume a white-box scenario where the attacker will know the internals of the detection model.
- Modification restriction: Even though the adversary has white-box access to the detection system, we will maintain the model modification restriction meaning the adversary will not be able to change the detection model or any of its parameters.
- Targeted attack: We only consider the targeted attack, where the attacker classifies street sign images as traffic lights.
- Training data poisoning: The adversary cannot poison the image classification system by altering training data.

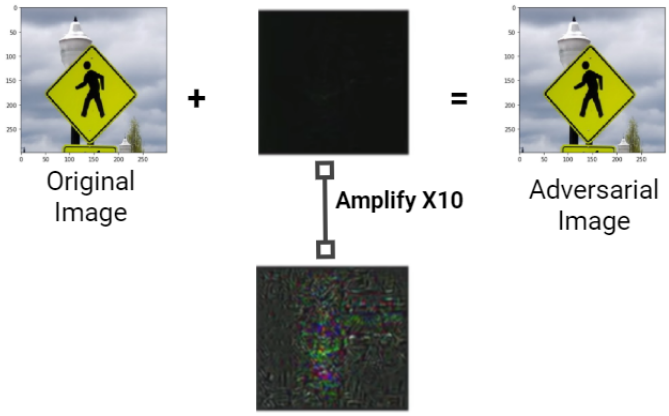

Fig. 4: Perturbation Added in Street Sign

V. Technical Challenges

In this section, we will discuss the several technical challenges that we faced during the project and had to overcome to craft adversarial samples for image classification model.

- Computational Capabilities: Deep neural networks work as the backbone of image classification systems. These networks require large amount of data and processing power for training the model. They require multiple Graphics Processing Units (GPU) for parallel computing. During this project, we did not have access to such powerful devices, we had to implement the model in CPU only environment. The training requires months to complete, so we used pre-trained model for our approach.
- Pre-trained Model: We used the pre-trained Inception V3 [15] model for our classification. This model is not dedicated for traffic light and street signs but it worked considerably well in classifying our test images. We initiated designing our custom model because we were unable to find dedicated pre-trained model for traffic lights and street signs.
- Transfer Learning: For our custom classifier, we used Inception V3 as our teacher model. We faced issue in saving the student model as a checkpoint. So, we overcame this issue by generating student model in the graph format. It can correctly classify different street signs and traffic lights.

VI. Experiment

In this section, we will discuss details of our approach for crafting adversarial samples to fool image classifier for detecting street signs. We have categorized this section in three subsections, (i) Data Description, (ii) Classification Model, (iii)Data Pre-processing, (iv) Custom Dataset & Model and (v) Test Data, (vi) Attack Method, (viii) Evaluation Matrices, and, (ix) Result. We will begin our discussion with dataset and classifier for traffic light and street sign recognition.

TABLE I: Parameter Specification of Experiment

Parameters	**Specification**	
Architecture	Inception-v3	Custom Classifier
Dataset	Google ImageNet	Custom Dataset (Blacksburg)
Dataset Size	15 Million	1,250
Image Size	Variable	300 X 300

A. Data Description

In this project, we used the ImageNet [16] dataset. It contains over 14 million images. Our classifier was trained over 1000 classes of ImageNet. The images have been collected from multiple sources and are of different resolutions. Among the 1000 classes, class #919 and #920 denote street signs and traffic lights respectively. ImageNet is a large dataset but our assumption is that it is used in a specific manner for navigation of autonomous vehicles. Autonomous vehicle manufacturers do not reveal their training and testing dataset to minimize security threats but there are only a limited number of large datasets and classifiers available. So, it is an easy assumption that the vehicle manufacturers extend the work of these datasets and classifiers for their cause. ImageNet is one of the largest available public datasets and for this reason, we decided to work with this dataset in our project. Table I shows the specifications of the dataset and classifier used.

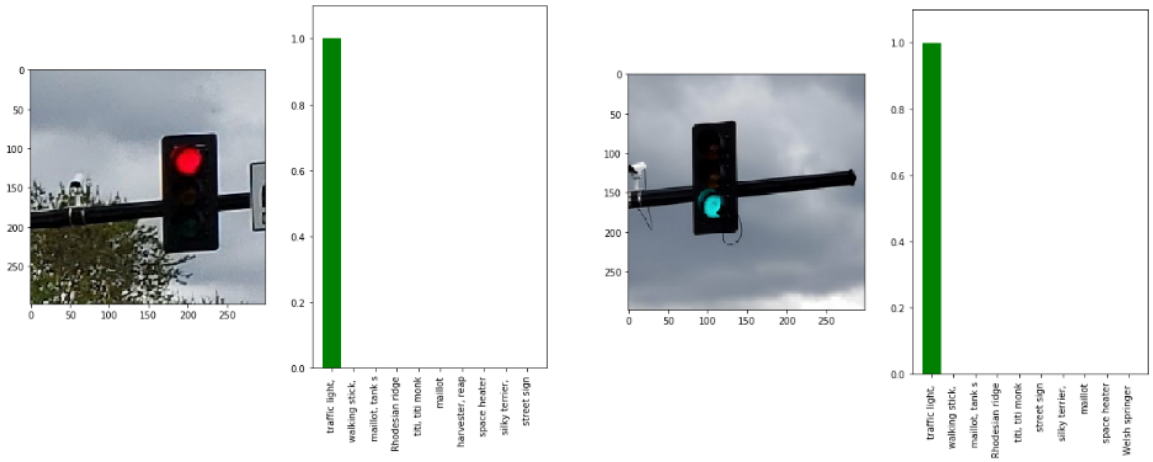

Fig. 5: Correct Classification of Traffic Light using Original Image

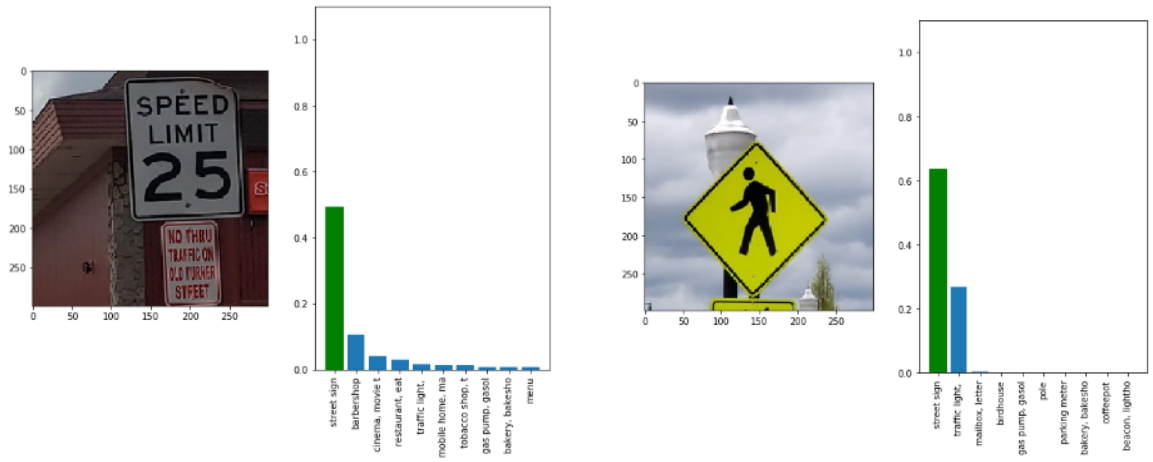

Fig. 6: Correct Classification of Street Sign using Original Image

B. Classification Model

In this project, we used the Inception-V3 [15] image classification model. It is trained on the ImageNet [16] dataset provided by Google. It has 1000 classes including street sign and traffic light which are of our key interest. The model is a 42 layer deep neural network where the input layer take the images as input and the softmax layer outputs the probability distribution of the input image on the 1000 classes. We classify the input image to class that resulted maximum probability. The Table I shows the specifications of the our dataset and classification model.

We did not go with the hassle of training our classifier on this huge dataset as it would take months to get a perfect model. We worked with the pretrained version of Inception-V3 model trained over 1000 classes of images from ImageNet.

Figure 5 and 6 shows the classification efficacy of our model. It is clearly visible that the model can classify traffic lights and street signs with high confidence.

C. Data Pre-processing

As we mention in the previous section, the Imagenet contains over 14 million images and our our classifier is trained on these images with 1000 classes. These images came from different sources and have different resolution. To make these images workable with our model, we had to resize them to 300x300 pixels resolution.

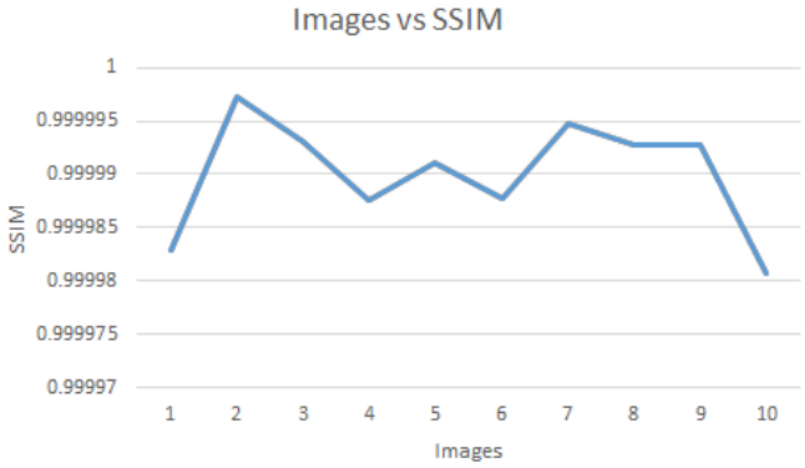

Fig. 7: Evaluation Result of 10 Adversarial Images in Terms of SSIM

We did this in our code. This resize approach breaks the aspect ratio of the image but it is the only way to process images for classification model.

D. Custom Dataset & Model

We explained earlier that the autonomous vehicle manufactures tend to use their custom dataset and classifiers for their vehicles to minimize security risk. For this reason we also decided to build up our own dataset and classifier specifically for the task of classifying street signs and traffic lights.
The processing of building the custom dataset and classifier is still at the initial phase. We collected over 1200 images from the city of Blacksburg. The images are spanned over 5 classes, namely red light, green light, yellow light, 25 MPG and pedestrian walk sign. Each class contains over 200 images. We plan to collect more images and increase the number of classes in future.
For dedicated classification of street sign and traffic lights, we designed a custom classifier. We used transfer learning for building the custom classifier. Inception V3 model was used as the teacher model of transfer learning. As we are still in the phase of collecting data for our custom classifier, the processing of fine tuning the classifier is ongoing.
In this paper, we demonstrate all classification and attack results with our custom collected dataset. The source images shown in the classification result in Figure 5 and 6 are taken from our custom dataset.

E. Test Data

To evaluate our method of attack, we ran the classification model against our custom built dataset. The images were collected from the wild (the city of Blacksburg). We evaluated the model with traffic light and street sign images.

F. Attack Method

For crafting adversarial samples of traffic lights from street signs to fool image classification systems, we tried several methods of adversarial image generation like Gradient Attach, L2 Iterative Attack, C&W Attack etc. Due to limitation in GPU power, these approaches were too slow in our machine. So, we went with the Fast Gradient Sign Descent Method (FGSM) [17] to generate adversarial samples for our attack.
The algorithm uses optimization method to generate perturbation. The single-image optimization problem searches for perturbation δ to be added to the input x, such that the perturbed instance $x^{'} = x+\delta$ is misclassified by the target classifier $f_\theta(.)$:

$$min \quad H(x+\delta, x), \quad s.t. \quad f_\theta(x+\delta) = y^* \tag{1}$$

where H is the $lp - norm$ of the distance function, and y^* is the target class. We are only considering targeted attack in this project. The above optimization problem is reformulated in the Lagrangian-relaxed form as:

$$\text{argmin}_{\delta} \quad \lambda||\delta||_p \quad + \quad J(f_\theta(x+\delta) = y^*) \tag{2}$$

Here, $J(...)$ is the Jacobian loss function, which measures the difference between the target label y^* and the model's prediction. λ controls the regularization and distance function H is specified here as $||\delta||_p$, denoting the l_p norm of δ.

For our attack, we include two stopping criteria, (i) a threshold of 0.0001 loss value and (ii) 3500 as maximum number of iterations. So, if the adversarial image gets classified as the original class even after 3500 iterations, then we call it a fail.

G. Evaluation Matrices

Here, we discuss about the evaluation metrices for our proposed adversarial attack on facial recognition systems. The attack success rate will be evaluated using the following equation [13]:

$$successRate = \frac{imagesMisclassifiedDueToAttack}{imagesClassifiedCorrectlyBeforeAttack} \tag{3}$$

Our second evaluation will be based on the Structural SIMilar measure (SSIM) [18] between the original and the adversarial image. The SSIM is the similarity metric for the perturbed image and the actual image. DSSIM (Structural Dissimilarity) is a distance metric derived from SSIM (Structural SIMilarity). The basic form of SSIM compares three aspects of the two image samples, luminance (l), contrast (c), and structure (s). The SSIM score is then described in the following equation:

$$SSIM(x, y) = l(x, y) * c(x, y) * s(x, y) \tag{4}$$

We will also use the l2 distance [19] for our evaluation.

$$D_{L_2}(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{5}$$

It is the Euclidean distance between the two images and for our case, we pass the original and the adversarial image in the function. In this scenario, the lower l2 distance we can get the better.

H. Result

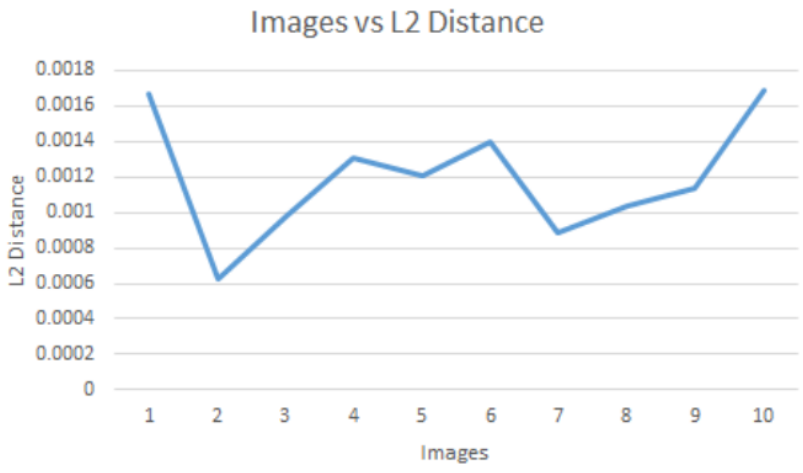

Fig. 8: Evaluation Result of 10 Adversarial Images in Terms of L_2 Distance

We evaluate the effectiveness of our approach by generating 10 adversarial examples of street signs of the class #919 to be classified as traffic light of the class #920. All crafted adversarial images were succesful in fooling the classification model. So, the evaluation measure in eq: 3 provides 100% success rate.

As mentioned earlier, we used both Structural Similarity Index (SSIM) [18] and l2 distance [19] between source image and adversarial image for evaluation. The result is summarized in Table II. In Figure 7 we see the variation of our SSIM scores over different source images and their corresponding adversarial images. Figure

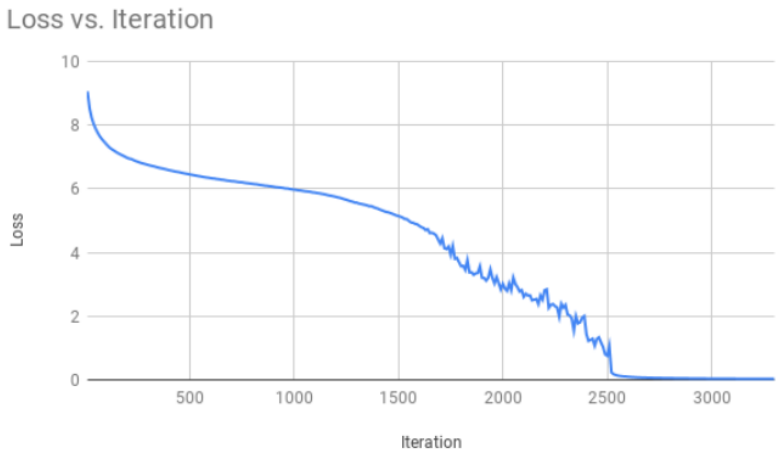

Fig. 9: Convergence of Adversarial Attack

TABLE II: Result Summary

Image #	SSIM	L2 Distance
1	0.999983	0.00167
2	0.999997	0.000624
3	0.999993	0.000978
4	0.999988	0.001306
5	0.999991	0.001203
6	0.999988	0.001401
7	0.999995	0.000882
8	0.999993	0.001033
9	0.999993	0.001134
10	0.999981	0.001688

8 shows the l2 distance measures of the different original-adversarial image pairs.
We include some examples of classification and attack results in this report. Figure 5 and 6 shows the classification before the attack. We can see that the model can perfectly classify traffic lights and street signs with high confidence.
Figure 10 (a) shows the attack results of our method. Here we can see, a pedestrian walk sign is being classified as a traffic light with near 100% confidence. The 10 (b) shows the generated perturbation for the attack. The left most image of 10 (b) is the original perturbation. As it is not visible, we first amplified it 10 times (middle one) and then 100 times for better comprehensibility.
Figure 9 shows the convergence of loss against number of iteration for our attack. It is clearly visible that the loss turns to near zero around 2500 iteration. We set our attack to run until a threshold of 0.0001 is reached or 3500 iterations were over.

VII. Discussion & Limitations

From the result section, we can say that our crafted adversarial samples were classified to the targeted class with very high confidence, high SSIM and low l2 distance. It shows the severeness of such attacks that are possible within very limited resource and time. Here, we will discuss some possible defence against our attack and limitation of our work.
Possible Defenses To defend against such adversarial attacks, we are proposing three defense mechanisms.

- Adversarial Training: We can synthetically generate adversarial samples and train a detector model to detect adversarial samples them as fake. Putting a detector before the classifier will help in this regard.
- Shape & Color Constraint: Traffic light and street signs come with different shape and colors. We can use this attribute to a validation before the classification process to limit the working domain of the classifier.
- Obfuscated Gradient: Obfuscated gradients can be introduced in the classification neural network so that adversarial attacks based on gradients can be defended.
- Noise Reduction: Different image processing and noise reduction methods can be used to remove the perturbations before the classification process.

Limitations Although our attack is 100% successful for this test dataset, one important thing is that our dataset is small and some hyper-parameters were needed to tune a lot for some of the input images to converge. This

(a) Original Image Transform to Adversarial Image (b) Perturbation

Fig. 10: Correct Classification of Street Sign using Original Image

particular attack is not robust which means it may or may not work in different environmental situations like rain, snow, or different angle and distance. We do plan to work on this section in near future.
The attack was also performed in the entire image which is not practical in real world scenario. So, for future work, we will include mask on source image for constraining the spatial region of perturbation.
In addition, the notion of inconspicuousness is subjective, and the only way to quantify it adequately requires to incorporate human-subject studies. We did SSIM(Structure Similarity Index) and l2 distance for that measure. In our future work, we plan to work on make this attack more robust and more compatible to the real-world. So that we can perturb different angles of a image and can generate an uniform perturbation for that image.

VIII. Conclusion

In this project, we crafted adversarial samples to attack on the traffic light detection and recognition system that are likely to be used in self-driving or autonomous vehicles. The industry of autonomous vehicle is uprising and the detection of traffic lights is very important for the safety of the car itself, the passengers as well as the nearby people and properties. We showed that such attacks are not impossible to craft and if such attack do occur, it will result in catastrophe. Our vision of this project is to see ahead of time so that there is enough time to create strong defense against such adversarial attack. In our report, we discuss possible extension of our attack to generate more robust adversarial samples and mention some possible defense mechanisms that might work against our proposed attack method but not all adversarial attacks. So, we believe, more thorough research experiments should take place to devise appropriate defense against such attacks.

References

[1] C. Sitawarin, A. N. Bhagoji, A. Mosenia, M. Chiang, and P. Mittal, "DARTS: deceiving autonomous cars with toxic signs," *CoRR*, vol. abs/1802.06430, 2018. [Online]. Available: http://arxiv.org/abs/1802.06430

[2] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. X. Song, "Robust physical-world attacks on deep learning visual classification," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1625–1634, 2018.

[3] F. Lindner, U. Kressel, and S. Kaelberer, "Robust recognition of traffic signals," in *IEEE Intelligent Vehicles Symposium, 2004*, June 2004, pp. 49–53.

[4] U. Franke, D. Gavrila, S. Gorzig, F. Lindner, F. Puetzold, and C. Wohler, "Autonomous driving goes downtown," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 6, pp. 40–48, Nov 1998.

[5] K. S. Athrey, B. M. Kambalur, and K. K. Kumar, "Traffic sign recognition using blob analysis and template matching," in *Proceedings of the Sixth International Conference on Computer and Communication Technology 2015*, ser. ICCCT '15. New York, NY, USA: ACM, 2015, pp. 219–222. [Online]. Available: http://doi.acm.org.ezproxy.lib.vt.edu/10.1145/2818567.2818609

[6] M. Omachi and S. Omachi, "Traffic light detection with color and edge information," in *2009 2nd IEEE International Conference on Computer Science and Information Technology*. IEEE, 2009, pp. 284–287.

[7] M. B. Jensen, M. P. Philipsen, C. Bahnsen, A. Møgelmose, T. B. Moeslund, and M. M. Trivedi, "Traffic light detection at night: Comparison of a learning-based detector and three model-based detectors," in *Advances in Visual Computing*, G. Bebis, R. Boyle, B. Parvin, D. Koracin, I. Pavlidis, R. Feris, T. McGraw, M. Elendt, R. Kopper, E. Ragan, Z. Ye, and G. Weber, Eds. Cham: Springer International Publishing, 2015, pp. 774–783.

[8] M. P. Philipsen, M. B. Jensen, A. Mgelmose, T. B. Moeslund, and M. M. Trivedi, "Traffic light detection: A learning algorithm and evaluations on challenging dataset," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, Sep. 2015, pp. 2341–2345.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 779–788.

[10] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *The 2011 International Joint Conference on Neural Networks*, July 2011, pp. 2809–2813.

[11] Y. Yang, H. Luo, H. Xu, and F. Wu, "Towards real-time traffic sign detection and classification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 2022–2031, July 2016.

[12] A. Pon, O. Adrienko, A. Harakeh, and S. L. Waslander, "A hierarchical deep architecture and mini-batch selection method for joint traffic sign and light detection," in *2018 15th Conference on Computer and Robot Vision (CRV)*, May 2018, pp. 102–109.
[13] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *CoRR*, vol. abs/1312.6199, 2013. [Online]. Available: http://arxiv.org/abs/1312.6199
[14] C. Sitawarin, A. N. Bhagoji, A. Mosenia, P. Mittal, and M. Chiang, "Rogue signs: Deceiving traffic sign recognition with malicious ads and logos," *CoRR*, vol. abs/1801.02780, 2018. [Online]. Available: http://arxiv.org/abs/1801.02780
[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.
[16] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.
[17] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017, pp. 39–57.
[18] Z. Wang, "The ssim index for image quality assessment," *https://ece. uwaterloo. ca/˜ z70wang/research/ssim*, 2003.
[19] L. Baccour and R. I. John, "Experimental analysis of crisp similarity and distance measures," in *2014 6th International Conference of Soft Computing and Pattern Recognition (SoCPaR)*. IEEE, 2014, pp. 96–100.