Does Physical Adversarial Example Really Matter to Autonomous Driving? Towards System-Level Effect of Adversarial Object Evasion Attack

Ningfei Wang Yunpeng Luo Takami Sato Kaidi Xu[†] Qi Alfred Chen
University of California, Irvine {ningfei.wang, yunpel3, takamis, alfchen}@uci.edu
[†]Drexel University kx46@drexel.edu

Abstract

In autonomous driving (AD), accurate perception is indispensable to achieving safe and secure driving. Due to its safety-criticality, the security of AD perception has been widely studied. Among different attacks on AD perception, the physical adversarial object evasion attacks are especially severe. However, we find that all existing literature only evaluates their attack effect at the targeted AI component level but not at the system level, i.e., with the entire system semantics and context such as the full AD pipeline. Thereby, this raises a critical research question: can these existing researches effectively achieve system-level attack effects (e.g., traffic rule violations) in the real-world AD context? In this work, we conduct the first measurement study on whether and how effectively the existing designs can lead to system-level effects, especially for the STOP sign-evasion attacks due to their popularity and severity. Our evaluation results show that all the representative prior works cannot achieve any system-level effects. We observe two design limitations in the prior works: 1) physical model-inconsistent object size distribution in pixel sampling and 2) lack of vehicle plant model and AD system model consideration. Then, we propose SysAdv, a novel system-driven attack design in the AD context and our evaluation results show that the system-level effects can be significantly improved, i.e., the violation rate increases by around 70%.

1. Introduction

Autonomous Driving (AD) vehicles are now a reality in our daily life, where a wide variety of commercial and private AD vehicles are driving on the road. For instance, the millions of Tesla cars [30] equipped with Autopilot [54] are publicly available. To ensure safe and correct driving, a fundamental pillar is *perception*, which is designed to detect surrounding objects in real time. Due to the safety- and security-criticality of AD perception, various prior works have studied its security, especially the ones that aim at causing the evasion of critical physical road objects (e.g., STOP signs and pedestrians), or *physical adversarial object evasion attack* [7, 8, 12, 16, 27, 32, 62, 66, 70].

Although these attacks are all motivated by causing erroneous driving behaviors at the AD system level (e.g., vehicle collisions and traffic rule violations), we find that so far they predominately only evaluate the attack success *at the targeted AI component level alone* (e.g., judged by per-frame object misdetection rates [12, 16, 27, 66, 70]), without further evaluation *at the system level*. Specifically, to systematically perform such system-level evaluation, we need to measure the end-to-end system-level attack success metrics (e.g., collision rates) with the full system-level attack context enclosing the attack-targeted AI component, for example, the remaining AD system pipeline such as object tracking, planning, and control, closed-loop control, and the attack-targeted driving scenario. In this paper, we call such system-level attack context *system model* for such adversarial attacks (§2). This thus raises a critical research question: *can these existing works on physical adversarial object evasion attacks effectively achieve the desired system-level attack effects in the realistic AD system settings?*

To systematically answer this critical research question, we conduct the first measurement study on representative prior object-evasion attacks with regard to their capabilities in causing system-level effects (§3). We propose a general framework, i.e., a system model, including perception modeling from the physical world, to measure STOP sign-evasion attack which is our target due to its high representativeness [53] and its direct impacts on driving correctness and road safety. Our results show that *all* the representative existing works cannot cause any STOP sign traffic rule violation within the system model including a representative closed-loop control AD system in the common speed range for STOP sign-controlled roads in the real world even though the most effective attack can achieve more than 70% average attack success rate at the AI component alone.

We further investigate the root causes and find that all the existing works have design limitations on achieving effective system-level effects due to the lack of a system model in AD context for attack design: 1) physical model-

inconsistent object size distribution in pixel sampling and 2) lack of vehicle plant model and AD system model consideration (detailed in §4). We further propose SysAdv, a system-driven attack design, which can be integrated with all state-of-the-art attack methods to significantly improve system-level effects by overcoming the two limitations.

We evaluate our novel proposed attack design in our platform and show that the system-level effect can be significantly improved in §5, i.e., the system violation rate can be increased by around 70%. To further validate the generality of our attack, we also examine generality on different AD system parameters (§5.2) and different object types (§5.3), which shows improvement at both component- and system-level. Demo videos are at the project website: **https://sites.google.com/view/cav-sec/sysadv**.

To sum up, this paper makes the following contributions:

- We conduct the first measurement study on the system-level effect of the representative prior object-evasion attacks with our proposed novel evaluation framework (i.e., system model) including 4 popular object detectors and 3 state-of-the-art object-evasion attacks.
- We identify the limitations of prior works which hinder them in potently achieving system-level effects and propose SysAdv, a system-driven adversarial object-evasion attack with the system model in AD context.
- We further evaluate SysAdv and show that the system-level effect of SysAdv can be significantly improved, i.e., the system violation rate increases by around 70%.

2. Related Work and Background

Camera-based AD perception. Camera-based AD perception generally leverages DNN-based object detection to detect or recognize road objects of various categories (e.g., traffic signs, vehicles, and pedestrians) in consecutive image frames [10]. State-of-the-art DNN-based object detectors can be classified into two categories: one-stage object detector, and two-stage object detector [74]. The former, such as YOLO [29, 45, 46], usually has higher detection speed, while the latter, such as Faster R-CNN [47], usually has higher detection accuracy. In this paper, we focus on the security aspects of camera-based AD perception and perform the corresponding experiments on both object detector categories. We perform the measurement study of physical adversarial object evasion attack in AD perception §3 including these two kinds of object detectors.

Physical adversarial object evasion attacks in AD context. Recent works find that DNN models are generally vulnerable to adversarial attacks [9, 18, 37, 38, 65, 69]. Due to the direct reliance of camera-based AD perception on DNN object detectors, various prior works have explored the feasibility of adversarial attacks in AD context [14,27,36,37,50,51,53,59,60,66,70,73]. Among them, *physical adversarial object evasion attacks*, which typically use physical-world attack vectors such as malicious patches to cause the disappearance of road objects (e.g., pedestrians and traffic signs) [12, 16, 27, 62, 66, 70], are especially severe due to their direct impacts on driving correctness and road safety. However, as detailed in later sections, we find that so far the considerations and integration of the corresponding *system models* (detailed below) in the prior works are *far from enough* in both attack designs and evaluation, which substantially jeopardizes the meaningfulness of their designs from the end-to-end AD driving perspective (§3).

Gap between AI component errors and their system-level effect. We do not intend to claim to be the first to point out, analyze, measure, or optimize the gap between AI component errors and their system-level effect in general; there exists a large body of prior works in various other problem contexts (e.g., computer vision system [24,26,44], image analysis [22, 68], camera surveillance [20, 21], video analytics [19, 55], planning [42, 43, 56], and control [56]) across academia and industry that have studied the characterization and/or optimization of end-to-end system performance [5, 17] with regard to AI/vision component errors. Nevertheless, to the best of our knowledge, none of them 1) quantified such gaps in the context of adversarial attacks on autonomous systems, especially those in real-world system setups; and 2) identified novel designs that can systematically address or fill such gaps on autonomous systems, which we believe are our novel and unique contributions.

Systems model for AD AI adversarial attacks. To understand the end-to-end system-level impacts of an adversarial attack against a targeted AI component in an AD system (e.g., whether it can indeed effectively cause undesired AD system-level property violations), we need to systematically consider and integrate the overall system semantics and context that enclose such AI component into the security analysis [15, 52]. In this paper, we call a systematic abstraction of such system semantics and context the *system model* of such AD AI adversarial attacks. Specifically, in the AD context we identify 3 essential sub-components in such system model: 1) *the AD system model*, i.e., the full-stack AD system pipeline that encloses the attack-targeted AI components and closed-loop control, e.g., the object tracking, planning, and control pipeline for the object detection AI component; 2) *the vehicle plant model* [15, 40], which defines the physical properties of the underlying vehicle system under control, e.g., maximum/minimum acceleration/deceleration, steering rates, sensor mounting positions, etc.; and 3) *the attack-targeted operation scenario model*, which defines the physical driving environment setup, driving norms (e.g., traffic rules), and the system-level attack goal (e.g., vehicle collision, traffic rule violation, etc.) targeted by the AD AI adversarial attack.

System model for adversarial object-evasion attacks.

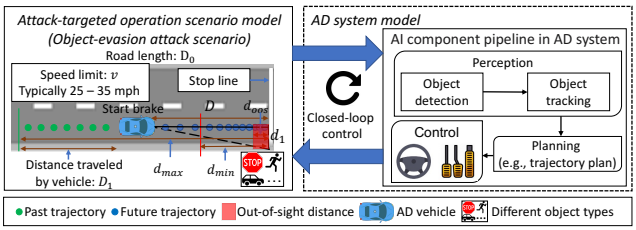


Figure 1: Illustration of the system model for adversarial object-evasion attacks in AD context.

Fig. 1 illustrates the aforementioned system model defined for the adversarial object-evasion attack. The AD system model for object detection, the targeted AI component in adversarial object-evasion attacks, mainly includes its downstream tasks of object tracking, planning, and control, and closed-loop control. The vehicle plant model mainly includes the physical properties related to longitudinal control, e.g., the minimum brake distance (d_{min}), and the distance to the stop line (stop to avoid violating traffic rules or crashes) where the stop line is out of sight in the camera image d_{oos} (depending on the hood length and the camera mounting position). The operation scenario model includes the speed limit, lane width, the relative positioning and facing of the object to the ego lane, the driving norm that the vehicle typically drives at constant speed before it starts to see the object (d_{max}), and the system-level attack goal that triggers the traffic rule violation (i.e., hit into the object or exceeding the stop line). We will use this system model in our studies in the following sections. There exists several example attacks for the system model such as STOP sign-evasion attack, which is the most extensively-studied physical adversarial object evasion attack in AD context [53], and thus will be the main focus of our study in later sections; pedestrian-evasion attack [66]; car-evasion attack [58]; etc.

3. System-Level Effect of Prior Works

Scientific gap in existing works: Lack of system-level evaluation. Despite a plethora of published attack works on physical adversarial object evasion attacks in AD context (§2), we find that actually *all of them* only evaluate their attack effect *at the targeted AI component level* (i.e., judged by per-frame object misdetection rates [16, 27, 66, 70]), without any evaluation *at the system level*, i.e., with the corresponding system models for such attacks as described in §2. However, in the Cyber-Physical System (CPS) area, it is widely recognized that in AD system, AI component-level errors do not necessarily lead to system-level effects (e.g., vehicle collisions) [15, 28, 52]. Thus, without system-level evaluation, it can be highly difficult to scientifically know whether the attack is actually meaningful from the end-to-end AD driving perspective. We view this as a critical scientific gap in this current research space, and to address this, we perform a measurement study on the existing

(a) Benign (b) RP_2-Y2 (c) SIB-Y3 (d) SIB-FR (e) FTE-Y3 (f) FTE-Y5

Figure 2: Visualisation of STOP signs attack reproduction (in Table 1) for measurement study in physical world.

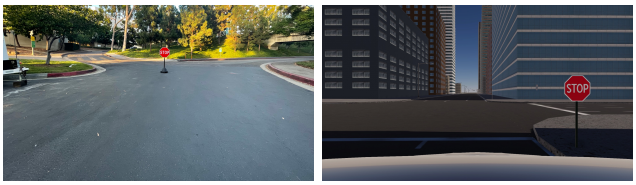
works about their system-level effects. We choose to focus on *adversarial STOP sign-evasion attck* as our target due to its high representativeness in this research space and also its direct impacts on driving correctness and road safety (§2).

3.1. Attack Formulation and Selection

Attack formulation. We assume that the attacker can arbitrarily manipulate pixels within restricted regions known as adversarial patch attack [4, 16, 70]. Such a patch attack is easy to deploy in the real-world and very stealthy. We consider the patch stays on the STOP sign shown in Fig. 2.

Selection of prior STOP sign attack works and their reproduction. There are various prior works on physical adversarial STOP sign-evasion attacks [12, 16, 27, 33, 34, 67, 70]. To perform our system-level effect measurement, we select the most effective ones at AI component level as representative examples. Four model designs (including one-stage and two-stage object detectors in §2) have been covered. For each model, we select the most effective attack design published so far which are shown in Table 1. However, all the STOP sign attacks in Table 1 do not provide the source code. Since we tried to contact the authors of the attacks for the source code but they all cannot provide it, we try our best to reproduce some of the works. Currently, we only have the reproduction for RP_2 and FTE. For SIB, we directly use the STOP sign images shared by the authors of that paper used for their physical-world experiments. We print the high-resolution STOP signs on multiple ledger-size papers and concatenate them together to form full-size real STOP signs which are shown in Fig. 2.

To demonstrate the reproduction correctness, we utilize their original evaluation setups for our trials. Our results are generally similar to theirs confirming the correctness of reproduction. For instance, the original RP_2 paper [16] reports an attack success rate of approximately 63.5% from 0 to 30 feet. With the same setup (outdoor), our results provide a 61.0% attack success rate — nearly mirroring the original. Note that SIB attack on the FR in Table 3 seems anomalous: it records around 47% attack success rate only from 40 to 45 meters, while consistently registering 0% in others. Despite the patch being provided by the authors, the pre-trained FR can be different, where we use MMDetection [11], a PyTorch-based object detection toolbox. Given such potential low transferability, the attack may be less effective compared to their original results. However, this is our best effort to reproduce their results faithfully.



(a) Physical-world scene (b) Simulation scene

Figure 3: Experiment scenes. (a) Real-world scene with real road and injected STOP sign; (b) SVL simulation scene with the San Francisco map in a sunny day at noon.

3.2. Measurement Methodology and Setup

To measure system-level effects, we adopt a simulation-centric evaluation methodology, which has been widely adopted both in academia [49, 57] and in industry [25, 61] due to the inherent limitations of real-road AD testing in cost, safety, efficiency, and corner-case coverage. In this study, we use SVL, a production-grade high-fidelity AD simulator designed for AD systems [48]. As repeatedly demonstrated in various prior works, the end-to-end AD system-level evaluation results in SVL can highly correlate with the same setup tested in the physical world [49,57]. To ensure the fidelity of our evaluation results, we improve the fidelity of the rendering process by modeling the perception results in the real world with a practical setup (details below). Note that the attacks themselves are agnostic to map and time by design, and thus are not generally affected by their changes. In SVL, we use San Francisco map on a sunny day at noon, which is the most representative setup.

Perception results modeling from physical world. To enhance the perception fidelity of simulators, we model the perception results using a practical setup in the real world. This approach represents our best effort to improve the fidelity of the simulation due to the experimental feasibility. Previous studies collect video frames by directly moving towards the STOP sign and simulate varying view angles by rotating the STOP sign itself. This approach is not practical since the vehicles do not directly drive towards the STOP sign, and the STOP sign should instead be located on the roadside as shown in Fig. 1. To improve such unrealistic setups, we follow the system model defined in §2. We recorded several pieces of video along the driving direction D using an iPhone 12 Pro Max starting from 45 m to 4 m (4 m is the d_{oos} in §2). We choose 45 m since 1) it is the minimal brake distance for speed above 50 mph, which exceeds the usual maximum speed of STOP sign areas, and 2) it is already much larger than the maximum distance evaluated in all the prior STOP sign-evasion attack works. We separate the whole range into 9 pieces, each spanning 5 m except the one near the STOP sign, which is 1 m long. Then, we record a video in each region and feed the video into the object detectors to model the perception results. We perform these experiments on sunny days as shown in Fig. 3. With that, we perform perception results injection at the output of the object detection task in the AD system, i.e., first read the ground-truth STOP sign detection results from the simulator and then drop/keep the detection results based on detection rate. For instance, if the attack success rate is 60% for a range, we will randomly drop the STOP sign detection results with a possibility of 60% in that range.

Table 1: Selection of the representative prior works. Specifically, for each of the 4 model types targeted by prior works, we select the most effective attack design published so far.

Model	YOLO v5 (Y5)	YOLO v3 (Y3)	YOLO v2 (Y2)	Faster RCNN (FR)
Attack	FTE [27]	SIB [70]	RP_2 [16]	SIB [70]

Table 2: System-level violation rate in the simulation-based testing and component-level overall ASR for model Y2, Y3, Y5, and FR in benign and attacked scenarios. 10 runs for each cell with different initial AD position. B: benign; Sys: system; Comp: component; ASR: attack success rate.

Eval. level	Speed (mph)	Y2		Y3			Y5		FR	
		B	RP_2	B	SIB	FTE	B	FTE	B	SIB
Sys (violation)	25, 30, 35	0%	0%	0%	0%	0%	0%	0%	0%	0%
Comp (ASR)	Overall	-	71.2%	-	53.1%	53.3%	-	41.0%	-	5.2%

Evaluated AD system pipeline. The AD system pipeline includes representative downstream tasks after object detection, which contains 1) a tracking step using a general Kalman Filter based multi-object tracker [35]; 2) a planning step using a lane-following planner from Baidu Apollo [1], an industry-grade full-stack AD system; and 3) a control step using classic controllers, i.e., PID for longitudinal control used in OpenPilot [41], a production-grade Level-2 AD system, and Stanley [23] for lateral control.

Speed selection. The driving speed is from 25 to 35 mph, with a step size of 5 mph, which is the most common speed range for STOP sign-controlled roads in the real world. 25 mph [3] is the common speed limit for the STOP sign-controlled road intersections, which is more likely to avoid a crash, and 35 mph [6] is the most common speed limit for city streets, which STOP signs are designed for.

3.3. Measurement Results

We evaluate the targeted AD system-level attack effect, i.e., STOP sign violation rate, by defining the STOP sign violation rate as $\frac{N_{\text{violation}}}{N_{\text{total}}}$, in which $N_{\text{violation}}$ means the number of runs where the AD vehicle exceeds the stop line and N_{total} is the number of total runs. Table 2 shows the results where each speed has 10 runs with random initialization of the AD vehicle position while the perception results modeling from real world is shown in Table 3. To our surprise, *none* of the existing representative attacks can trigger STOP

Table 3: Detection rates of different objectors in benign, RP_2-, SIB-, and FTE-attacked scenarios tested in the physical world for perception results modeling (shown in §3.2). Each detection rate below is calculated with at least 400 video frames.

Object Detector \ Distance range (m)		4 - 5	5 - 10	10 - 15	15 - 20	20 - 25	25 - 30	30 - 35	35 - 40	40 - 45
YOLO v2 (Y2)	Benign	100%	100%	71.3%	31.3%	0%	0%	0%	0%	0%
	RP_2 [16]	58.2%	90.0%	76.2%	34.6%	0.1%	0%	0%	0%	0%
YOLO v3 (Y3)	Benign	100%	100%	100%	100%	80.1%	11.8%	6.7%	1.0%	0%
	SIB [70]	93.7%	100%	100%	90.4%	38.2%	0%	0%	0%	0%
	FTE [27]	89.9%	100%	100%	87.3%	42.9%	0.6%	0%	0%	0%
YOLO v5 (Y5)	Benign	100%	100%	100%	100%	98.7%	89.4%	52.3%	25.3%	0%
	FTE [27]	91.2%	100%	100%	99.7%	88.2%	48.4%	3.9%	0%	0%
Faster-RCNN (FR)	Benign	100%	100%	100%	100%	100%	100%	100%	100%	100%
	SIB [70]	100%	100%	100%	100%	100%	100%	100%	100%	53.2%

sign violations in *any* of the common speeds for STOP sign-controlled roads when the benign performs well, though most of the attacks are effective in the component (i.e., with about 45% average attack success rate across the 5 attacks). After inspecting the details, we find that the STOP sign is always tracked at the object tracking step before reaching the minimum brake distance of the AD vehicle due to the low attack success rate in such regions. Taking SIB attack on Y3 as an example, the brake distance for 30 mph is around 15 m. In the benign scenario, the detection rate for 15-20 m is 100%, while the SIB attack can still have 90.4% detection rate as shown in Table 3, which is not enough to make the tracking vanish before the minimum braking distance.

4. System-Driven Attack Design

After realizing that existing works cannot provide any system-level violation in AD context, we propose SysAdv, a system-driven attack design, which can be integrated with all the existing attacks to improve system-level effects.

4.1. System-Driven Attack Design Framework

For the attack design in the prior works [12, 16, 27, 70], we can abstract the key part for attack generation:

$$\arg\min_{p_a} \ \mathbb{E}_{s\sim\mathcal{S}}[\mathcal{L}(M(p_a, O, s, B), \gamma)] \quad (1)$$

$\mathcal{S}$ is the distribution to sample different object sizes in pixels, which is a very important factor in achieving the robust attack at different distances between the AD vehicle and the object. The $\mathcal{L}$ is the loss function used in the prior attacks to achieve high attack effectiveness, p_a is the adversarial patch, O is the object, and function $M(p_a, O, s, B)$ indicates applying p_a to O, then resizing object size in pixel to s, and applying O into the background B, γ means other inputs for loss function in the prior works (e.g., the bounding box information and threshold) related to the object detector alone. After investigation on Eq. (1), we find out that the system model can be involved into two parts, i.e., $\mathcal{S}$ and function $M(.)$, which do not rely on object detector alone.

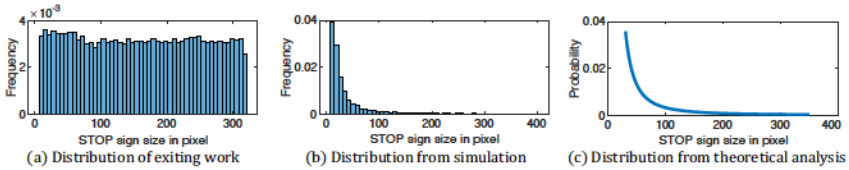
After exploring all the prior works, we discover that all of them do not consider such a system model information into their attack designs, which hinder them to achieve potent system-level effects in AD context. Thus, we propose two novel system-driven designs to significantly improve the system-level effects. Specifically, we involve the system model information into $\mathcal{S}$ and function $M(.)$ from Eq. (1).

4.2. Physical Model-Inconsistent Object Size Distribution in Pixel Sampling

In the prior works [16, 27, 70], to make the attack robust to different distance, Expectation over Transformation (EoT) [2] is used to *uniformly* sample the object size ($\mathcal{S}$ in Eq. (1)) in a certain range [2, 12, 27]. However, with system model (§2), we find that this assumption is not held, which leads to the first observation: *physical model-inconsistent object size distribution in pixel sampling.* To justify the observation, we perform the experimental and theoretical analysis with STOP sign system model as an example.

Experimental analysis. With the same setup in §3, we simulate the real driving scenario in SVL. The STOP sign size in pixels and the distance between the vehicle and the STOP sign can be directly obtained from SVL (§3.2). With that, we can get the frequency distribution histogram over different STOP sign sizes in pixels as shown in the Fig. 4 (b), in which the AD vehicle runs for 30 rounds at speed 25 mph. The distribution shown in Fig. 4 (b) is not uniform, which is wrongly assumed by the prior works [12, 27]. To compare, we sample the STOP sign size from the most recent prior work [27], which designs an algorithm to determine the STOP sign size in a *uniform* way. We run that algorithm 30k times and collect the STOP sign size shown in Fig. 4 (a). The difference between Fig. 4 (a) and (b) indicates that our observation is held experimentally.

Theoretical analysis. Assuming a uniform motion for AD vehicles, we leverage the camera pin-hole model (Fig. 5) for the theoretical analysis. From Fig. 5, we abstract the relationship of real object size (L), real distance(D), focal length(f), and object size in pixel(s) with similar tri-



(a) Distribution of exiting work (b) Distribution from simulation (c) Distribution from theoretical analysis

Figure 4: Different STOP sign size distribution: (a) state-of-the-art existing work [27], (b) our experimental analysis, and (c) our theoretical analysis.

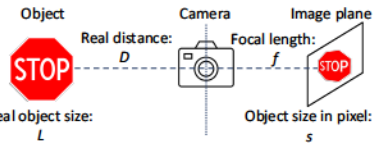


Figure 5: Theoretical analysis of §4.2, i.e., the camera pin-hole model.

angles: $\frac{L}{D} = \frac{s}{f}$. With the system model shown in Fig. 1, we assume that the initial vehicle to STOP sign distance is the road length D_0 and the current vehicle to STOP sign distance is D. Due to uniform motion, the vehicle traveled distance can be formulated as $D_1 = v * t$, where v is the vehicle speed (usually it is the speed limit) and t is the time. To build the relationship between s and the sampled frequency (i.e., the frame number) F, we formulated the time t as $t = \frac{F}{\eta}$, where the η is the image capturing frequency from the camera. Due to $D_1 + D = D_0$ and the camera pin-hole model (Fig. 5), we can obtain the following equation:

$$D_0 = D + v * \frac{F}{\eta} = \frac{L * f}{s} + v * \frac{F}{\eta} \rightarrow F = (D_0 - \frac{L * f}{s}) * \frac{\eta}{v} \quad (2)$$

Eq. (2) is the CDF of s, since the F is accumulated frames. To obtain PDF, CDF's derivative is calculated:

$$F' = \frac{dF}{ds} = \frac{\eta * L * f}{v * s^2} \quad (3)$$

From Eq. (3), the probability distribution is definitely not uniform. We also plot Eq. (3) as shown in Fig. 4 (c) with $\eta = 20$, $L = 1.5$, $v = 25mph$, and $f = 25mm$ (commonly used in AD system such as Baidu Apollo). The distribution is similar to the distribution in the experimental analysis shown in Fig. 4 (b), which supports our observation.

Our system-driven solution (S1). With that, we propose our system-driven solution (S1) to address this inconsistency above. Leveraging the system model in Fig. 1, we define a novel object size distribution based on Eq. (3): $\mathcal{S} = \{s_1, s_2, ..., s_N\}$ as a discrete distribution, where s_i is the object size in pixels. Based on the Eq. (3), the probability of s_i can be abstract as $p(s_i) = \frac{1}{s_i^2} / \sum_{k=1}^{N} \frac{1}{s_k^2}$. Such new object size distribution can be used to address the inconsistency observation and easily integrated into the attack design (Eq. (1)). However, to get the detailed distribution, we have to know the range of $\mathcal{S}$, which will be addressed in the following system-driven solution (S2) in §4.3.

4.3. Lack of Vehicle Plant Model and AD System Model Consideration

In the EoT process, uniformly sampling the object size ($\mathcal{S}$ in Eq. (1)) in a *range* is generally used. In the prior works, they just treat it as hyper-parameters without any reasons [12, 27]. In practice, not every range is equivalently important to achieve system-level effects. Taking STOP sign case as an example, within d_{min} in Fig 1, despite applying maximum deceleration, AD vehicle still cannot fully stop before the stop line. Thereby, such a range is not important to achieve system-level effects. However, none of the prior works involve the system-critical range related to *the vehicle plant model* and *AD system model* in their attack designs, which leads to the second observation: *lack of vehicle plant model and AD system model consideration.*

Table 4: Attack success rate of RP_2 for Y2 evaluated in simulation with both small and large STOP sign pixel sizes.

		Distance (m)							
		4 - 5	5 - 10	10 - 15	15 - 20	20 - 25	25 - 30	30 - 35	Ave
Comp. ASR	Small	6.7%	37.1%	68.3%	81.1%	100%	100%	100%	70.5%
	Large	98.6%	6.1%	0%	1.0%	58.5%	99.1%	100%	51.9%

Note that previous studies which indiscriminately utilize a broad range of object sizes for attacks have exhibited reduced effectiveness in comparison to those employing a small size range. For example, when the object size is small (implying the AD vehicle is far away from the object), attack convergence becomes challenging [27], which indicates that it is harder to attack. Therefore, generally, utilizing a more optimally defined range, as opposed to an excessively broad one, enhances the efficacy of the attack.

To elucidate the disparity in attack effectiveness between utilizing a broad versus a narrow object size range, we conducted experiments comparing the attack success rates with small and large range of the STOP sign size. We follow a similar evaluation setup as in §3.2 but use a pure simulation-based setup for RP_2 attack. Specifically, the small range for the STOP sign spans from 30 px to 100 px, whereas the large range extends from 30 px to 416 px, representing the maximum range at which the benign STOP sign is detectable. Results presented in Table 4 reveal a superior average attack success rate for the small range over the large range. Although the large range demonstrates promising convergence at close distances, its performance diminishes between 5 to 30 m. This suggests that simply opting for a larger range does not guarantee enhanced performance.

Our system-driven solution (S2). We introduce our Solution (S2) to ascertain the system-critical range from the *vehicle plant model* and the *AD system model*. With these, we directly deduce the d_{min} and d_{max} values as shown in Fig. 1. Then, we convert these distances to the corresponding object sizes in pixels ($\mathcal{S}$). From the system

model (Fig.1), it's evident that the minimum braking distance can be used as d_{min}. Within this distance, detection results have a negligible impact on system-level effects. As for the d_{max}, several tasks in the AD system, such as object detection and tracking, can influence its determination. For object detection, the maximum distance can be the furthest benign distance where an object is detected. For object tracking, we select a conservative tracking [28] since attackers might not always access the precise tracking parameters of the targeted AD system and a conservative tracking provides a broader system-critical range generally. To achieve system-level effects, the object should not be tracked when the vehicle reaches the d_{min}. Due to conservative tracking, such tracking distance (i.e., if within this distance, the object can never be detected, the tracker will be deleted) usually exceeds the distance where the object detector can detect the object. Thus, simplifying this, we select the distance where the benign object can be detected with a small detection rate as d_{max}. Having deduced the d_{min} and d_{max}, the next step involves translating these distances into pixel object sizes ($\mathcal{S}$) and determining the appropriate object location in the background (function $M(.)$ in §4.1). We suggest two methodologies to solve address it: 1) camera-based rendering [5, 8, 65] and 2) manual annotation [66]. Employing these methods allows us to acquire precise specifications about position and pixel size range (in §5.1 and §5.3).

5. Evaluation

We adopt the same evaluation methodology and setup as §3.2. The printed STOP signs with the newly generated patches are in Fig. 6. We evaluate some attacks on one-stage object detectors, i.e. Y2, Y3, and Y5 due to their better real-time performance compared to two-stage ones [74]. RP_2 and FTE are selected as the evaluated attacks. Attack generality is evaluated in §5.2 and §5.3. The combination for attacks and object detectors are RP_2, FTE-Y3, and FTE-Y5.

5.1. System-level Attack Effectiveness Evaluation

Attack generation. We adopt the attack methodology in §4. We employ a camera-based rendering method and utilize the nuScenes dataset [5] to translate the system-critical range from the physical world to the pixel range in images. Notably, nuScenes offers APIs that facilitate rendering objects within images. Specially, we render the four corners of the STOP sign and obtain its size in pixels by measuring the distance between these four corner points in the image. With S1 and S2, we can embed the system-model property into the attack generation process to improve system-level effects. To further validate the effects of S1 and S2, we perform ablation studies by generating the attack with S1 only and S2 only, and comparing them to the attack generated with/without both S1 and S2. Details of attacks without S1 and S2 (i.e., original attacks) are in §3.

Figure 6: Visualization of STOP sign attacks with system-driven design. S1: with S1 only; S2: with S2 only; S1 + S2: with both S1 and S2; TV*: S1 + S2 with TV loss (§5.1).

Results. The STOP sign attack images are in Fig. 6, which are printed in physical world and the perception modeling results from the physical world are in Table 5. From the results in Table 5, almost all the results (bolded in the table) with our system-driven attack improvement can outperform the original attack. As shown in Table 6, with our system-driven attack designs, the system-level violation rate can increase by around 70% on average, where we only include the results where the benign cases have a 0% system-level violation rate. The p-value (Table 6) is generally at the statistically significant level (e.g., generally < 0.05 or at a similar magnitude, especially for S1+S2). With S1 + S2, the overall component attack success rate can increase by around 33% on average. Especially, in the system critical range, the attack success rate can increase by 122%, which can significantly improve the system-level effects. Taking FTE-Y5 at 35 mph as an example, the brake distance of 35 mph is around 20 m and the attack success rate from 20 - 35 m shown in Table 5 is around 98%, which shows a high chance to make the STOP sign not tracked before the brake distance, which leads to the 100% violation rate (Table 6).

For FTE-Y5 at 25 mph, due to the low effectiveness (i.e., around 4%) from 10 m to 15 m, the tracker cannot be deleted, which leads to 0% system violation. Thus, we provide a *special improvement* by applying the total variation (TV) loss as prior works [16, 65] which benefits the attack effectiveness. The perception modeling results from the physical world are in Table 5 and the attack visualization is shown in Fig. 6. The system violation rate increases to 10% after improvement as shown in Table 6 with *. Based on results in Table 5, the attack success rate in a near distance is generally lower, which aligns well with the results of prior work [70]. This leaves space for future works: improving component attack success rate in the near distance.

The results of the ablation study are also summarized in Table 6. Although in the majority of cases, S1 cannot significantly improve the system-level effects (20% on average), the component attack success rate in the system-critical range is improved. Compared to S1, S2 has better results (around 28% on average). Only combining S1 and S2 can further benefit the system-level effects (around 70% on average), which shows the necessity of both S1 and S2.

5.2. Generality on Different AD System Parameters

Methodology and setup. We select the most safety-

Table 5: Attack success rates of RP_2, FTE-Y3, and FTE-Y5 on STOP sign-evasion attack (§5) and ADV-Tshirt on pedestrian-evasion attack (§5.3) for our attack design evaluation with perception results modeling from physical world. + S1: with S1 only; + S2: with S2 only; + S1 + S2: with S1 and S2; + S1 + S2 (TV): + S1 + S2 with TV loss (§5.1). **Bolded** numbers indicate the cases where our design outperforms the original baseline attack ("Original") within the system-critical range.

Object detector	Attack design	Distance (m): Gray color means the attack success rate within the system-critical range								
		4 - 5	5 - 10	10 - 15	15 - 20	20 - 25	25 - 30	30 - 35	35 - 40	40 - 45
YOLO v2 (Y2)	Original [16]	41.8%	10.0%	23.8%	65.4%	99.9%	100%	100%	100%	100%
	RP_2 + S1	4.4%	**13.7%**	**51.2%**	**99.3%**	100%	100%	100%	100%	100%
	RP_2 + S2	5.6%	**44.9%**	**57.8%**	**98.7%**	100%	100%	100%	100%	100%
	RP_2 + S1 + S2	36.1%	**65.8%**	**88.0%**	**100%**	100%	100%	100%	100%	100%
YOLO v3 (Y3)	Original [27]	10.1%	0%	0%	12.7%	57.1%	99.4%	100%	100%	100%
	FTE-Y3 + S1	0%	0%	0%	**14.0%**	**72.2%**	95.9%	100%	100%	100%
	FTE-Y3 + S2	0%	0%	0%	**13.4%**	**81.4%**	94.4%	97.2%	100%	100%
	FTE-Y3 + S1 + S2	5.3%	0%	**34.7%**	**94.0%**	**99.4%**	**100%**	100%	100%	100%
YOLO v5 (Y5)	Original [27]	8.8%	0%	0%	0.3%	11.8%	51.6%	96.1%	100%	100%
	FTE-Y5 + S1	0.3%	0%	0%	**1.3%**	**13.9%**	**69.3%**	94.1%	99.0%	100%
	FTE-Y5 + S2	1.5%	0%	**0.1%**	**1.7%**	**32.7%**	**81.9%**	**99.0%**	100%	100%
	FTE-Y5 + S1 + S2	16.5%	0%	**4.3%**	**47.2%**	**93.4%**	**99.7%**	**100%**	100%	100%
	FTE-Y5 + S1 + S2 (TV)	43.6%	**51.7%**	**42.1%**	**26.3%**	**23.8%**	**66.1%**	**97.7%**	99.7%	100%
YOLO v2 (Y2)	Original [66]	13.5%	0%	31.3%	86.1%	96.1%	90.7%	86.0%	100%	100%
	ADV-Tshirt + S1 + S2	0%	0%	**34.2%**	**89.0%**	91.7%	83.5%	78.5%	98.7%	100%
YOLO v3 (Y3)	Original [66]	3.8%	0%	3.8%	32.2%	75.7%	89.8%	90.5%	91.5%	95.1%
	ADV-Tshirt + S1 + S2	0%	0%	**33.6%**	**88.2%**	**91.3%**	**92.4%**	89.7%	90.7%	87.7%
YOLO v5 (Y5)	Original [66]	35.9%	6.8%	17.1%	36.7%	37.4%	72.0%	88.6%	92.3%	91.5%
	ADV-Tshirt + S1 + S2	2.6%	1.0%	**61.6%**	**74.7%**	**58.3%**	**89.6%**	**90.5%**	64.1%	61.1%

Table 6: System-level violation rate tested in simulation and component-level ASR evaluation including baseline comparison (i.e., Original and ablation studies). Each cell contains 10 runs with different initial positions of the AD vehicle. S1: with S1 only; S2: with S2 only; S1 + S2: with S1 and S2; SCR: System-critical range (§4.3). * with special improvements (§5.1).

Evaluation level	Speed (mph)	RP_2				FTE-Y3				FTE-Y5			
		Original [16]	S1	S2	S1 + S2	Original [27]	S1	S2	S1 + S2	Original [27]	S1	S2	S1 + S2
System (violation rate)	25	0%	90%	100%	100%	0%	0%	0%	40%	0%	0%	0%	10%*
	30	-	-	-	-	0%	0%	30%	100%	0%	0%	0%	80%
	35	-	-	-	-	-	-	-	-	0%	30%	40%	100%
p-value		-	0.00	0.00	0.00	-	-	0.08	0.00	-	0.08	0.04	0.00
Component (ASR)	Overall	71.2%	74.2%	78.6%	87.8%	53.3%	53.6%	54.0%	70.4%	41.0%	42.0%	46.3%	62.3%
	SCR	33.1%	54.7%	67.1%	84.6%	33.8%	36.4%	37.8%	65.6%	26.6%	29.8%	35.9%	57.4%

Table 7: System-level violation rate tested in simulation on different AD parameter settings which are highly critical to the system-level effects. The perception modeling results from physical world are in Table 3 and Table 6.

Tracking param (H, R)	(4, 6) [1]		(3, 5) [31]		(4, 40) [71]	
Brake (m/s^2)	-3.4	-6.0	-3.4	-6.0	-3.4	-6.0
Original [27]	20%	0%	50%	0%	40%	0%
Ours	**100%**	**100%**	**100%**	**90%**	**100%**	**100%**

critical parameters on system-level effects in AD systems for this evaluation including the tracking parameters (H, R), where the tracking creates a tracker for an object only when it is continuously detected for H frames, and deletes its tracker only when the object continuously disappears for R frames [1, 28, 31, 72], and the brake deceleration where we use the safe vehicle deceleration and max vehicle deceleration [13]. We select the tracking parameters from Baidu Apollo [1] and Autoware.AI [31], and the representative research paper [71]. All the parameter details are in Table 7 and for others, we follow the same setup in §5.1. We select the FTE-Y5 since it is the most representative attack so far and 35 mph as the target speed due to its high safety impact.

Results. The system-level attack effect results (violation rate) are summarized in Table 7, where we compared our attack with the original naive attacks (§3). The results show that our attack can outperform the original attack in all the different AD parameter settings on the system-level effect. On average, we have around 98% system violation rate (5 times larger than the original one) while the original naive

Table 8: Pedestrian collision rate tested in simulation with ADV-Tshirt attack on different object detectors. 10 runs for each cell with different initial AD vehicle position.

Speed (mph)	YOLO v2 (Y2)		YOLO v3 (Y3)		YOLO v5 (Y5)	
	Original [66]	**Ours**	Original [66]	**Ours**	Original [66]	**Ours**
25	20%	**50%**	0%	**50%**	0%	**70%**
30	100%	**100%**	50%	**100%**	10%	**80%**
35	100%	**100%**	80%	**100%**	60%	**90%**

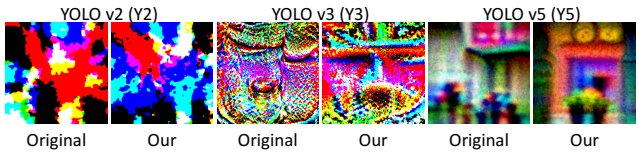


Figure 7: Visualization of ADV-Tshirt attack with and without system-driven design.

attack only has 18%. The results further point out that our attack is general to different critical AD system parameters.

5.3. Generality on a Different Object Types

Methodology and setup. We select the "pedestrian" as our target object type since making the pedestrian vanish will cause a significant impact on AD. We select the most representative patch attack – adversarial T-shirt [66], which is called ADV-Tshirt in our paper. For object detectors, we select Y2, Y3, and Y5 and follow the same setup in ADV-Tshirt paper [66], and we collect the videos from the real world (similar methodology in §3) for attack generation and manually annotate the four corner points for placing the patch (obtaining the size and position §4.3). Each video segment has around 200 frames. We perform digital perception result modeling with real-world data we collected.

Results. The perception results modeling results for ADV-Tshirt are shown in Table 5 and the generated patches are visualized in Fig. 7. We define the system-level effect metric as pedestrian collision rate: $\frac{K_{\text{collision}}}{K_{\text{total}}}$, in which $K_{\text{collision}}$ means the number of runs where the AD vehicle crash into the pedestrian, and K_{total} is the number of total runs. The system-level evaluation with the comparison with the original attack [66] is shown in Table 8. In average, our attack designs can achieve around 82% pedestrian collision rate while the original attack can only achieve around 47% pedestrian collision rate. Especially, for the most advanced object detector such as Y3 and Y5, our pedestrian collision rate has significant improvement compared to the original attack. Y2 is more fragile than others which makes the original attack have very high attack effectiveness in the component level and leads to pedestrian collision rates at the similar level as ours. The results show the generality of our attack designs to different object types which further shows the generality to different system models (§2).

6. Discussion

Potential mitigation. The ongoing tug-of-war between adversarial attacks and their defenses has yielded a range of mitigation strategies, such as adversarial training [39]. Since several object-evasion attacks in AD context have been identified [27, 70], there is an immediate need for defense exploration. Before pursuing novel mitigation strategies, it is imperative to first measure how existing defenses affects system-level attack effectiveness in AD, especially the ones with theoretical guarantees [63, 64], which should be a future work. Another promising direction involves cross-checking with alternate perception sources. For example, AD systems might verify camera-based pedestrian detection with LiDAR perception. Despite not offering a fundamental defense strategy [8], they may make system-level attack effects more difficult to achieve. Thus, we leave a systematic exploration of these defenses to future work.

Limitation and future work. First, although we leverage the perception results that modeling from the physical world and demonstrate the system-level effects in AD system with LGSVL, the feasibility of the attack effects on real AD systems in physical world remains unclear. Thus, the exploration of the attack practicality is a valuable future work. Second, our attack is within white-box threat model, which is less practical compared to black-box one. Thereby, the development of a novel attack with a practical threat model is a potential future work. Third, although we explore the generality on different AD system parameters in §5.2, our evaluation results and findings are limited by the current AD system setups introduced in §3.2. Therefore, the system-level effect measurement on commercial AD systems such as Tesla is an important future direction.

7. Conclusion

In this paper, we ask whether previous works can achieve system-level effects (e.g., vehicle collisions, traffic rule violations) under real AD settings. Then, we perform the first measurement study to answer this research question. Our evaluation results show that all representative prior works cannot achieve any system-level effects in a closed-loop AD setup due to the lack of the system model. With our newly proposed system-driven designs, i.e., SysAdv, the system-level effects can be significantly improved. We hope that the concept of the system model could guide future security analysis/testing for real/practical AD systems.

8. Acknowledgments

We would like to thank Ziwen Wan, Junjie Shen, Tong Wu, Junze Liu, Fayzah Alshammari, Trishna Chakraborty, and the anonymous reviewers for their valuable and insightful feedback. This research was supported by the NSF under grants CNS-1932464, CNS-1929771, and CNS-2145493.

References

[1] Baidu Apollo. Baidu Apollo. https://www.apollo.auto/, 2022. 4, 8

[2] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing Robust Adversarial Examples. In *International conference on machine learning*, pages 284–293. PMLR, 2018. 5

[3] DOT Atlanta. 25 MPH Speed Limit - Frequently Asked Questions, 2020. 4

[4] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial Patch. *arXiv preprint arXiv:1712.09665*, 2017. 3

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 2, 7

[6] Visit California. Driving in California. https://www.visitcalifornia.com/experience/driving-california/, 2022. 4

[7] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. 3D Adversarial Object against MSF-based Perception in Autonomous Driving. In *MLSys*, 2020. 1

[8] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. Invisible for both Camera and LiDAR: Security of Multi-Sensor Fusion based Perception in Autonomous Driving Under Physical World Attacks. In *IEEE S&P 2021*, May 2021. 1, 7, 9

[9] Nicholas Carlini and David A. Wagner. Towards Evaluating the Robustness of Neural Networks. In *Proceedings of IEEE Symposium on Security and Privacy (S&P)*, 2017. 2

[10] Manuel Carranza-García, Jesús Torres-Mateo, Pedro Lara-Benítez, and Jorge García-Gutiérrez. On the Performance of One-stage and Two-stage Object Detectors in Autonomous Vehicles using Camera Data. *Remote Sensing*, 13(1):89, 2020. 2

[11] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 3

[12] Shang-Tse Chen, Cory Cornelius, Jason Martin, and Duen Horng Polo Chau. ShapeShifter: Robust Physical Adversarial Attack on Faster R-CNN Object Detector. In *ECML PKDD*, pages 52–68. Springer, 2018. 1, 2, 3, 5, 6

[13] CopRadar.com. Vehicle Acceleration and Braking. https://copradar.com/chapts/references/acceleration.html, 2017. 8

[14] Christopher DiPalma, Ningfei Wang, Takami Sato, and Qi Alfred Chen. Security of Camera-based Perception for Autonomous Driving under Adversarial Attack. In *2021 IEEE Security and Privacy Workshops (SPW)*, pages 243–243. IEEE, 2021. 2

[15] Tommaso Dreossi, Daniel J. Fremont, Shromona Ghosh, Edward Kim, Hadi Ravanbakhsh, Marcell Vazquez-Chanlatte, and Sanjit A. Seshia. VerifAI: A toolkit for the formal design and analysis of artificial intelligence-based systems. In *31st International Conference on Computer Aided Verification (CAV)*, July 2019. 2, 3

[16] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Physical Adversarial Examples for Object Detectors. In *WOOT*, 2018. 1, 2, 3, 4, 5, 7, 8

[17] Ionel Gog, Sukrit Kalra, Peter Schafhalter, Matthew A Wright, Joseph E Gonzalez, and Ion Stoica. Pylot: A modular platform for exploring latency-accuracy tradeoffs in autonomous vehicles. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8806–8813. IEEE, 2021. 2

[18] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2015. 2

[19] Michael Greiffenhagen, Dorin Comaniciu, Heinrich Niemann, and Visvanathan Ramesh. Design, Analysis, and Engineering of Video Monitoring Systems: An Approach and a Case Study. *Proceedings of the IEEE*, 89(10):1498–1517, 2001. 2

[20] Michael Greiffenhagen, Visvanathan Ramesh, Dorin Comaniciu, and Heinrich Niemann. Statistical modeling and performance characterization of a real-time dual camera surveillance system. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 335–342. IEEE, 2000. 2

[21] Michael Greiffenhagen, Visvanathan Ramesh, and Heinrich Niemann. The systematic design and analysis cycle of a vision system: A case study in video surveillance. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 2, pages II–II. IEEE, 2001. 2

[22] Robert M Haralick. Performance characterization in image analysis: thinning, a case in point. *Pattern Recognition Letters*, 13(1):5–12, 1992. 2

[23] Gabriel M Hoffmann, Claire J Tomlin, Michael Montemerlo, and Sebastian Thrun. Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing. In *2007 American control conference*, pages 2296–2301. IEEE, 2007. 4

[24] Ramesh C Jain and Thomas O Binford. Ignorance, myopia, and naivete in computer vision systems. *CVGIP: Image Understanding*, 53(1):112–117, 1991. 2

[25] Viktor Lundqvist Jessie Smith. Scaling Simulation. https://aurora.tech/blog/scaling-simulation/, 2021. 4

[26] Qiang Ji and Robert M Haralick. Error propagation for computer vision performance characterization. In *International Conference on Imaging Science, Systems, and Technology, Las Vegas*, 1999. 2

[27] Wei Jia, Zhaojun Lu, Haichun Zhang, Zhenglin Liu, Jie Wang, and Gang Qu. Fooling the Eyes of Autonomous Vehicles: Robust Physical Adversarial Examples Against Traffic Sign Recognition Systems. *NDSS*, 2022. 1, 2, 3, 4, 5, 6, 8, 9

[28] Yunhan Jia Jia, Yantao Lu, Junjie Shen, Qi Alfred Chen, Hao Chen, Zhenyu Zhong, and Tao Wei Wei. Fooling Detection Alone is Not Enough: Adversarial Attack against Multiple Object Tracking. In *International Conference on Learning Representations (ICLR'20)*, 2020. 3, 7, 8

[29] Glenn Jocher. YOLOv5. https://github.com/ultralytics/yolov5, 2022. 2

[30] Mark Kane. Tesla Sold 2 Million Electric Cars: First Automaker To Reach Milestone, 2021. 1

[31] Shinpei Kato, Shota Tokunaga, Yuya Maruyama, Seiya Maeda, Manato Hirabayashi, Yuki Kitsukawa, Abraham Monrroy, Tomohito Ando, Yusuke Fujii, and Takuya Azumi. Autoware on Board: Enabling Autonomous Vehicles with Embedded Systems. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, pages 287–296. IEEE, 2018. 8

[32] Giulio Lovisotto, Henry Turner, Ivo Sluganovic, Martin Strohmeier, and Ivan Martinovic. {SLAP}: Improving physical adversarial examples with {Short-Lived} adversarial perturbations. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 1865–1882, 2021. 1

[33] Jiajun Lu, Hussein Sibai, and Evan Fabry. Adversarial Examples that Fool Detectors. *arXiv preprint arXiv:1712.02494*, 2017. 3

[34] Jiajun Lu, Hussein Sibai, Evan Fabry, and David Forsyth. No Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles. *arXiv preprint arXiv:1707.03501*, 2017. 3

[35] Wenhan Luo, Junliang Xing, Anton Milan, Xiaoqin Zhang, Wei Liu, and Tae-Kyun Kim. Multiple Object Tracking: A Literature Review. *AI*, 293:103448, 2021. 4

[36] Yunpeng Luo, Ningfei Wang, Bo Yu, Shaoshan Liu, and Qi Alfred Chen. WIP: Infrastructure-Aided Defense for Autonomous Driving Systems: Opportunities and Challenges. In *NDSS Workshop on Automotive and Autonomous Vehicle Security (AutoSec)*, 2022. 2

[37] Chen Ma, Ningfei Wang, Qi Alfred Chen, and Chao Shen. WIP: Towards the Practicality of the Adversarial Attack on Object Tracking in Autonomous Driving. In *ISOC Symposium on Vehicle Security and Privacy (VehicleSec)*, 2023. 2

[38] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. *stat*, 1050:9, 2017. 2

[39] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 9

[40] Raghavendra Nese, Nabal Kishore Pandey, and Satish Thimmalapura. A systematic approach of vehicle plant model development for vehicle virtual testing & calibration. In *2015 IEEE International Transportation Electrification Conference (ITEC)*, pages 1–10. IEEE, 2015. 2

[41] OpenPilot. OpenPilot. https://github.com/commaai/openpilot, 2022. 4

[42] Jonah Philion, Amlan Kar, and Sanja Fidler. Learning to Evaluate Perception Models using Planner-Centric Metrics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14055–14064, 2020. 2

[43] John Phillips, Julieta Martinez, Ioan Andrei Bârsan, Sergio Casas, Abbas Sadat, and Raquel Urtasun. Deep multi-task learning for joint localization, perception, and prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4679–4689, 2021. 2

[44] Visvanathan Ramesh, RM Haralick, AS Bedekar, X Liu, DC Nadadur, KB Thornton, and X Zhang. Computer vision performance characterization. *RADIUS: Image Understanding for Imagery Intelligence*, pages 241–282, 1997. 2

[45] Joseph Redmon and Ali Farhadi. YOLO9000: Better, Faster, Stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017. 2

[46] Joseph Redmon and Ali Farhadi. YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2

[47] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in neural information processing systems*, 28, 2015. 2

[48] Guodong Rong, Byung Hyun Shin, Hadi Tabatabaee, Qiang Lu, Steve Lemke, Mārtiņš Možeiko, Eric Boise, Geehoon Uhm, Mark Gerow, Shalin Mehta, et al. LGSVL Simulator: A High Fidelity Simulator for Autonomous Driving. In *2020 IEEE 23rd International conference on intelligent transportation systems (ITSC)*, pages 1–6. IEEE, 2020. 4

[49] Takami Sato, Junjie Shen, Ningfei Wang, Yunhan Jia, Xue Lin, and Qi Alfred Chen. Dirty Road Can Attack: Security of Deep Learning based Automated Lane Centering under Physical-World Attack. In *USENIX Security*, 2021. 4

[50] Takami Sato, Junjie Shen, Ningfei Wang, Yunhan Jack Jia, Xue Lin, and Qi Alfred Chen. Hold Tight and Never Let Go: Security of Deep Learning based Automated Lane Centering under Physical-World Attack. *arXiv preprint arXiv:2009.06701*, 2020. 2

[51] Takami Sato, Junjie Shen, Ningfei Wang, Yunhan Jack Jia, Xue Lin, and Qi Alfred Chen. WIP: Deployability Improvement, Stealthiness User Study, and Safety Impact Assessment on Real Vehicle for Dirty Road Patch Attack. In *Workshop on Automotive and Autonomous Vehicle Security (AutoSec)*, volume 2021, page 25, 2021. 2

[52] Sanjit A Seshia, Dorsa Sadigh, and S Shankar Sastry. Towards Verified Artificial Intelligence. *Communications of the ACM*, 65(7):46–55, 2022. 2, 3

[53] Junjie Shen, Ningfei Wang, Ziwen Wan, Yunpeng Luo, Takami Sato, Zhisheng Hu, Xinyang Zhang, Shengjian Guo, Zhenyu Zhong, Kang Li, et al. SoK: On the Semantic AI Security in Autonomous Driving. *arXiv*, 2022. 1, 2, 3

[54] Tesla. Future of Driving. https://www.tesla.com/autopilot, 2022. 1

[55] Neil A Thacker, Adrian F Clark, John L Barron, J Ross Beveridge, Patrick Courtney, William R Crum, Visvanathan

Ramesh, and Christine Clark. Performance characterization in computer vision: A guide to best practices. *Computer vision and image understanding*, 109(3):305–334, 2008. 2

[56] Sever Topan, Karen Leung, Yuxiao Chen, Pritish Tupekar, Edward Schmerling, Jonas Nilsson, Michael Cox, and Marco Pavone. Interaction-dynamics-aware perception zones for obstacle detection safety evaluation. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pages 1201–1210. IEEE, 2022. 2

[57] Ziwen Wan, Junjie Shen, Jalen Chuang, Xin Xia, Joshua Garcia, Jiaqi Ma, and Qi Alfred Chen. Too Afraid to Drive: Systematic Discovery of Semantic DoS Vulnerability in Autonomous Driving Planning under Physical-World Attacks. In *Network and Distributed System Security (NDSS) Symposium, 2022*, April 2022. 4

[58] Donghua Wang, Tingsong Jiang, Jialiang Sun, Weien Zhou, Zhiqiang Gong, Xiaoya Zhang, Wen Yao, and Xiaoqian Chen. Fca: Learning a 3d full-coverage vehicle camouflage for multi-view physical adversarial attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2414–2422, 2022. 3

[59] Ningfei Wang, Yunpeng Luo, Takami Sato, Kaidi Xu, and Qi Alfred Chen. Poster: On the System-Level Effectiveness of Physical Object-Hiding Adversarial Attack in Autonomous Driving. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 3479–3481, 2022. 2

[60] Wei Wang, Yao Yao, Xin Liu, Xiang Li, Pei Hao, and Ting Zhu. I Can See the Light: Attacks on Autonomous Vehicles Using Invisible Lights. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 1930–1944, 2021. 2

[61] Waymo. Waymo Safety Report. https://storage.googleapis.com/waymo-uploads/files/documents/safety/2021-12-waymo-safety-report.pdf, 2021. 4

[62] Zuxuan Wu, Ser-Nam Lim, Larry S Davis, and Tom Goldstein. Making an invisibility cloak: Real world adversarial attacks on object detectors. In *European Conference on Computer Vision*, pages 1–17. Springer, 2020. 1, 2

[63] Chong Xiang and Prateek Mittal. DetectorGuard: Provably Securing Object Detectors against Localized Patch Hiding Attacks. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 3177–3196, 2021. 9

[64] Chong Xiang, Alexander Valtchanov, Saeed Mahloujifar, and Prateek Mittal. ObjectSeeker: Certifiably Robust Object Detection against Patch Hiding Attacks via Patch-agnostic Masking. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 1329–1347. IEEE, 2023. 9

[65] Chaowei Xiao, Dawei Yang, Bo Li, Jia Deng, and Mingyan Liu. MeshAdv: Adversarial Meshes for Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6898–6907, 2019. 2, 7

[66] Kaidi Xu, Gaoyuan Zhang, Sijia Liu, Quanfu Fan, Mengshu Sun, Hongge Chen, Pin-Yu Chen, Yanzhi Wang, and Xue Lin. Adversarial T-shirt!Evading Person Detectors in A Physical World. In *European conference on computer vision*, pages 665–681. Springer, 2020. 1, 2, 3, 7, 8, 9

[67] Mingfu Xue, Chengxiang Yuan, Can He, Jian Wang, and Weiqiang Liu. NaturalAE: Natural and Robust Physical Adversarial Examples for Object Detectors. *Journal of Information Security and Applications*, 57:102694, 2021. 3

[68] Quan-shi Zhang and Song-Chun Zhu. Visual interpretability for deep learning: a survey. *Frontiers of Information Technology & Electronic Engineering*, 19(1):27–39, 2018. 2

[69] Xinyang Zhang, Ningfei Wang, Hua Shen, Shouling Ji, Xiapu Luo, and Ting Wang. Interpretable deep learning under fire. In *29th {USENIX} security symposium ({USENIX} security 20)*, 2020. 2

[70] Yue Zhao, Hong Zhu, Ruigang Liang, Qintao Shen, Shengzhi Zhang, and Kai Chen. Seeing isn't Believing: Towards More Robust Adversarial Attack Against Real World Object Detectors. In *ACM CCS 2019*, pages 1989–2004, 2019. 1, 2, 3, 4, 5, 7, 9

[71] Ji Zhu, Hua Yang, Nian Liu, Minyoung Kim, Wenjun Zhang, and Ming-Hsuan Yang. Online Multi-Object Tracking with Dual Matching Attention Networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 366–382, 2018. 8

[72] Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. Distractor-aware siamese networks for visual object tracking. In *ECCV*, 2018. 8

[73] Alon Zolfi, Moshe Kravchik, Yuval Elovici, and Asaf Shabtai. The Translucent Patch: A Physical and Universal Attack on Object Detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15232–15241, 2021. 2

[74] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 2023. 2, 7