Poster: SlowTrack: Increasing the Latency of Camera-Based Perception in Autonomous Driving Using Adversarial Examples

Chen Ma[1⋆] Ningfei Wang[2⋆] Qi Alfred Chen[2] Chao Shen[1]
⋆These authors contributed equally
[1]Xi'an Jiaotong University [2]University of California, Irvine
ershang@stu.xjtu.edu.cn ningfei.wang@uci.edu alfchen@uci.edu chaoshen@xjtu.edu.cn

Abstract

In Autonomous Driving (AD), real-time perception is a critical component responsible for detecting surrounding objects to ensure safe driving. While researchers have extensively explored the integrity of AD perception due to its safety and security implications, the aspect of availability (real-time performance) or latency has received limited attention. Existing works on latency-based attack have focused mainly on *object detection*, i.e., a component in camera-based AD perception, overlooking the entire camera-based AD perception, which hinders them to achieve effective system-level effects, such as vehicle crashes. In this paper, we propose SlowTrack, a novel framework for generating adversarial attacks to increase the execution time of camera-based AD perception. We propose a novel two-stage attack strategy along with the three new loss function designs. Our evaluation is conducted on four popular camera-based AD perception pipelines, and the results demonstrate that SlowTrack significantly outperforms existing latency-based attacks while maintaining comparable imperceptibility levels. Furthermore, we perform the evaluation on Baidu Apollo, an industry-grade full-stack AD system, and LGSVL, a production-grade AD simulator, with two scenarios to compare the system-level effects of SlowTrack and existing attacks. Our evaluation results show that the system-level effects can be significantly improved, i.e., the vehicle crash rate of SlowTrack is around 95% on average while existing works only have around 30%.

I. Main content

This research [1] is recently published in AAAI 2024. The original abstract and author list are shown above. We post the paper links with arXiv version[1].

II. Acknowledgments

We would like to thank Ziwen Wan, Yunpeng Luo, Tong Wu, Xinyang Zhang, and the anonymous reviewers for their valuable and insightful feedback. This research was supported in part by National Key R&D Program of China (2020AAA0107702); the NSF under grants CNS-1929771, CNS-2145493, and CNS-1932464; USDOT UTC Grant 69A3552348327; National Natural Science Foundation of China (U21B2018, 62161160337, 62132011, 62376210, 62006181, U20B2049); Shaanxi Province Key Industry Innovation Program (2021ZDLGY01-02); and Fundamental Research Funds for the Central Universities under grant (xtr052023004, xtr022019002). Chao Shen and Qi Alfred Chen are the corresponding authors.

References

[1] C. Ma, N. Wang, Q. A. Chen, and C. Shen, "SlowTrack: Increasing the Latency of Camera-Based Perception in Autonomous Driving Using Adversarial Examples," in *Proceedings of AAAI*, 2024.

[1]https://arxiv.org/abs/2312.09520

Poster: SlowTrack: Increasing the Latency of Camera-Based Perception in Autonomous Driving Using Adversarial Examples

Chen Ma[1*], Ningfei Wang[2*], Qi Alfred Chen[2], Chao Shen[1]

[*] These authors contributed equally

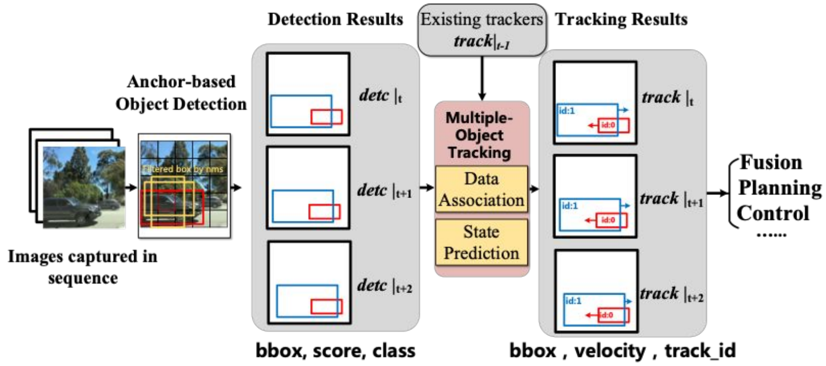
[1]Xi'an Jiaotong University [2]University of California, Irvine (UCI)

Accepted by AAAI 2024

SlowTrack

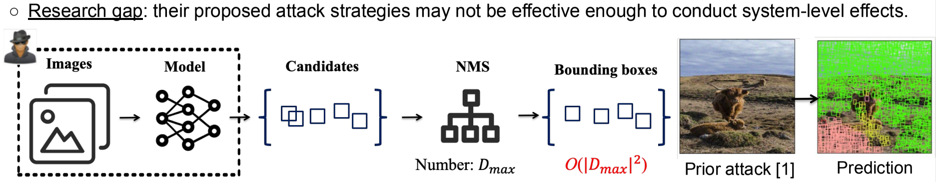
Autonomous Driving (AD) Visual Perception

- AD visual perception consists of **object detection** and **object tracking**.



Research Gap

- ***Autonomous Driving (AD) Perception*** is safety-critical
  - Many prior works have studied its security, especially on integrity: e.g., object evasion attack
    - Cause traffic rule violations or crashes, which is called system-level effect
  - However, availability aspect has been relatively underexplored
  - While some prior works have studied availability in object detection, they do not encompass entire AD perception, including both object detection and tracking
  - Research gap: their proposed attack strategies may not be effective enough to conduct system-level effects.

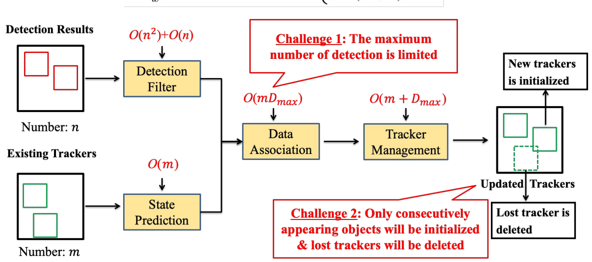


Problem Formulation & Tracking Analysis

- Under the constraint of limiting the maximum number of detection boxes, we formulate the SlowTrack

$$\arg\max_{x^*} T(P(x^*)) \quad \text{s.t.} \quad \begin{cases} |\mathcal{D}|_{max} = N \\ \Delta(x^*, x) \le \epsilon \end{cases}$$

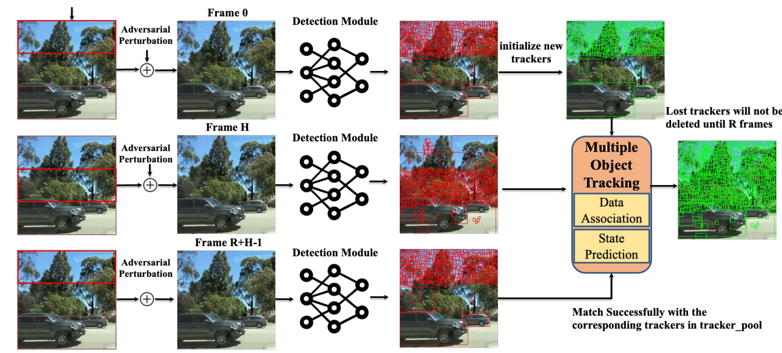
- Tracking analysis



[1] Chen et al., Overload: Latency Attacks on Object Detection for Edge Devices, arXiv preprint arXiv:2304.05370 (2023)

SlowTrack Attack Design

- Propose SlowTrack to systematically generate latency-based adversarial attacks on AD perception



Attack Effectiveness Evaluation

- Evaluation setup
  - Dataset: MOT17DET and BDD
  - Model: SORT (Y5), FairMOT, ByteTrack, and BoT-SORT
  - Baseline: PS [2] and Overload [1]
- Evaluation metric: R-Track, R-Lat, and #Track

$$\text{R-Track} = \frac{\text{Track-Lat}(x^*) - \text{Track-Lat}(x)}{\text{Track-Lat}(x)}$$

$$\text{R-Lat} = \frac{\text{Total-Lat}(x^*) - \text{Total-Lat}(x)}{\text{Total-Lat}(x)}$$

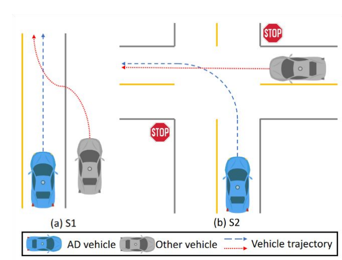
$$\#\text{Track} = \frac{\text{Tracker\#}(x^*) - \text{Tracker\#}(x)}{\text{Tracker\#}(x)}$$

Effectiveness results SlowTrack with two baselines [1] [2] and average L2 norm in different models and hardware. Bold denotes the best results (i.e., highest R-Track, R-Lat, #Track, and lowest L2) in each row.

Model	Dataset	Hardware	PS [2]				Overload [1]				SlowTrack			
			R-Track	R-Lat	#Track	$\mathcal{L}_2$	R-Track	R-Lat	#Track	$\mathcal{L}_2$	R-Track	R-Lat	#Track	$\mathcal{L}_2$
SORT (Y5)	BDD	Titan V	73.1	6.2	69.1	0.042	156.1	12.1	94.6	0.011	**247.0**	**17.5**	**141.8**	**0.010**
		2080 Ti	64.9	11.7			205.1	34.7			**310.3**	**49.4**		
		3090	76.4	6.2			186.3	13.8			**338.3**	**23.0**		
	MOT17	Titan V	37.5	3.8	32.4	0.041	93.0	8.7	47.0	**0.013**	**208.4**	**18.5**	**73.5**	**0.013**
		2080 Ti	33.2	8.6			82.6	19.3			**225.5**	**50.9**		
		3090	39.8	3.7			111.1	9.8			**283.9**	**25.7**		
FairMOT	BDD	Titan V	517.7	10.4	401.1	0.036	1505.9	29.1	939.9	0.029	**2647.6**	**51.3**	**1334.9**	**0.028**
		2080 Ti	390.1	8.7			1258.7	26.7			**2260.7**	**48.1**		
		3090	236.0	3.8			836.4	13.3			**1572.9**	**25.2**		
	MOT17	Titan V	67.4	8.6	48.7	0.036	181.7	21.5	106.2	**0.026**	**341.0**	**41.5**	**159.0**	**0.026**
		2080 Ti	49.3	7.1			149.0	19.9			**279.5**	**38.4**		
		3090	32.4	3.2			108.5	10.1			**220.7**	**21.3**		
ByteTrack	BDD	Titan V	40.2	2.5	79.2	0.032	173.0	9.2	224.7	0.027	**290.0**	**14.7**	**307.0**	**0.022**
		2080 Ti	39.5	2.3			184.0	9.0			**266.4**	**12.5**		
		3090	57.8	3.0			217.7	10.1			**341.4**	**15.1**		
	MOT17	Titan V	29.0	1.9	38.0	0.030	95.0	5.4	78.3	0.025	**173.0**	**9.8**	**130.1**	**0.022**
		2080 Ti	26.9	1.8			97.7	5.4			**168.7**	**9.5**		
		3090	45.2	2.4			115.2	5.9			**204.0**	**10.5**		
BoT-SORT	BDD	Titan V	53.2	19.4	70.0	0.033	79.0	28.9	221.9	0.029	**92.1**	**33.6**	**280.8**	**0.025**
		2080 Ti	57.6	19.6			89.1	30.6			**103.6**	**35.5**		
		3090	61.0	22.1			93.9	34.1			**135.3**	**49.7**		
	MOT17	Titan V	31.7	14.0	40.7	0.034	42.1	18.5	83.9	**0.025**	**52.2**	**23.0**	**127.6**	**0.025**
		2080 Ti	34.2	14.6			45.6	19.5			**59.1**	**25.3**		
		3090	44.0	19.9			47.2	21.1			**69.4**	**30.8**		

End-to-end Simulation Evaluation Results

- Baidu Apollo and LGSVL simulator
- Two scenarios S1 and S2



- Results
  - SlowTrack can achieve 95% vehicle crash rate on average while for other attacks, they can only have around 30%

System-level evaluation (vehicle crash rate). 10 runs for each cell.

Model	S1			S2		
	PS	Overload	SlowTrack	PS	Overload	SlowTrack
SORT(Y5)	10%	30%	**100%**	20%	40%	**90%**
FairMOT	20%	40%	**100%**	20%	40%	**100%**
ByteTrack	0%	40%	**80%**	0%	40%	**90%**
BoT-SORT	40%	50%	**100%**	50%	50%	**100%**

[2] Shapira et al., Phantom Sponges: Exploiting Non-Maximum Suppression To Attack Deep Object Detectors, WACV 2023