Adaptive Adversarial Videos on Roadside Billboards: Dynamically Modifying Trajectories of Autonomous Vehicles

Naman Patel[1], Prashanth Krishnamurthy[1], Siddharth Garg[2], Farshad Khorrami[1]

***Abstract*— Deep neural networks (DNNs) are being incorporated into various autonomous systems like self-driving cars and robots. However, there is a rising concern about the robustness of these systems because of their susceptibility to adversarial attacks on DNNs. Past research has established that DNNs used for classification and object detection are prone to attacks causing targeted misclassification. In this paper, we show the effectiveness of an adversarial dynamic attack on an end-to-end trained DNN controlling an autonomous vehicle. We launch the attack by installing a billboard on the roadside and displaying videos to approaching vehicles to cause the DNN controller in the vehicle to generate steering commands that cause, for example, unintended lane changes or motion off the road causing accidents. The billboard has an integrated camera estimating the pose of the on-coming vehicle. The approach enables dynamic adversarial perturbation that adapts to the relative pose of the vehicle and uses the dynamics of the vehicle to steer it along adversary-chosen trajectories while being robust to variations in view, lighting, and weather. We demonstrate the effectiveness of the attack on a recently published off-the-shelf end-to-end learning-based autonomous navigation system in a high-fidelity simulator, CARLA (CAR Learning to Act). The proposed approach may also be applied to other systems driven by an end-to-end trained network.**

I. Introduction

With the availability of massive amounts of data, faster compute and efficient learning algorithms, DNNs are increasingly being utilized in several domains like robotics, health-care, etc. However, trust and security remain a critical concern for deployment of these algorithms in the context of cyber-physical systems that interact with human beings.

DNNs are known to be vulnerable to adversarial perturbations, i.e., small but cleverly designed changes to the input which cause it to be mis-classified. Although initial work on generating adversarial attacks focused on generating adversarial attacks digitally (where the adversary directly manipulates the pixel) there has been recent work on generating real-world adversarial perturbations that target face recognition, classification and object detection systems.

DNNs have recently been used to implement end-to-end control policy for autonomous navigation. The DNN takes as input raw data from one or more sensors, for example a vision and/or LIDAR sensor, and outputs a speed and/or steering command for the robot. In this paper, we seek to investigate the susceptibility of these systems to adversarial perturbation attacks using autonomous driving as a specific instantiation of a robotic system controlled by a DNN.

We consider an attack scenario wherein the adversary places a billboard on the side of a road that can play videos (specifically modify displayed images at run-time). The adversary's objective is to cause an autonomous vehicle to deviate from its intended trajectory and follow an adversary-controlled trajectory. As illustrated in Figure 1, the attacker does so by displaying an adversarial *sequence* of images (i.e., an adversarial movie) on the billboard that continuously causes the vehicle's speed and heading to change as desired by the adversary. Compared to a static attack (i.e., displaying only a single image), the dynamic attack enables the attacker to control, in real-time, the vehicle's motion regardless of its distance and pose with respect to the billboard.

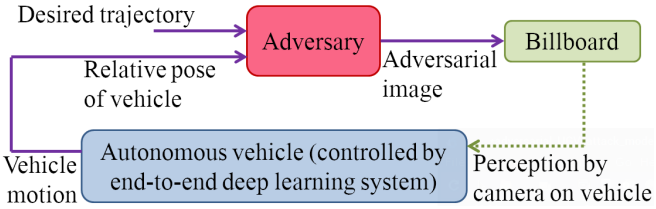

Fig. 1: Adversary model: adversary displays a sequence of images on a physical billboard in the environment to move the autonomous vehicle along an adversary-chosen trajectory.

The effectiveness of our algorithm is demonstrated in a high-fidelity simulated environment, CARLA [1], as it enables testing the robustness of our approach under various environmental conditions (e.g., lighting, weather, traffic). Our adversarial attack is applied to an autonomous vehicle trained using conditional-imitation learning [2] which takes as input the current image, destination and suggested action as input to generate a steering and speed command. The contributions of the paper are as follows:

- A framework to enable an adversary to dynamically modify the environment to cause the vehicle to move along an adversary-specified trajectory.
- A methodology to generate adversarial perturbations that are dynamic (temporally varying), adaptive (to vehicle relative pose and trajectory), and robust to various environmental changes and vehicle pose uncertainties.
- An iterative data generation policy for efficient dynamic adversarial attack on an autonomous vehicle.
- Demonstration of robustness of the proposed approach in multiple navigation scenarios with varying environmental conditions using a simulator.

[1]The authors are with the Controls/Robotics Research Laboratory (CRRL), Dept. of Electrical and Computer Engineering, NYU Tandon School of Engineering, 6 MetroTech Center, Brooklyn, NY 11201 USA. [2]The author is with the Center for Cyber-security (CCS), NYU Tandon School of Engineering, 370 Jay St., Brooklyn, NY 11201 USA. Part of this work was funded by NSF Grant 1801495. {nkp269, pk929, sg175, khorrami}@nyu.edu

II. Related Work

Deep neural networks (DNN) have been shown to achieve state-of-the-art performance for a range of tasks in computer vision, speech recognition, etc. [3], [4]. Motivated by these successes, there have been several efforts to use deep learning for autonomous navigation [2], [5]–[12] using imitation and reinforcement learning (RL) based approaches.

Despite their success, recent work has shown that DNNs can be surprisingly fragile. In particular, in their seminal paper, Szegedy et al. [13] show that DNNs are susceptible to so-called *adversarial perturbations*, small but targeted changes to the inputs of a DNN causing mis-classification. This was also demonstrated in our earlier work using adversarial attack on a UGV LIDAR [14]. It has also been shown that the adversarial perturbations are transferable (i.e., perturbations for one DNN can be applied to a different DNN trained for the same task), generated in black-box settings [15], [16] and can be deployed in real-world settings to attack face recognition [17], image classification [18]–[20] and object detection [21], [22] systems. These attacks have also been shown to make RL based systems vulnerable [20], [22]. Despite several efforts to develop robust defenses [23], [24] against these perturbations, guaranteed defenses that work for large-scale DNNs have remained elusive.

Most closely related to our work is [18], which shows that specially designed stickers pasted on traffic signs can cause the signs to be mis-classified. In comparison, our attack is on a DNN that performs a regression task (i.e., the output of our DNN is the vehicle heading and speed, which are analog outputs), and our attack is *dynamic* in that it responds to the victim autonomous vehicle's movement. Using dynamic perturbations, we enable the attacker to dynamically control the victim vehicle's trajectory which would not be possible using a single static perturbation.

III. Dynamic Physical Adversarial Attack

A. Problem Formulation

As shown in Figure 1, the adversary's objective is to cause the autonomous vehicle to move along a desired trajectory (instead of its original trajectory). Given a desired trajectory and a starting pose of the vehicle (relative to the billboard), the adversary generates a *sequence* of adversarial images that adapt to the vehicle's evolving relative pose; the evolution of the vehicle's relative pose, in turn, depends on the adversarial image displayed on the billboard, the generated actuation commands from the vehicle's end-to-end learning-based system, and on the vehicle dynamics. In particular, at each step, the adversary determines the necessary vehicle actuation (specifically, the steering angle) given the desired motion trajectory and computes a billboard image that causes the vehicle's navigation DNN to output the desired actuation. It is to be noted that the adversary can only cause a physical modification in the environment (displaying an image on the physical billboard) but does not have direct access to the sensor input on the vehicle. Hence, the image actually seen by the vehicle's camera could be significantly different from what the adversary intends due to varying environmental conditions (lighting, weather, etc.) and differences in the vehicle's relative pose. The adversary, therefore, seeks to design perturbations that are robust against environmental and pose variations. We found, empirically, that building in this robustness is critical to the attack's success.

B. Approach

The overall architecture of our approach is shown in Figure 2. The optimizer seeks to find a matrix M, which represents the image to be displayed on the billboard. When viewed from the vehicle's camera, the image-space location of the billboard is dependent on the vehicle's pose relative to the billboard. Denoting this relative pose as p, the image formed by superimposing the matrix M onto the appropriate region of the image C observed by the vehicle's camera at that location is given by a (pose-dependent) function of form $f_p(C, M)$. It is to be noted, as discussed in Section III-A, the actual image $\overline{C}$ seen by the vehicle's camera could be significantly different from the synthetically constructed superimposition $f_p(C, M)$. To address this difference (i.e., that the adversary cannot directly modify the sensor input) and build robustness into the matrix M, a family $\mathcal{T}$ of transformations (modeled as a distribution) is considered that include variations in pose, lighting, and visibility.

These transformations are generated by taking into account variabilities occurring due to noise from the camera as well as the environment. To make the system invariable to small movements, the transformation distribution consists of translational affine transformations ($\pm 5\%$) in the image domain and Gaussian blur (kernel size up to 5) of billboard to emulate the camera view from farther distance. The adversarial billboard should also be robust to lighting changes caused due to the sensing limits of the camera as well as the weather/environmental conditions. This is mitigated by having individual channel-based as well as full image-based additive (± 15) and multiplicative (0.75 to 1.1) lightening/darkening. The lightening/darkening is performed on billboard and background individually. Thus, the input image at a particular pose is transformed and optimized over a distribution of lighting, pose, and visibility changes.

Denoting one such transformation in $\mathcal{T}$ as T, the adversary effectively attempts to make $N(T(f_p(C, M)))$ close to s_{des}, where s_{des} is the desired adversarial vehicle steering angle at that time instant, and N denotes the vehicle's navigation DNN mapping camera images to steering angles which the adversary has access to. To address the distribution $\mathcal{T}$ of transformations and generate a matrix M, robust to these transformations, the following loss function is optimized:

$$L(M) = \mathbb{E}_{T\sim\mathcal{T}}\{|N(T(f_p(C, M))) - s_{des}|\} + R(M) \quad (1)$$

where the expectation $\mathbb{E}\{.\}$ is computed over the distribution $\mathcal{T}$, $|.|$ denotes a suitable norm (e.g., L_2), and $R(M)$ is a regularizer that smooths the matrix M (removing pixelations to increase the rate of attack [17], [21]) by penalizing spatial rate of variations in M (i.e., total variation or TV loss). More generally, while (1) considers one camera image C obtained at one relative pose p, a set of camera images C_i obtained at various relative poses p_i are considered in the loss function

$$L(M) = \sum_i \mathbb{E}_{T\sim\mathcal{T}}\{|N(T(f_{p_i}(C_i, M))) - s_{des,i}|\} + R(M) \quad (2)$$

where $s_{des,i}$ are the desired adversarial steering angles at each relative pose. In our implementation, the set of relative

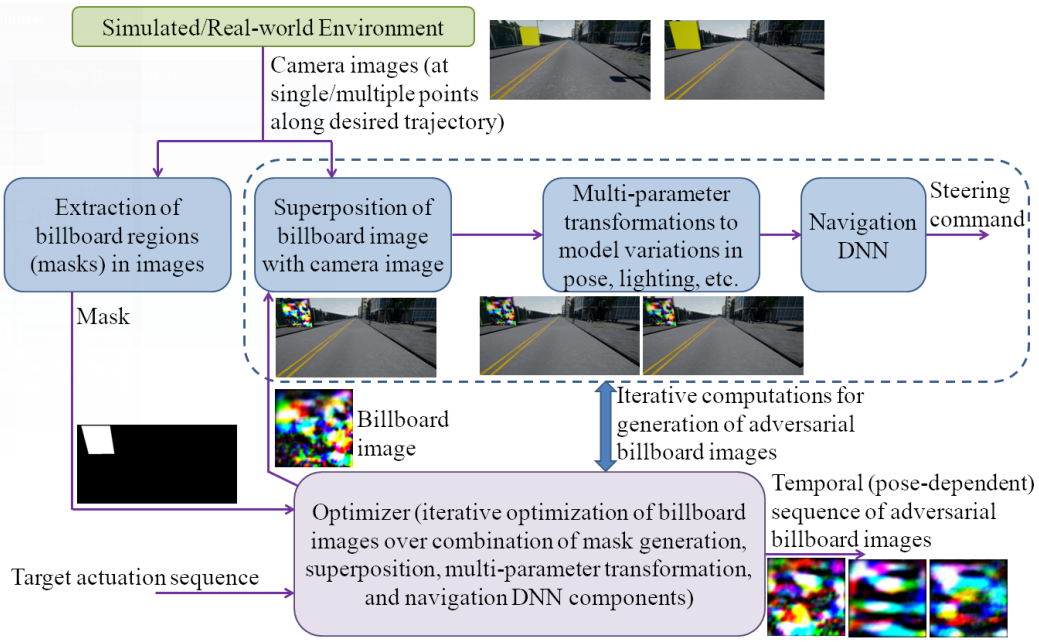

Fig. 2: Overall architecture of the proposed approach for dynamic adversarial perturbation for real-time adversary-controlled steering of an autonomous vehicle.

poses p_i is iteratively constructed starting from the initial pose by generating an adversarial perturbation in each step and computing the vehicle's new pose in response to these perturbations. This process iteratively converges to a final set of (pose-dependent) adversarial images that cause the vehicle to move along the desired adversarial trajectory.

The loss function in (2) is optimized through projective gradient descent [25]. The adversarial perturbation on M is initialized to zero and is then optimized through projected gradient descent of a fixed step-size (e.g., 5) using an Adam optimizer. In each iteration, an image C_i is drawn randomly from the image set and the matrix M is refined (starting from the matrix obtained in the previous iteration) to minimize the loss $L(M)$ computed for that image C_i. Thus, the overall loss (2) is effectively minimized by randomly cycling through the set of images C_i and randomly sampling sets of transformations T during each iteration.

IV. Experimental Results

A. Simulation Testbed

We evaluate our dynamic attack framework using an Unreal Engine 4 based simulator, CARLA [1], which is specifically designed for testing autonomous navigation algorithms. The engine provides high-fidelity rendering quality and realistic physics as well as NPC (Non-Player Character) logic. The environment in CARLA is a mini-town with adjustable weather and is composed of buildings, vegetation, traffic signs, and infrastructure. Thus, it is an ideal platform for testing the robustness of our approach.

The controllable agent in the simulator is a car with sensors for perception and depth which can be controlled by changing its steering, throttle, and brake. The car's camera is only used for our autonomous navigation task. Various parameters of the camera can be changed such as field of view, resolution, location and orientation with respect to the car's coordinate system, frame rate, and depth of view. The camera also post-processes the image through effects such as vignette, grain jitter, bloom auto exposure, lens flares, and depth of field, through the simulator. The post-processing effects greatly enhance the realism of the simulator through the transformation added between the environment and the camera, to mimic real-world effects dependent on the environment and the camera mechanism.

The DNN used to test our adversarial attack system is a conditional imitation learning network from the recent work in [2] for navigation. The network takes as input the current image, recommended command or vector to goal, and the current speed of the car to compute the steering and velocity commands for the next state. The resolution of the image is 200x88 pixels and the network consists of an image module, measurement module, and command module. The final layers of these modules are combined to output the steering and velocity commands. The steering and velocity commands are output as a percentage [-100,100] of maximum steering and speed. Details on the architecture and training procedure of the DNN can be found in [2]. A crucial reason for selecting this DNN for the vehicle's navigation was that this network was trained to recover from failure cases via data augmentation and noise injection to induce abnormal vehicle states, thus making the network robust to failures. However, as seen in Section IV-B, this network is vulnerable to attacks generated by our attack framework.

B. Results for Sample Scenarios

The experimental results below address the evaluation of our proposed attack framework for different navigation scenarios such as straight motion in a lane and turning. In addition, we also evaluate the robustness of our attack to varying environmental conditions such as changing weather and traffic. We also show through our experiments that

for some scenarios, it is possible to generate a desired adversarial trajectory with a static image on the billboard.

The scenarios considered are: (1) attacking a car moving in a straight trajectory to cause it to abruptly turn left, i.e., to move onto the wrong side (left side) of the road; (2) attacking a car making a right turn to cause it to instead move straight.

We achieve scenario (1) by displaying a temporal sequence of adversarial images on the billboard depending on the relative pose of the car with respect to the billboard. As seen from the top rows of Figure 3 and Figure 4 in first and third person views, the car is able to safely navigate in a straight trajectory in the absence of the sequence of adversarial billboard images. The first person (vehicle's camera) and third person (view from behind the vehicle in the simulator) views of the attack to generate the desired trajectory can be seen in the middle rows of Figure 3 and Figure 4, respectively. The robustness of the adversarial perturbation to environmental changes is tested by displaying the same sequence of adversarial billboard images during heavy rain with traffic (in the form of NPC vehicles). This sequence of adversarial billboard images is able to make the car change lanes, which results in a crash with another car and a two-wheeler as can be seen in first person and third person views in the last rows of Figure 3 and Figure 4.

In scenario (2), as shown in the top rows of Figure 7 and Figure 8, the car is autonomously making a right turn. The desired adversarial objective is to move the car in a straight trajectory towards the poster instead of making the right turn. We demonstrate, as seen in the middle rows of Figure 7 and Figure 8, that the car is successfully able to follow the desired trajectory with just a *single* adversarial billboard image (note that the previous attack scenario did not succeed with a single image, demonstrating the need for adversarial videos, in general). We test the robustness of our approach in a low light (during dusk) environment. The car is still able to follow the desired trajectory of moving towards the poster as shown in the first person and third person views in the last rows of Figure 7 and Figure 8.

The X-Y trajectories and steering angle commands for the normal/adversarial scenarios are shown in Figures 3–8 and in Figures 5–6. As observed in Figures 3–8, the introduction of the adversarial perturbations causes modifications of the vehicle actuation commands (steering angles) to cause it to move along the desired adversarial trajectories (i.e., moving to the wrong side of the road for the straight motion scenarios and moving straight instead of turning for the right turn scenarios). These results demonstrate that our dynamic physical adversarial attack framework is successfully able to cause the autonomous vehicle to move along adversary-desired trajectories in different scenarios with changing environmental conditions such as weather, lighting, and traffic.

V. Conclusion

Thus, a robust attack on autonomous navigation system running on a state-of-the-art off-the-shelf DNN, was presented and demonstrated on a high-fidelity simulator, CARLA. The attack makes use of a billboard mounted by the side of a road that observes the relative pose of a vehicle and displays appropriate adversarial images to make it follow an adversary-chosen trajectory. Directions for future research include extensions of the proposed methodology to black-box settings and evaluations of the possibility of detecting these physical perturbations through methods such as our prior work on on-line process-aware monitoring [7] for verification of the overall dynamics of the system.

References

[1] A. Dosovitskiy, G. Ros, F. Codevilla, A. López, and V. Koltun, "CARLA: an open urban driving simulator," in *Proceedings of the Annual Conference on Robot Learning*, Mountain View, California, Nov. 2017, pp. 1–16.

[2] F. Codevilla, M. Miiller, A. López, V. Koltun, and A. Dosovitskiy, "End-to-end driving via conditional imitation learning," in *Proceedings of the International Conference on Robotics and Automation*, Brisbane, Australia, May 2018, pp. 1–9.

[3] "ImageNet large scale visual recognition competition," http://www.image-net.org/challenges/LSVRC/2012/, 2012.

[4] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013, pp. 6645–6649.

[5] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[6] Y. LeCun, U. Muller, J. Ben, E. Cosatto, and B. Flepp, "Off-road obstacle avoidance through end-to-end learning," in *Proceedings of the Advances in Neural Information Processing Systems NIPS*, Vancouver, Canada, Dec. 2005, pp. 739–746.

[7] N. Patel, A. N. Saridena, A. Choromanska, P. Krishnamurthy, and F. Khorrami, "Adversarial learning-based on-line anomaly monitoring for assured autonomy," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, Madrid, Spain, Oct. 2018, pp. 6149–6154.

[8] N. Patel, P. Krishnamurthy, and F. Khorrami, "Semantic segmentation guided slam using vision and lidar," in *Proceedings of the of the 50th International Symposium on Robotics*, Munich, Germany, Jun. 2017, pp. 352–358.

[9] L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, "Lidar-based driving path generation using fully convolutional neural networks," in *Proceedings of the International Conference on Intelligent Transportation Systems*, Yokohama, Japan, Oct. 2017, pp. 1–6.

[10] N. Patel, P. Krishnamurthy, Y. Fang, and F. Khorrami, "Reducing operator workload for indoor navigation of autonomous robots via multimodal sensor fusion," in *Proceedings of the Companion of the ACM/IEEE International Conference on Human-Robot Interaction*, Vienna, Austria, Mar. 2017, pp. 253–254.

[11] N. Patel, A. Choromanska, P. Krishnamurthy, and F. Khorrami, "Sensor modality fusion with cnns for UGV autonomous driving in indoor environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, Vancouver, Canada, Sept. 2017, pp. 1531–1536.

[12] M. Wulfmeier, D. Rao, D. Z. Wang, P. Ondruska, and I. Posner, "Large-scale cost function learning for path planning using deep inverse reinforcement learning," *International Journal of Robotics Research*, vol. 36, no. 10, pp. 1073–1087, 2017.

[13] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *Proceedings of the International Conference on Learning Representations, ICLR*, Banff, Canada, Apr. 2014.

[14] N. Patel, K. Liu, P. Krishnamurthy, S. Garg, and F. Khorrami, "Lack of robustness of lidar-based deep learning systems to small adversarial perturbations," in *Proceedings of the of the 50th International Symposium on Robotics*, Munich, Germany, Jun. 2017, pp. 359–365.

[15] W. Brendel, J. Rauber, and M. Bethge, "Decision-based adversarial attacks: Reliable attacks against black-box machine learning models," in *Proceedings of the International Conference on Learning Representations*, Vancouver, Canada, Apr. 2018.

[16] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin, "Black-box adversarial attacks with limited queries and information," in *Proceedings of the International Conference on Machine Learning*, Stockholm, Sweden, Jul. 2018, pp. 2142–2151.

[17] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proceedings of the Conference on Computer and Communications Security*, Vienna, Austria, Oct. 2016, pp. 1528–1540.

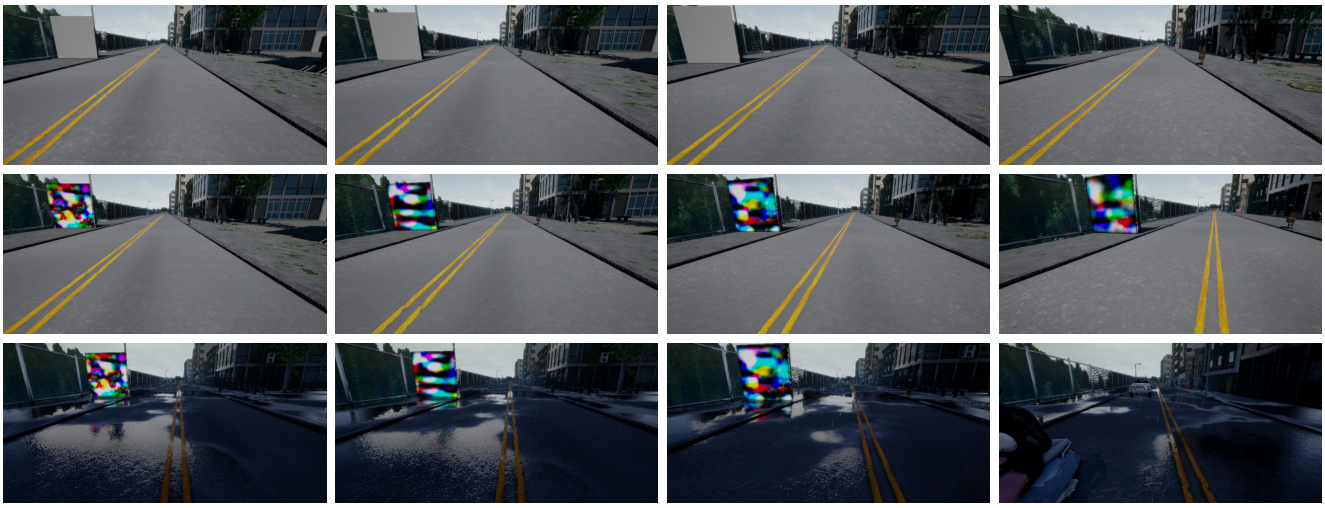

Fig. 3: First person images (i.e., from the vehicle's camera) of several normal/adversarial scenarios in a straight road segment.The top row shows the the vehicle staying in its lane when the billboard on the left-side pavement displays a blank image. In the middle row, the adversary displays a sequence of images on the billboard to make the vehicle turn left. In the bottom row, the weather condition in the simulator is changed to rainy and NPC vehicles are added on the other side of the road; the adversary displays the same sequence of images and it is seen that a similar vehicle motion is observed as in the middle row. Furthermore, as the vehicle passes the billboard while on the wrong side of the road, the vehicle collides head-on with an oncoming NPC vehicle.

Fig. 4: Third person views for the normal/adversarial scenarios shown in Figure 3. Each row of pictures above shows the views from behind the autonomous vehicle in the simulator for each corresponding row in Figure 3.

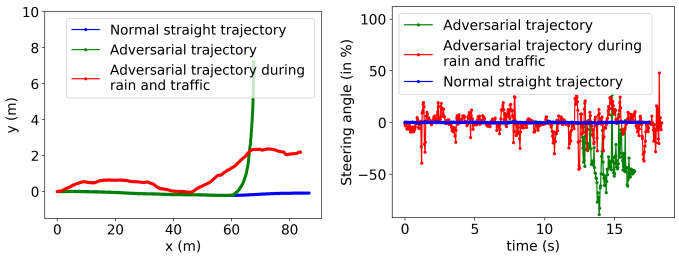

Fig. 5: Trajectories (left) and steering commands (right) of the vehicle in the normal/adversarial scenarios shown in Figures 3 and 4. While the trajectory under normal conditions is appropriately straight, the trajectories with adversarial billboard shows motion towards the wrong side of the road.

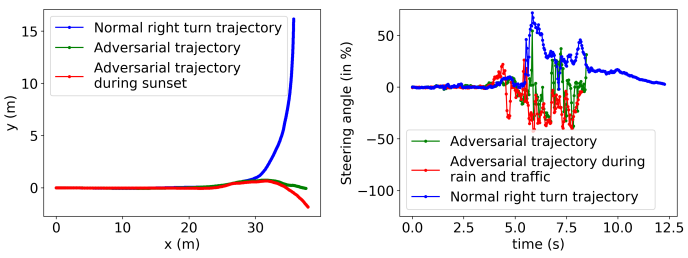

Fig. 6: Trajectories (left) and steering commands (right) of the vehicle in the normal/adversarial scenarios shown in Figures 7 and 8. While the trajectory under normal conditions involves the vehicle making a right turn at the intersection, the adversarial billboard causes the vehicle to move in the wrong direction.

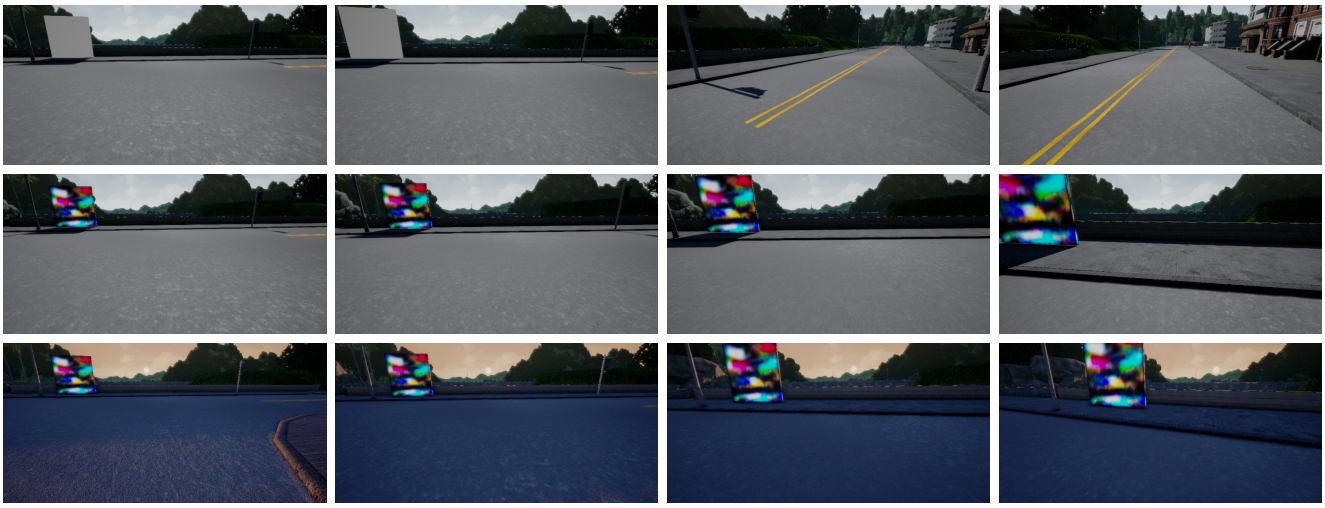

Fig. 7: First person images of several normal/adversarial scenarios in a road segment where the autonomous vehicle is making a right turn at an intersection. As shown in the top row, the vehicle takes a right turn when the billboard mounted on the far side of the intersection is displaying a blank image. In the middle row, the adversary displays an image on the billboard to make the vehicle move straight instead of turning right. In the bottom row, the weather condition in the simulator is changed to low-light (during dusk); the adversary displays the same image as in the middle row and it is seen that a similar vehicle motion is observed.

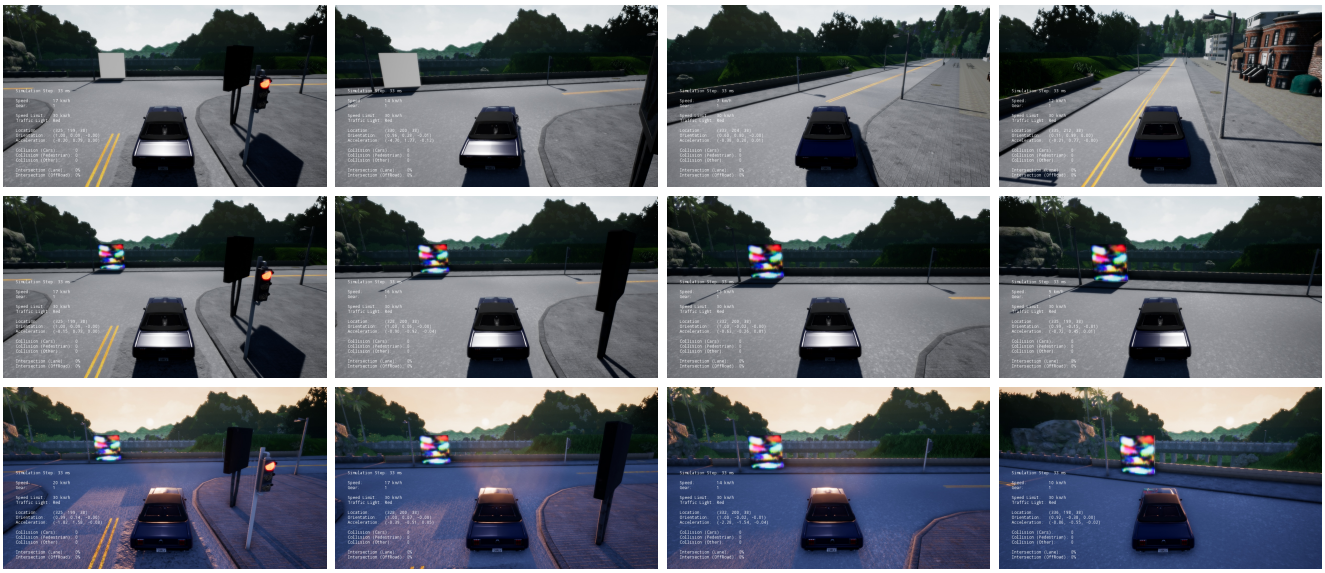

Fig. 8: Third person views for the normal/adversarial scenarios shown in Figure 7. Each row of pictures above shows the views from behind the autonomous vehicle in the simulator for each corresponding row in Figure 7.

[18] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, Jun. 2018, pp. 1625–1634.

[19] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *Proceedings of the International Conference on Machine Learning*, Stockholm, Sweden, Jul. 2018, pp. 284–293.

[20] Y. Huang and S. Wang, "Adversarial manipulation of reinforcement learning policies in autonomous agents," in *Proceedings of the International Joint Conference on Neural Networks*, Rio de Janeiro, Brazil, July 2018, pp. 1–8.

[21] D. Song, K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, F. Tramèr, A. Prakash, and T. Kohno, "Physical adversarial examples for object detectors," in *Proceeddings of the USENIX Workshop on*

[22] S. Chen, C. Cornelius, J. Martin, and D. H. P. Chau, "Shapeshifter: Robust physical adversarial attack on faster R-CNN object detector,"

Offensive Technologies, WOOT, Baltimore, MD, USA, Aug. 2018.
in *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, Dublin, Ireland, Sept. 2018, pp. 52–68.

[23] A. Raghunathan, J. Steinhardt, and P. Liang, "Certified defenses against adversarial examples," in *Proceedings of the International Conference on Learning Representations, ICLR*, Vancouver, BC, Canada, May 2018.

[24] E. Wong and J. Z. Kolter, "Provable defenses against adversarial examples via the convex outer adversarial polytope," in *Proceedings of the International Conference on Machine Learning, ICML*, Stockholm, Sweden, Jul. 2018, pp. 5283–5292.

[25] N. Carlini and D. Wagner, "Adversarial examples are not easily detected: Bypassing ten detection methods," in *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, Dallas, Texas, USA, Nov. 2017, pp. 3–14.