

1 **IMPACT ANALYSIS OF INFERENCE TIME ATTACK OF PERCEPTION SENSORS ON**
2 **AUTONOMOUS VEHICLES**

3
4
5
6 **Hanlin Chen, Corresponding Author, ORCID: 0000-0001-6508-7715**

7 Lyles School of Civil Engineering, Purdue University
8 550 Stadium Mall Drive, West Lafayette, IN 47907
9 Email: chen1368@purdue.edu

10
11 **Simin Chen**

12 Department of Computer Science, The University of Texas at Dallas
13 800 W Campbell Rd, Richardson, TX 75080
14 Email: sxc180080@utdallas.edu

15
16 **Wenyu Li**

17 Department of Civil and Environmental Engineering, The Hong Kong University of Science and
18 Technology
19 Hong Kong 999077, China
20 Email: wenyu.li.cn@gmail.com

21
22 **Wei Yang, Ph.D.**

23 Department of Computer Science, The University of Texas at Dallas
24 ECSS 4.225, 800 W. Campbell Rd., Richardson, TX 75080
25 Email: wei.yang@utdallas.edu

26
27 **Yiheng Feng, ORCID: 0000-0001-5656-3222**

28 Lyles School of Civil Engineering, Purdue University
29 550 Stadium Mall Drive, West Lafayette, IN 47907
30 Email: feng333@purdue.edu

31
32
33 Word Count: 5260 words + 0 table(s) \times 250 = 5260 words

34
35
36
37
38
39
40 Submission Date: March 29, 2025

1 ABSTRACT

2 As a safety-critical cyber-physical system, cybersecurity and related safety issues for Autonomous
3 Vehicles (AVs) have been important research topics for a while. Among all the modules on AVs,
4 perception is one of the most accessible attack surfaces, as drivers and AVs have no control over
5 the outside environment. Most current work targeting perception security for AVs focuses on
6 perception correctness. In this work, we propose an impact analysis based on inference time attacks
7 for autonomous vehicles. We demonstrate in a simulation system that such inference time attacks
8 can also threaten the safety of both the ego vehicle and other traffic participants.

9

10 *Keywords:* Autonomous Vehicle, Cybersecurity, Perception security, Cyber-physical system

1 INTRODUCTION

2 Autonomous vehicles shed light on the development of smart transportation system as they can
3 improve the safety, mobility and energy efficiency, as shown in the report issued by USDOT (1).
4 One critical feature of autonomous vehicles, comparing to their human counterpart, is that they
5 have a very short response time between an event and the vehicle reaction. The processing time
6 for the perception-decision-response process for the human-driven vehicle is determined by human
7 response time, which can be affected by driver's distraction (2), age (3) and the possibility of DUI
8 (4). For autonomous vehicles, the computer-vision-based system has a shorter response time (5),
9 and the reaction time is mainly determined by the mechatronic properties of the vehicle itself.

10 Given the fact that autonomous vehicles are safety-critical cyber-physical systems, there
11 has been a lot of research work targeting the vulnerabilities within the system. Lot of research on
12 vulnerabilities in computer vision (6–8) is also applicable for launching an attack on perception
13 module of autonomous vehicles (9–11). Generative adversarial examples can also enable generat-
14 ing perturbations that slowdown the inference speed of deep-learning modules, as shown in (12–
15 15). For example, if a neural network normally inference around 0.1 second(which is a common
16 standard for autonomous driving perception system), when given some attack data, some neural
17 networks' inference speed can be greatly reduced. As our work shows in section 7.1, the vitim
18 model will have an average inferece time around 0.1 seconds. But, when under attack its inference
19 time goes up to 3 seconds. This means when the inference process finished, the vehicle though
20 it is responding to the most current environment state, which was actually happened 3 seconds
21 ago. This is way too slow for safety-critical cyber-physical system like autonomous vehicles. In
22 this work, we perform an initial impact analysis of inference time attack on autonomous vehicle's
23 perception system. As the first target, we apply such an attack towards an autonomous vehicle at
24 a signalized intersection, since in such scenario, the perception system of the autonomous vehicle
25 not only need to detect other road users, but also the traffic signals.

26 With such a threat model, the attack goal in this paper is to maximize the degradation of
27 the performance of an autonomous vehicle in the intersection navigation scenario. The attacker's
28 primary target is to downgrade the autonomous vehicle's safety performance while also seeking to
29 compromise mobility and comfort. To achieve the attack goal, smart selection of attack policy is
30 essential, which is the main contribution of this paper.

31 The contribution and novelty of our work are shown below:

- 32 • We are the first to study the effect of inference time attacks on the perception module
33 of autonomous vehicles. Specifically, we find that the optimal attack policy requires a
34 careful selection of both attack intensity and the launch time. This attack is different
35 from the DoS attack occurring in V2V communication shown in previous work (16), as
36 the information retrieval from the environment is not completely blocked out and we also
37 involve the prediction module in state-of-the-art middleware system, which is designed
38 to mitigate the information retrieval delay from perception.
- 39 • We performed end-to-end evaluation on both inference time only and performance from
40 vehicle side and transportation engineering side. We showed that such attack can threaten
41 both the ego vehicle and other traffic participants as well.

42 The structure of this paper is as follows: Section 3 shows current work regarding au-
43 tonomous vehicle perception time and inference time attack. Section 4 shows the threat model
44 in this work, mainly about attacker assumption and attacker's goal. Section 5 shows how the attack
45 is formulated and how adversarial example is obtained. Section 6 shows the impact analysis with

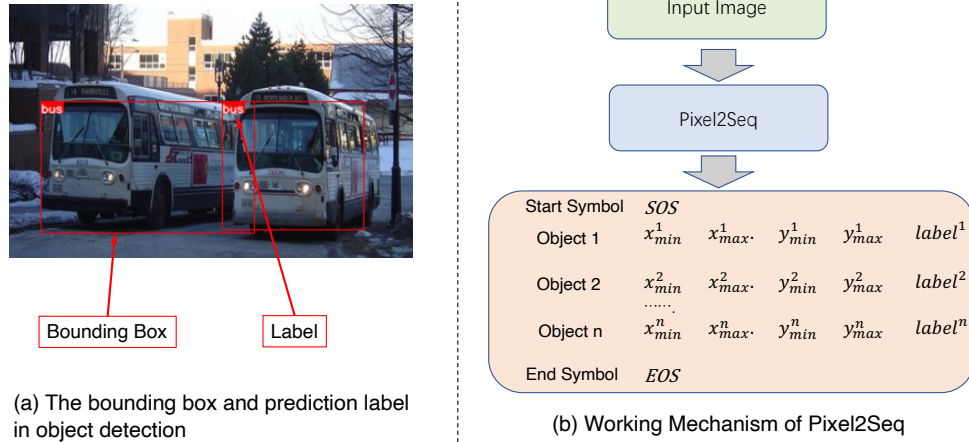


FIGURE 1: Problem definition of Object detection and the working mechanism of Pixel2Seq

1 end-to-end simulation system, on inference only, vehicle and transportation side. Section 8 shows
 2 potential future work with regard to defense, and section 9 concludes this work.

3 LITERATURE REVIEW

4 The perception and control of autonomous vehicles

5 Autonomous vehicles use the information obtained from sensors to make control decisions. In this
 6 paper, we focus on attacking the object detection perception module for the following reasons:
 7 (1) the object detection module plays a vital role in autonomous vehicles; (2) the object detection
 8 module is the most widely deployed in autonomous vehicles (17).

9 As shown in Figure 1 (a), given one input image, the object detection module seeks to
 10 output the bounding box and the class labels of each object in the image. Here, we briefly in-
 11 troduce several working mechanisms of the neural networks-based object detection module. The
 12 first type is R-CNN, the masterpiece of two-stage object detection neural networks. In the first
 13 stage, R-CNN applies its region extraction neural networks to generate object region proposals. In
 14 the second stage, R-CNN uses its feature extraction neural networks to extract features for each
 15 extracted object region and uses an SVM to classify the category of each region. The second
 16 type is YOLO, which combines the stage of object region proposal and region classification into
 17 one neural network. YOLO applies only one network to divide the input image into regions and
 18 predicts bounding boxes and probabilities for each region. Recently, state-of-the-art, Pixel2Seq,
 19 applies a language model (LM) to cast the object detection task. Object descriptions (*e.g.*, coor-
 20 dinate of the bounding boxes and class labels) are expressed as sequences of discrete tokens, and
 21 the neural networks are trained to generate the desired sequence. Figure 1(b) shows the working
 22 mechanism of Pixel2Seq. The backend neural networks start with the start symbol (*i.e.*, SOS) and
 23 iteratively compute the coordinate of the bounding boxes and class labels for each object until the
 24 output reaches the end symbol (*i.e.*, EOS). From the working mechanism of Pixel2Seq, an impor-
 25 tant observation is that Pixel2Seq will not stop computation until its output is EOS. Such property
 26 brings in a new vulnerability: the attacker can craft an adversarial example to make the output of
 27 Pixel2Seq not reach EOS and increase the response latency.

28 For autonomous vehicles, we inherit the assumption that for perception inference time,

the inference interval is about 0.1s. The such setting had been applied in a previous study for information retrieval (16) and also in the University of Michigan’s Safety Pilot Model Deployment program for data acquisition(18), which is way shorter than human drivers’ perception response time (3). Based in that assumption, Pixel2seq is possible for application on AV or ADAS systems as it can performance pretty well considering both inference time and inference accuracy. We show in section 7.1 for baseline system’s inference performance and section 7.2 for baseline end2end system response. Therefore, based on the two baseline evaluations, our target model can be used when not under attack.

Inference time attack in CPS system

In recent times, a remarkable emergence of inference time attacks has drawn significant attention, as these attacks are specifically designed to rigorously assess the robustness of cyber-physical systems (CPS). The sensitivity of CPS to latency makes these inference time attacks particularly concerning, as they can lead to serious consequences with far-reaching implications.

Inference time attacks (14, 19) involve the deliberate crafting of adversarial samples, aimed at consuming significantly more computational resources from the victim system. As a result, these attacks cause a substantial increase in the system’s response time, introducing delays and disruptions that can have critical impacts on real-world scenarios. The potential ramifications of such attacks are extensive, impacting various sectors where CPS plays a crucial role. For instance, a prominent example of an inference time attack is SlowLiDAR (19), which targets the LiDAR-Based Detection module in CPS. SlowLiDAR has been shown to increase the victim’s LiDAR system response latency by an alarming rate of up to more than 3000%. This escalation in latency could be detrimental, especially in applications like autonomous vehicles, where timely and accurate sensor data processing is essential for safe navigation. Another noteworthy example is EfficFrog (20), which focuses on attacks against the dynamic image reception module in CPS. Unlike some other attacks that compromise accuracy, EfficFrog cunningly impacts only the victim’s inference latency while having a minor impact on its accuracy. This stealthiness enhances the attacker’s ability to exploit vulnerabilities without raising immediate suspicion, making it a potent threat.

Given the pivotal roles that Cyber-Physical Systems (CPS) play in various domains, the potential consequences of inference time attacks can indeed be far-reaching. Consequently, evaluating the robustness of CPS under such attacks holds tremendous significance.

THREAT MODEL

Attacker’s Goal

In this research, our primary objective is to investigate the manipulation of the inference time of a targeted autonomous vehicle’s (AV) perception module, which can have serious implications on its security (19, 20). Our study revolves around the creation of adversarial perturbations specifically designed to increase the inference time of the AV’s perception module, potentially leading to disruptions in its essential functions. The introduction of adversarial perturbations to the AV’s perception module aims to impede its accurate and efficient processing of incoming data. This could result in delays in decision-making, compromising the overall safety and reliability of the AV’s operations. The impact of such attacks can directly lead to several safety hazards in real-world scenarios: (1) Vehicle collisions: If the inference time of the targeted AV increases suddenly, it may not be able to apply timely braking or make appropriate maneuvers, increasing the risk of col-

lisions with other vehicles or obstacles, and (2) Violation of traffic rules and public security issues: The compromised perception module might fail to detect traffic signals, pedestrian crossings, or road signs, leading to the AV breaking traffic rules and posing security risks to the public.

Attacker’s Assumption

In this work, we assume that the attacker has some knowledge about the autonomous vehicle’s perception module and has already given a set of trojan triggers with different attack intensities. In this specific scenario, the stronger attack intensity refers to the longer inference time delay caused by the attack. The attacker can also launch the attack with a selected starting time. For example, place the trigger on a billboard or put it as a sticker on the side of the road. We inherit the attack assumption in work (15). Once the attack intensity is determined, the attacker can use a universal trojan trigger. Thus the attacker does not have to generate different perturbations given each frame of perception data.

Our primary focus lies within the white-box attack setting, wherein the attacker possesses complete knowledge of the perception module utilized in the target Autonomous Vehicle (AV) systems. This assumption of having full access to the system’s internals is consistent with prior adversarial attacks on AVs relying on camera or LiDAR technology (19, 21). To achieve this white-box scenario, the attacker can acquire a victim AV through purchase or rental and subsequently engage in reverse engineering of its perception module. The feasibility of such reverse engineering has been demonstrated in the case of Tesla Autopilot, indicating the plausibility of accessing and comprehending the inner workings of the AV’s perception module. By exploring the white-box attack setting in this manner, we aim to gain deeper insights into the vulnerabilities of AV systems and devise more robust defense strategies to enhance their security against potential adversarial threats.

In addition to the aforementioned scenarios, we also consider the possibility that the attacker can covertly paste an adversarial perturbation on the victim’s camera. This assumption is consistent with existing research and holds significant practical relevance in real-world situations (22, 23). For example, the attacker could secretly affix a sticker or a specially crafted object to the camera when the victim vehicle is parked in a parking lot or any other vulnerable location.

ATTACK FORMULATION

Problem Formulation

This paper is focused on attacking the perception modules of autonomous vehicles, with a specific target being the state-of-the-art object detection neural networks such as Pixel2Seq. Models such as Pixel2Seq formulate the object detection problem as a sequential generation task and employ neural networks to accomplish this task. The key characteristic of the sequential generation model is its iterative generation of discrete tokens to represent bounding boxes and class labels for each detected object. The process continues until the end of the sequence (EOS) symbol is reached, indicating the end of the generation. In this kind of model, the inference process will not terminate until EOS occurs, thus our attack aims to minimize the likelihood of the EOS symbol, compelling the model not to terminate prematurely. In essence, we want to prolong the object detection process to disrupt the network’s performance. Formally, our attack can be formulated as the following optimization problem:

$$\begin{aligned} & \Delta = \operatorname{argmax}_{\delta} \text{Latency}_{\mathcal{F}}(x + \delta) \\ \text{s.t. } & \|\delta\| \leq \varepsilon \wedge \|x + \delta\| \in [0, 1]^n \end{aligned} \quad (1)$$

where the x in the original input captured by the camera, δ is the optimal adversarial perturbation under solving, $\text{Latency}_{\mathcal{F}}(\cdot)$ is the function that measures the object detection neural networks latency. The constraint $\|\delta\| \leq \varepsilon$ limits the size of the adversarial perturbation and makes the perturbation unnoticeable, and $\|x + \delta\| \in [0, 1]^n$ limits the adversarial examples should be a realistic one. Following existing work (8), we set unnoticeable perturbation size constraint ε as 0.03; such a setting can ensure the adversarial perturbations are unnoticeable.

EVALUATION

Simulation System Setup

Two kinds of vehicles will be discussed in all scenarios, Vehicle Under Test (VUT) and Background Vehicle (BV). VUT here refer to the autonomous vehicle, and BV refer to other traffic participants that can have impact and interact with VUT. In this paper, VUT is an autonomous vehicle where all information about surrounding vehicles is detected from its perception module.

The evaluation platform consists of a multi-resolution simulation system composed of a perception module, a background vehicle controller, and middleware. In this system, the perception module relies on the CARLA simulation, which generates perception data for the VUT. The movements of the background vehicle and VUT are reflected in CARLA, impacting the corresponding scenarios and changing the perception data. The background vehicle is controlled by SUMO (24), a microscopic traffic simulator, which follows traffic rules and generates traffic signals in the traffic network.

In this work, we use Robot Operating System(ROS) as a standardized middleware, providing the functionality of generating vehicle planning and control given the inferred perception information. We chose ROS because it is widely applied in autonomous driving systems such as Autoware (25) and Apollo (26). We implemented the perception, prediction and control module within ROS to test the impact of inference time attack on both vehicle side and transportation side. The Pix2seq algorithm is implemented as the inference model for perception, taking perception data from CARLA and outputting the inference information for both background vehicles and traffic lights. This inference information serves as the input for the VUT middleware system.

For evaluation, we let VUT interact with background vehicle and traffic signal without launching inference time attack. This is to indicate that the functionality of perception, prediction and control module we implemented is fine. Then we launch the inference time attack with the same background setting for VUT, trying to see the impact of attack. A figure of the evaluation platform is shown below.

Experiment Setup

Following other works relate to AV cybersecurity, In our experiment, perception data is generated in CARLA and background vehicle is generated and controlled by SUMO. Two simulators are synchronized and the AV in experiment is controlled by ROS where an optimal controller is used to generate longitudinal vehicle speed control command. All background vehicles will follow the IDM car-following model (27). Parameters of the IDM car-following model is determined by SUMO. All background vehicle will disable the emergency collision avoidance function.

All scenarios are reconstructed/build in CARLA simulator and the perception data is also

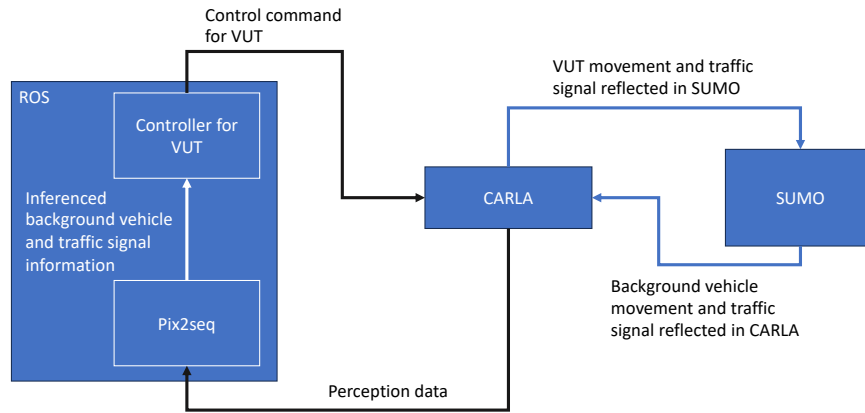


FIGURE 2: System architecture for end-to-end evaluation

1 generated by CARLA. The evaluation scenario is in Town04 map provided by CARLA simulator
 2 as shown in figure 3.

3 For vehicle control, we implement a model predictive controller (MPC) for VUT, which
 4 is a widely used control model for autonomous vehicles (28, 29). A detailed modeling of vehicle
 5 controller will be discussed in section 6.3.

6 **Description of background vehicle and vehicle under test**

7 We take the assumption that the VUT observe surrounding vehicle through pixel2seq(30), and the
 8 relative coordinates of BV is obtained along with corresponding depth information. We assume that
 9 the depth information given to VUT is correct, as error in depth information does not contribute to
 10 the error introduced in the system. We then obtain the relative coordinate from the 2d front-person-
 11 view data and convert to relative coordinate with regard to ego vehicle. Given multiple cameras
 12 are mounted on VUT, redundancy reduction is performed to lower the impact of duplicated objects
 13 on the downstream application of prediction and planning. Based on the relative coordinates after
 14 redundancy reduction and the VUT's ego status, the relative coordinates of background vehicles
 15 obtained from perception is then obtained and used as inputs for VUT's prediction, planning and
 16 control.

17 For VUT, the planning and control decision is made based on the observed BV information.
 18 If BV information has delays according to the system time stamp, then the possible BV location
 19 will be calculated by the prediction module within the controller. The prediction module had been
 20 integrated with the controller. The controller formulation of VUT is described below.

21 In this scenario, vehicle n is the background vehicle traveling on the highway main lane
 22 and may interact with VUT thus VUT had to respond to it using perception information. Vehicle
 23 i is the autonomous vehicle under investigation, also the victim vehicle in this work. The control
 24 (i.e., acceleration) of vehicle i is determined by a nonlinear optimal controller described later.

25 The status of the scenario is described by vehicle n, i . As we assume that all vehicles are
 26 traveling at the center of the lane, no lateral movement is considered. Thus we use 1-D location



FIGURE 3: Snapshot of evaluation scenarios used for both perception evaluation and end to end evaluation

x and longitudinal speed v to describe the status of each vehicle. For each vehicle involved in the system, we denote the vehicle status as s_j^k , where k refers to the timestamp and $j \in \{n, m, i\}$, referring to the vehicle index. We then have $s_j^k = (x_j^k, v_j^k)$

We treat the system as a discrete-time system and use a constant speed kinematic model to describe the state transition of the system. This status update function has been used in the prediction module of Autoware.AI (25), which we apply in this paper. Such transition is described below as:

$$z_{k+1} = z_k + F(z_k, u_k) \quad (2)$$

where $z_k = [s_n^k, s_i^k]$ and $F(z_k, u_k) = [\Delta t * v_k, 0]^T$ for state estimate function from observation, note that the state estimation is based on the assumption that vehicle n travels on the highway main lane.

The longitudinal speed planning strategy for the victim vehicle is that, given perception information, it wants to arrive at the conflict point as fast as possible while maintaining a certain level of performance for safety, mobility, and comfort. We set the conflict point as the back of vehicle n with safety minimum gap as 2 meters. In this paper, victim vehicle V_i generates its control command using the optimal control method, which we formulated as solving a finite-time constraint discrete non-linear model predictive control problem. The problem is solved following a rolling horizon manner. Let N be the prediction horizon, and we have the following formulation

1 for the trajectory control problem:

$$\begin{aligned}
 & \min_{u_{1:N}} \sum_{k=1}^N w_1(x_i - x_c)^2 + w_2(v_i - v_i^{ff})^2 \\
 & \quad + w_3 u_i^2 + w_4 \left(\left(\frac{x_i - x_c}{v_i} - \frac{x_n - x_c}{v_n} \right)^2 - h \right)^2 \\
 & \text{s.t. } z_{k+1} = F(z_k, u_k) \\
 & \quad z_k \in \mathbf{Z}, u_k \in \mathbf{U} \\
 & \quad k = 0, \dots, N-1
 \end{aligned} \tag{3}$$

3 By solving this nonlinear model predictive control problem, we have the control input to the au-
 4 tonomous vehicle at each time stamp. $w_1(x_i - x_c)^2$ regulate the distance between the autonomous
 5 vehicle and the conflict point, and we want the autonomous vehicle to arrive at conflict point as
 6 fast as possible. $w_2(v_i - v_i^{ff})^2$ regulate the speed difference between the vehicle's speed and the
 7 free-flow speed of the road. $w_3 u_i^2$ penalizes large control input to improve the comfort performance
 8 for the autonomous vehicle. $w_4 \left(\left(\frac{x_i - x_c}{v_i} - \frac{x_n - x_c}{v_n} \right)^2 - h \right)^2$ enforce the safety constraints based on the
 9 time-to-collision between the ego vehicle and the background vehicle. In this optimization prob-
 10 lem, \mathbf{Z} is the feasible set of all states and \mathbf{U} is the feasible set of all control commands of the victim
 11 vehicle, which considers the acceleration and de-acceleration limits of the actuator.

12 We assume that the victim's state estimation of the surrounding vehicles is based only on
 13 perception, which is common in autonomous vehicle settings.

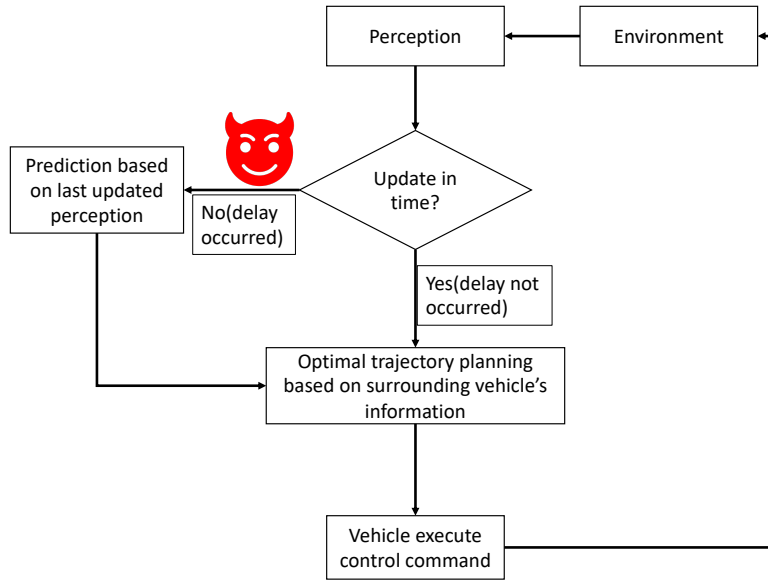


FIGURE 4: Information update and command generation for the victim autonomous vehicle when under or not under attack

14 For response to traffic signal regarding autonomous vehicle, we deploy the evaluation in
 15 signalized intersection in Town04, a virtual map provided by CARLA simulator. In this scenario,
 16 we deploy a traffic signal controller at the T intersection on highway, and we apply a fixed signal
 17 time. For the highway direction, the signal time has 20 seconds green time, and 15 seconds red.

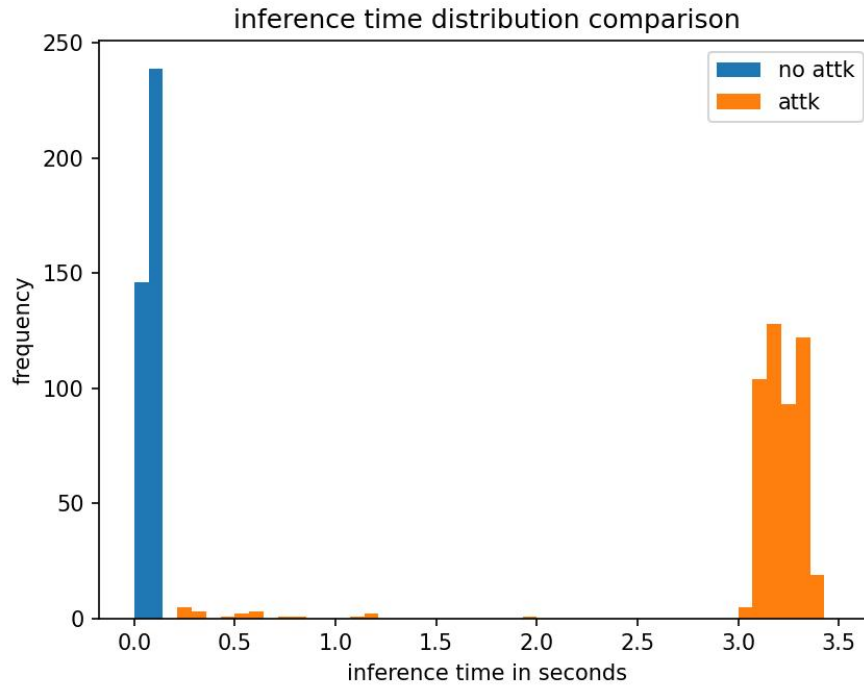


FIGURE 5: Histogram between baseline and pixel2seq under attack

1 RESULT

2 Delay on End-to-end system

3 Given that the impact of an inference time attack on the AV system affects both inference cor-
 4 rectness and latency, we first evaluated the difference in inference time from an end-to-end system
 5 perspective. The reason why we perform the evaluation on inference time only at the en-to-end sys-
 6 tem at this time is that, as a safety-critical cyber-physical system, both response time and inference
 7 result has important impact on the decision making for AV system. What's more, the perception to
 8 surrounding environment with lower frequency can also result in a delayed interaction for AV with
 9 regard to surrounding environment. Even with the help of prediction,

10 Assuming the same initial speed and driving length, a significant difference in inference
 11 time can result in a substantial variation in the number of inferences executed. For the sticker
 12 attack, the mean inference time for pixel2seq without an attack is about 0.05-0.15 seconds, while
 13 the mean inference time for pixel2seq under attack is around 3.2 seconds. If the total driving time
 14 per run is 40 seconds, pixel2seq can perform 400 inferences without an attack, and 12.5 inferences
 15 when under attack. This significant imbalance in the number of observations makes it challenging
 16 to conduct a fair comparison.

17 Therefore, we plotted the accumulated inference time data to demonstrate the effectiveness
 18 of the attack. The plot is shown in Figure 5. By accumulating multiple experiments for inference
 19 time attack with 40 times, we obtained the histogram for perception module when under attack.
 20 As shown in the figure, the inference time under attack is around 3.0-3.5 seconds, which is not
 21 suitable for a time-critical cyber-physical system like AV.

1 Car-following case

2 We conducted evaluations on two different scenarios and two different attack models as mentioned
 3 in section 4. For the car-following case with a sticker attack, we ran 40 runs and achieved a
 4 collision rate of 100%. Moreover, upon examining the experiment log, we found that the inference
 5 process was delayed for 100% of the frames due to adversarial perturbation. We define 'delayed
 6 inference' by comparing the inference time during end-to-end evaluation.

7 We observed that the inference time attack resulted not only in a delayed response but also
 8 in a falsified inference result. Both the delayed response and the falsified result violate the time-
 9 criticality requirement for AV, which is a time-sensitive cyber-physical system. The time length of
 10 each run ranged from 30-40 seconds.

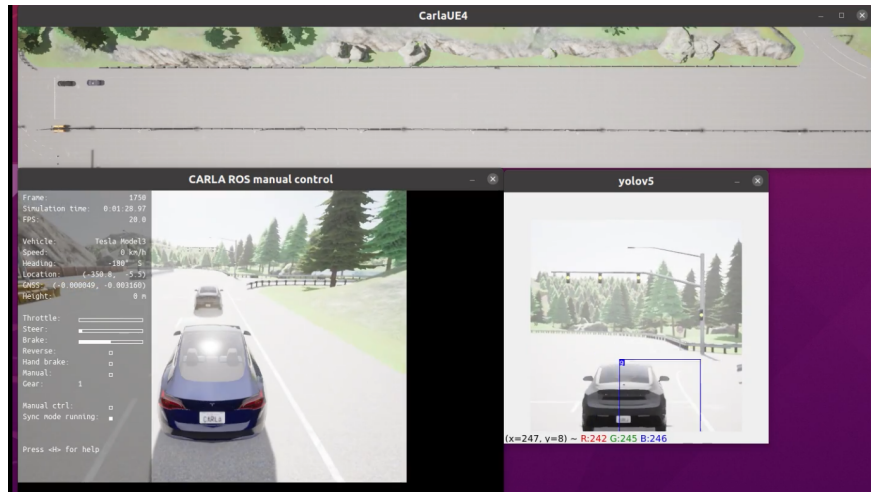


FIGURE 6: System snapshot and CAV perception in benign case

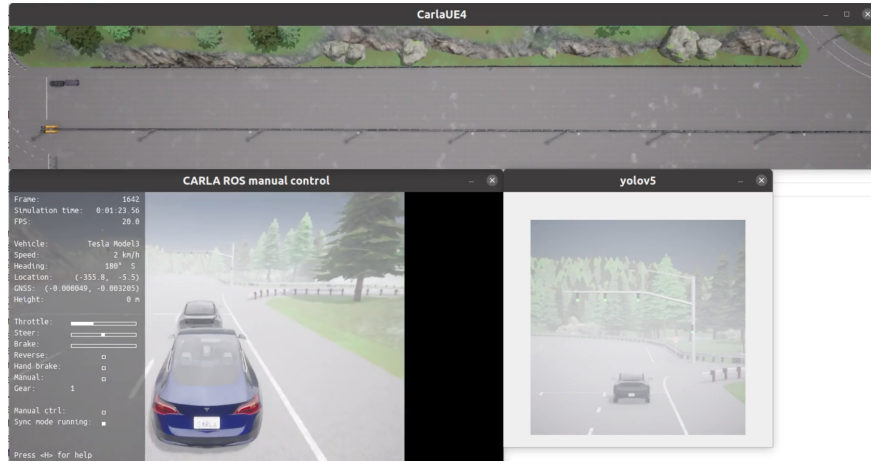


FIGURE 7: System snapshot and CAV perception under attack

11 In contrast, in benign cases, our implemented longitudinal controller always responds
 12 promptly and appropriately to the detected lead vehicle, generating a control command that en-
 13 sures both safety and rider comfort. No collisions were observed in these benign cases.

We present a snapshot of our simulation system, illustrating how the Vehicle Under Test (VUT) responds to a background vehicle (BV) based on the corresponding perception result. Figure 6 showcases the Bird's Eye View (BEV) of the evaluation scenario in the upper part, displaying the relative distance between the VUT and BV. In the lower left, we provide a real-time camera feed from a Front-Person-View (FPV) for the VUT, also indicating the relationship between the VUT and BV. The lower right section presents the inference and perception result from the Pix2seq algorithm. In the benign case, the BEV, FPV, and inference results from Pix2seq are consistent with each other, suggesting that no delay occurred during the evaluation scenario.

Figure 7 has a similar layout for cross-reference and represents an example of Pix2seq under attack. The BEV view reveals that a collision has already occurred, a fact also visible from the FPV that bypasses the inference. However, the lower right section, showing the inference result, still indicates that the BV is far from the VUT. This discrepancy demonstrates a "delayed inference and response," suggesting that the VUT is responding to environmental data from some time ago, thus violating the time-criticality principle intrinsic to safety-critical systems like autonomous vehicles.

For impact analysis on vehicle and the transportation system, we first evaluate a typical case where a background vehicle stay stationary in the road network, and see what is the impact of inference time attack on AV system. The time-space diagram and speed profile of victim vehicle in benign case is shown below:

In Town04, the ground of testing scenario is not even, therefore we let the controller apply a brake instead of setting throttle and brake as zero when planning speed is zero. This caused the speed drop of target vehicle at figure 8b around time 30s. Also the controller takes perception error from vehicle perception data, thus the estimation of the distance between the ego vehicle and front vehicle varies as the perception result itself has some error. The fluctuation of distance estimation from perception result in the planned speed for target vehicle, therefore the speed profile of ego vehicle did not showed a perfect constant speed.

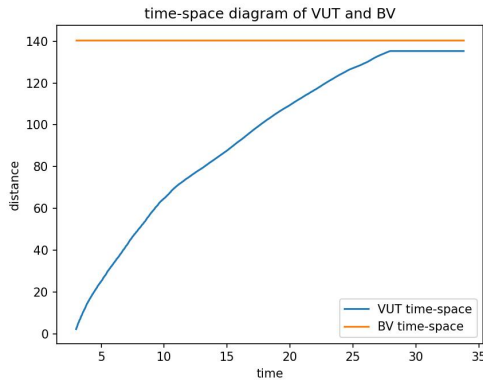
We also launched the inference time attack for the same case where the background vehicle remained in the same position and tried to evaluate the target vehicle's response to the background vehicle. We obtained the time-space diagram of the VUT and the background vehicle. We also obtained the speed profile of the VUT to see the impact of the attack on VUT. Both benign and malicious cases are shown in figure 8.

Traffic Signal Response scenarios

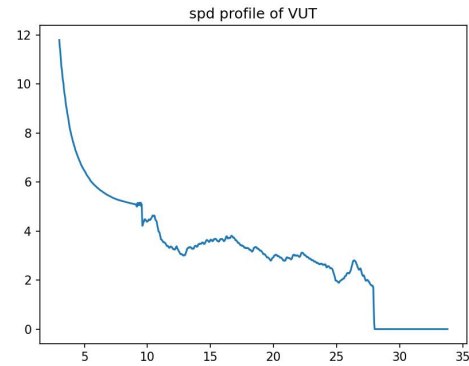
For Traffic Signal Response scenarios, the cases differ slightly from those in the car-following scenario. In traffic signal response cases, delayed perception can result in red-light-running, or delayed start-up.

Red-light running, which violates traffic rules and threatens safety, occurs when the actual traffic signal is red but the perception result for the VUT indicates it's green due to delayed perception. Delayed start-up mainly impact the mobility within the transportation network. This situation is the opposite of red-light running, resulting from a delayed perception when the actual traffic signal is green, but the VUT perceives it as red. It's also possible that delayed perception has no impact on the VUT's response to traffic lights. In such cases, the traffic light remains the same when the VUT passes the intersection.

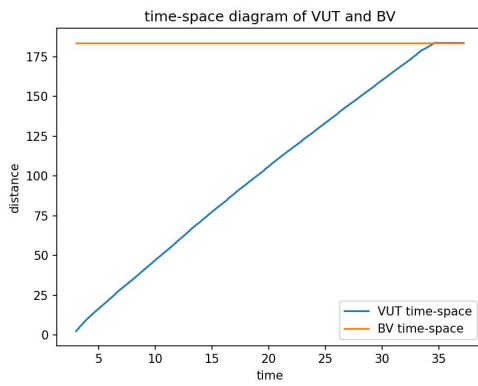
For the sticker attack, we obtain the time-space diagram and speed profile of VUT in both benign and malicious cases. In the benign case, our implemented latitudinal controller could al-



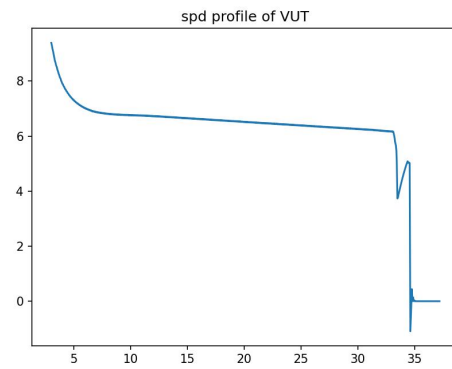
(a) Time-space diagram of both target vehicle and background vehicle in benign case



(b) Speed profile of target vehicle in benign case



(c) Time-space diagram of both target vehicle and background vehicle when target vehicle is under attack



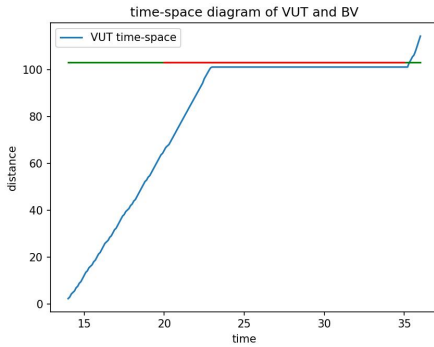
(d) Speed profile of target vehicle when under attack

FIGURE 8: Time-space diagram and speed profile of vehicles in benign case and under attack, when responding to background vehicle

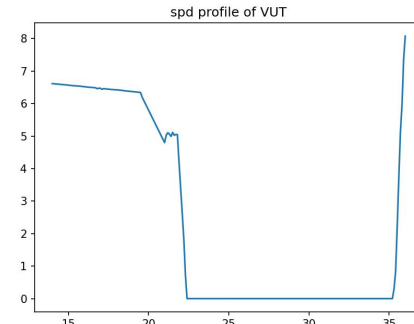
ways respond properly and timely to the traffic signal once it detected the traffic lights. It generated a longitudinal control command that satisfied safety, rider comfort, and mobility requirements, considering the impact on the entire traffic network. We observed no collisions in the benign case and no violation of traffic signal.

For malicious case, we also evaluate the scenario with time-space diagram and speed profile for VUT. As show in figure 9c and 9d, the delayed response of VUT leads to the violation of traffic rule, as no deacceleration is performed in response to the traffic signal in red.

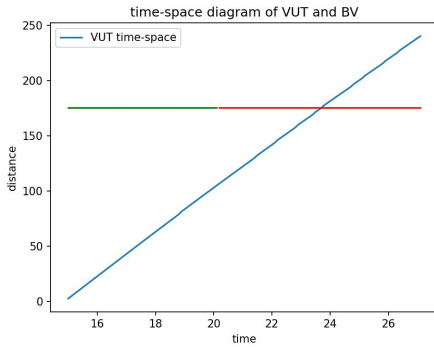
To evaluate the impact of inference time attack on AV's response to the traffic signal, we first obtained a baseline performance of AV respond to TSC system. From the baseline response of AV, we can see that the AV can respond to traffic signal correctly and stop at stop bar when it detects a red signal, and resume driving once the signal turns green. In contrast, when under attack, due to the delayed inference the VUT did not respond to the traffic signal and directly do a



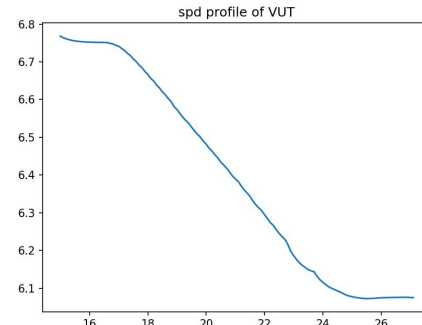
(a) Time-space diagram of both target vehicle and background vehicle in benign case



(b) Speed profile of target vehicle in benign case



(c) Time-space diagram of both target vehicle and background vehicle when target vehicle is under attack



(d) Speed profile of target vehicle when under attack

FIGURE 9: Time-space diagram and speed profile of vehicles, benign case and under attack

1 red-light running. Both cases are shown in figure 9.

2 DISCUSSION

3 In this work, our primary focus is on the inference attack. We accept the attacker's assumption that
 4 once the attack is launched and the VUT is under attack, the inference delay is maximized. Also,
 5 in this work, the main product is the inference time delay, while the falsified inference result is a
 6 side product.

7 Therefore, from a vehicle and transportation perspective, we tend to consider a weaker
 8 assumption, that when an attack is launched, the delay may not reach its maximum. Additionally,
 9 we are interested in understanding if the delay in inference can have an accumulative effect on both
 10 vehicle control and the traffic state.

11 In this context, one possible direction for future work would be to treat the generation of
 12 adversarial samples as a sequence instead of a single adversarial sample. In other words, we'd like
 13 to explore if smartly choosing the inference delay time sequence can result in a greater impact not
 14 only on a single vehicle but also on traffic flow. The generation of such a sequence and connecting

1 it with the generation of adversarial perturbation is one possible avenue for future work.

2 At the vehicle control and transportation system level, one possible defense method is to
3 utilize the last observed perception information that has not been attacked. The current prediction
4 method for vehicle-side control suffers from a relatively short prediction horizon. It's important
5 to note that the common setting for the prediction horizon of the vehicle-side controller is approx-
6 imately 2 seconds. However, during the inference time attack mentioned in section 4, based on
7 the end-to-end evaluation, the inference delay could be as long as 3.2 seconds, which exceeds the
8 common planning horizon.

9 Therefore, utilizing traffic-level prediction that has a lower update interval but longer infor-
10 mation validity could be a possible direction. Integrating traffic-level information into the vehicle-
11 side prediction can better equip the VUT to cope with the surrounding traffic state, and thus miti-
12 gate the attack to some extent.

13 **CONCLUSION**

14 In this work, we revealed a potential threat to the AV system through perception module. We
15 showed that certain kind of attack can delay inference of perception module in AV system and
16 thus threaten the safety for AV. We evaluated the attack on an end-to-end simulation system us-
17 ing standardized middleware structure with real-time system. We performed the attack under two
18 scenarios: response to background vehicle and response to traffic signal. In both cases we show
19 significant difference in performance with and without attack. Future work regarding this vulnera-
20 bility in inference time can relate to traffic-informed defense, and identification of AV that is under
21 such attack.

22 **AUTHOR CONTRIBUTIONS STATEMENT**

23 The authors confirm their contributions to the paper as follows: study conception and design:
24 Chen, Chen, Yang, Feng; system construction: Chen, Chen, Li; data collection: Chen, Chen, Li;
25 analysis and interpretation of results: Chen, Feng; manuscript preparation: all authors. All authors
26 reviewed the results and approved the final version of the manuscript. ChatGPT4 is used to correct
27 the grammar issues within the manuscript.

1 REFERENCES

- 2 1. Smith, S., J. Bellone, S. Bransfield, A. Ingles, G. Noel, E. Reed, M. Yanagisawa, et al.,
3 *Benefits estimation framework for automated vehicle operations*. United States. Depart-
4 ment of Transportation. Intelligent Transportation Systems Joint Program Office, 2015.
- 5 2. Laberge, J., C. Scialfa, C. White, and J. Caird, Effects of passenger and cellular phone
6 conversations on driver distraction. *Transportation Research Record*, Vol. 1899, No. 1,
7 2004, pp. 109–116.
- 8 3. Caird, J. K., S. Chisholm, C. J. Edwards, and J. I. Creaser, The effect of yellow light
9 onset time on older and younger drivers' perception response time (PRT) and intersection
10 behavior. *Transportation research part F: traffic psychology and behaviour*, Vol. 10, No. 5,
11 2007, pp. 383–396.
- 12 4. Yadav, A. K. and N. R. Velaga, Modelling the relationship between different Blood Alco-
13 hol Concentrations and reaction time of young and mature drivers. *Transportation research*
14 *part F: traffic psychology and behaviour*, Vol. 64, 2019, pp. 227–245.
- 15 5. Beregi, S., S. S. Avedisov, C. R. He, D. Takacs, and G. Orosz, Connectivity-based delay-
16 tolerant control of automated vehicles: theory and experiments. *IEEE Transactions on*
17 *Intelligent Vehicles*, 2021, pp. 1–1.
- 18 6. Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair,
19 A. Courville, and Y. Bengio, Generative adversarial networks. *Communications of the*
20 *ACM*, Vol. 63, No. 11, 2020, pp. 139–144.
- 21 7. Goodfellow, I. J., J. Shlens, and C. Szegedy, Explaining and harnessing adversarial exam-
22 ples. *arXiv preprint arXiv:1412.6572*, 2014.
- 23 8. Szegedy, C., W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus,
24 Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.
- 25 9. Patel, N., P. Krishnamurthy, S. Garg, and F. Khorrami, Adaptive Adversarial Videos on
26 Roadside Billboards: Dynamically Modifying Trajectories of Autonomous Vehicles. In
27 *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019,
28 pp. 5916–5921.
- 29 10. Zhou, H., W. Li, Z. Kong, J. Guo, Y. Zhang, B. Yu, L. Zhang, and C. Liu, DeepBillboard:
30 Systematic Physical-World Testing of Autonomous Driving Systems. In *2020 IEEE/ACM*
31 *42nd International Conference on Software Engineering (ICSE)*, 2020, pp. 347–358.
- 32 11. Eykholt, K., I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno,
33 and D. Song, Robust physical-world attacks on deep learning visual classification. In *Pro-*
34 *ceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp.
35 1625–1634.
- 36 12. Chen, S., Z. Song, M. Haque, C. Liu, and W. Yang, NICGSlowDown: Evaluating the
37 Efficiency Robustness of Neural Image Caption Generation Models. In *Proceedings of the*
38 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15365–
39 15374.
- 40 13. Haque, M., A. Chauhan, C. Liu, and W. Yang, Ilfo: Adversarial attack on adaptive neural
41 networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*
42 *Recognition*, 2020, pp. 14264–14273.
- 43 14. Hong, S., Y. Kaya, I.-V. Modoranu, and T. Dumitraş, A Panda? No, It's a Sloth:
44 Slowdown Attacks on Adaptive Multi-Exit Neural Network Inference. *arXiv preprint*
45 *arXiv:2010.02432*, 2020.

- 1 15. Chen, S., H. Chen, M. Haque, C. Liu, and W. Yang, *SlothBomb: Efficiency Poisoning*
2 *Attack against Dynamic Neural Networks*, 2023.
- 3 16. Wang, P., X. Wu, and X. He, Modeling and analyzing cyberattack effects on connected au-
4 to-
5 mated vehicular platoons. *Transportation research part C: emerging technologies*, Vol. 115, 2020, p. 102625.
- 6 17. Feng, D., A. Harakeh, S. L. Waslander, and K. Dietmayer, A review and comparative study
7 on probabilistic object detection in autonomous driving. *IEEE Transactions on Intelligent*
8 *Transportation Systems*, 2021.
- 9 18. Bezzina, D. and J. Sayer, Safety pilot model deployment: Test conductor team report.
10 *Report No. DOT HS*, Vol. 812, No. 171, 2014, p. 18.
- 11 19. Liu, H., Y. Wu, Z. Yu, Y. Vorobeychik, and N. Zhang, SlowLiDAR: Increasing the Latency
12 of LiDAR-Based Detection Using Adversarial Examples. In *Proceedings of the IEEE/CVF*
13 *Conference on Computer Vision and Pattern Recognition*, 2023, pp. 5146–5155.
- 14 20. Chen, S., H. Chen, M. Haque, C. Liu, and W. Yang, The Dark Side of Dynamic Rout-
15 ing Neural Networks: Towards Efficiency Backdoor Injection. In *Proceedings of the*
16 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 24585–
17 24594.
- 18 21. Cao, Y., N. Wang, C. Xiao, D. Yang, J. Fang, R. Yang, Q. A. Chen, M. Liu, and B. Li,
19 Invisible for both camera and lidar: Security of multi-sensor fusion based perception in
20 autonomous driving under physical-world attacks. In *2021 IEEE Symposium on Security*
21 *and Privacy (SP)*, IEEE, 2021, pp. 176–194.
- 22 22. Song, D., K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, F. Tramer, A. Prakash,
23 and T. Kohno, Physical adversarial examples for object detectors. In *12th USENIX work-*
24 *shop on offensive technologies (WOOT 18)*, 2018.
- 25 23. Komkov, S. and A. Petiushko, Advhat: Real-world adversarial attack on arcface face id
26 system. In *2020 25th International Conference on Pattern Recognition (ICPR)*, IEEE,
27 2021, pp. 819–826.
- 28 24. Lopez, P. A., M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich,
29 L. Lücken, J. Rummel, P. Wagner, and E. Wiessner, Microscopic Traffic Simulation us-
30 ing SUMO. In *2018 21st International Conference on Intelligent Transportation Systems*
31 *(ITSC)*, 2018, pp. 2575–2582.
- 32 25. *Autoware.AI*, 2022.
- 33 26. *Baidu Apollo team (2023), Apollo: Open Source Autonomous Driving*. <https://github.com/ApolloAuto/apollo>, 2023, accessed: 2023-08-01.
- 34 27. Treiber, M., A. Hennecke, and D. Helbing, Congested traffic states in empirical observa-
35 tions and microscopic simulations. *Physical review E*, Vol. 62, No. 2, 2000, p. 1805.
- 36 28. Cavanini, L., P. Majecki, M. J. Grimble, V. Ivanovic, and H. E. Tseng, LPV-MPC Path
37 Planning for Autonomous Vehicles in Road Junction Scenarios. In *2021 IEEE Interna-*
38 *tional Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 386–393.
- 39 29. Liu, C., S. Lee, S. Varnhagen, and H. E. Tseng, Path planning for autonomous vehicles
40 using model predictive control. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, IEEE,
41 2017, pp. 174–179.
- 42 30. Chen, T., S. Saxena, L. Li, D. J. Fleet, and G. Hinton, Pix2seq: A Language Modeling
43 Framework for Object Detection. In *International Conference on Learning Representa-*
44 *tions*, 2021.
- 45