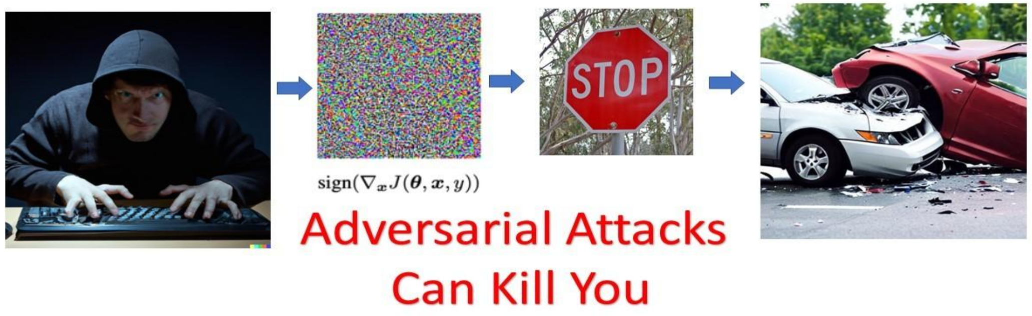
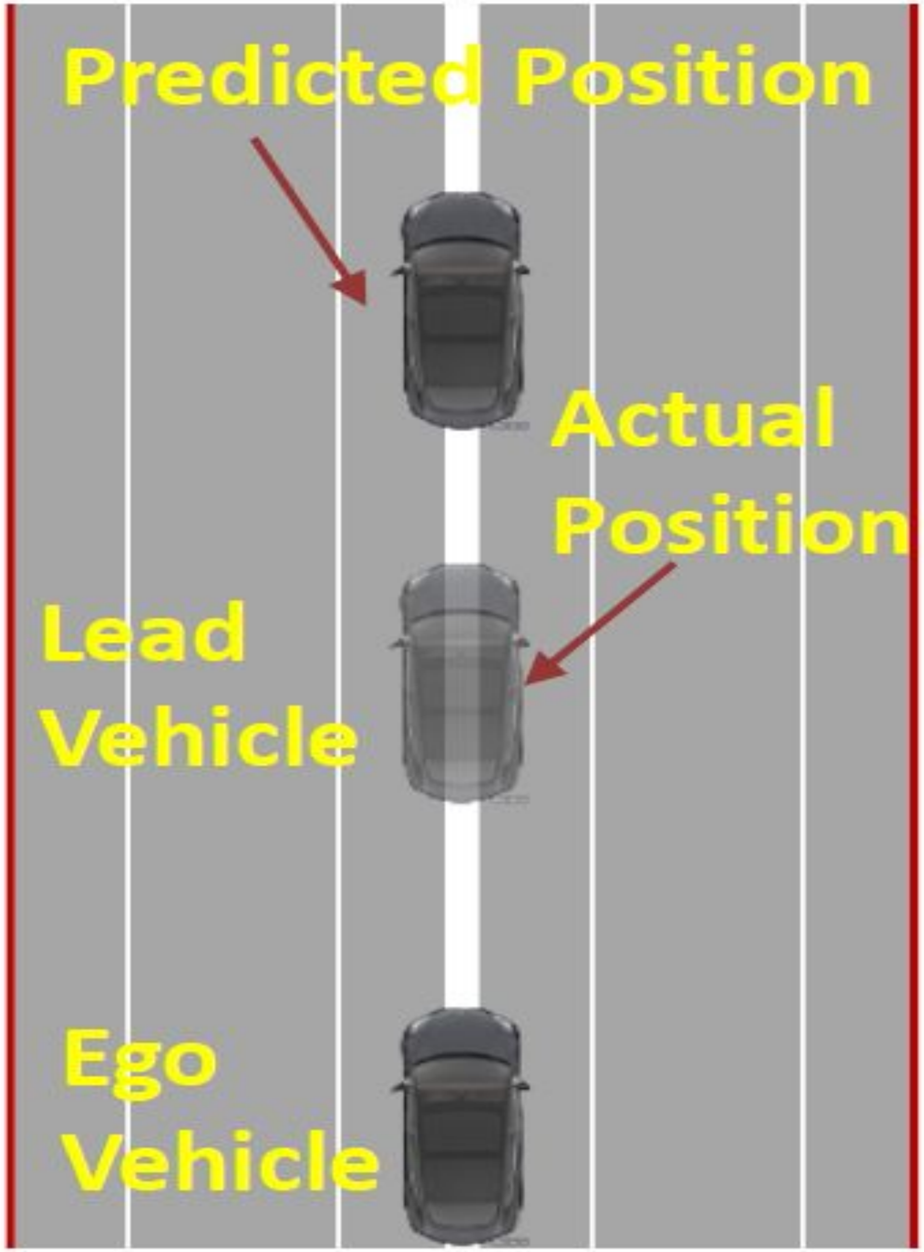
Runtime Stealthy Perception Attacks against DNN-based Adaptive Cruise Control Systems

Xugui Zhou, Anqi Chen, Maxfield Kouzel, Morgan McCarty, Cristina Nita-Rotaru, Homa Alemzadeh



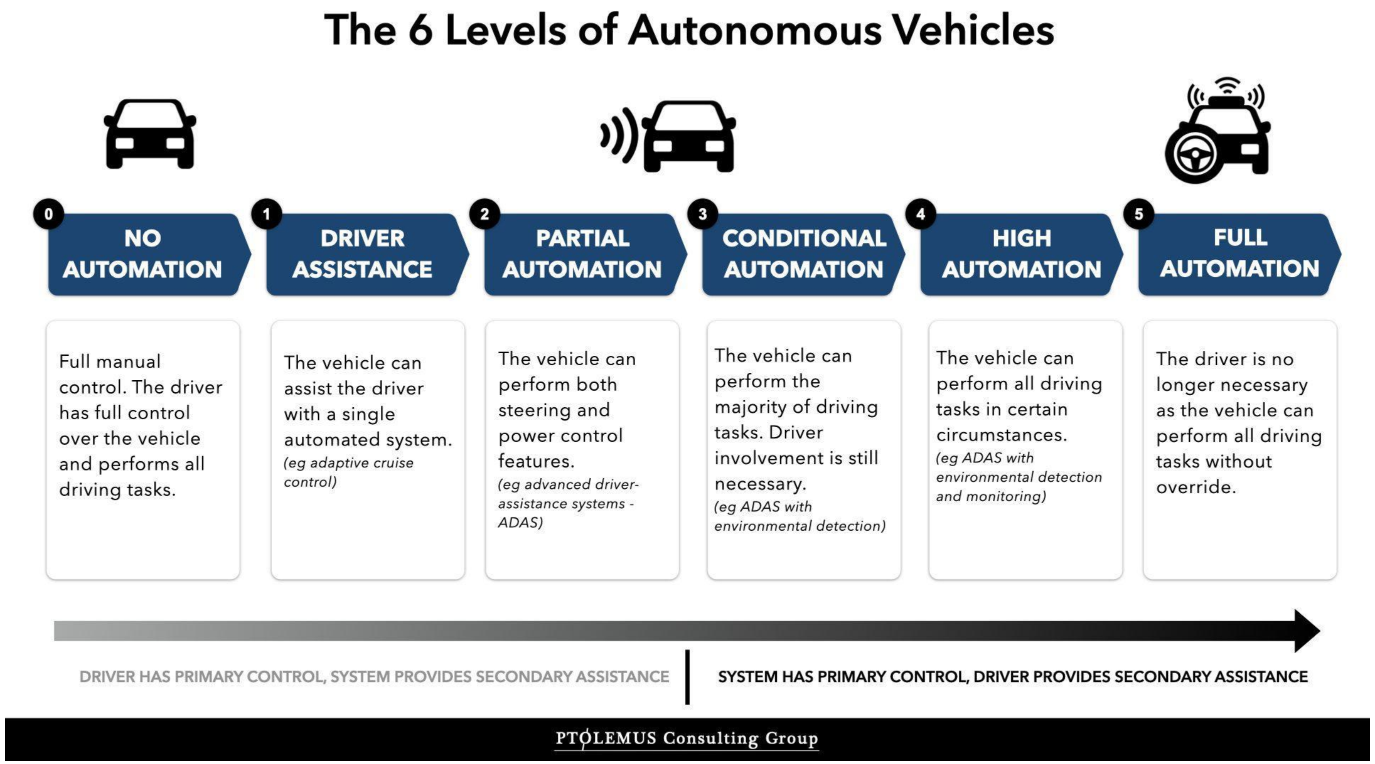


Presenter: Obiora Odugu

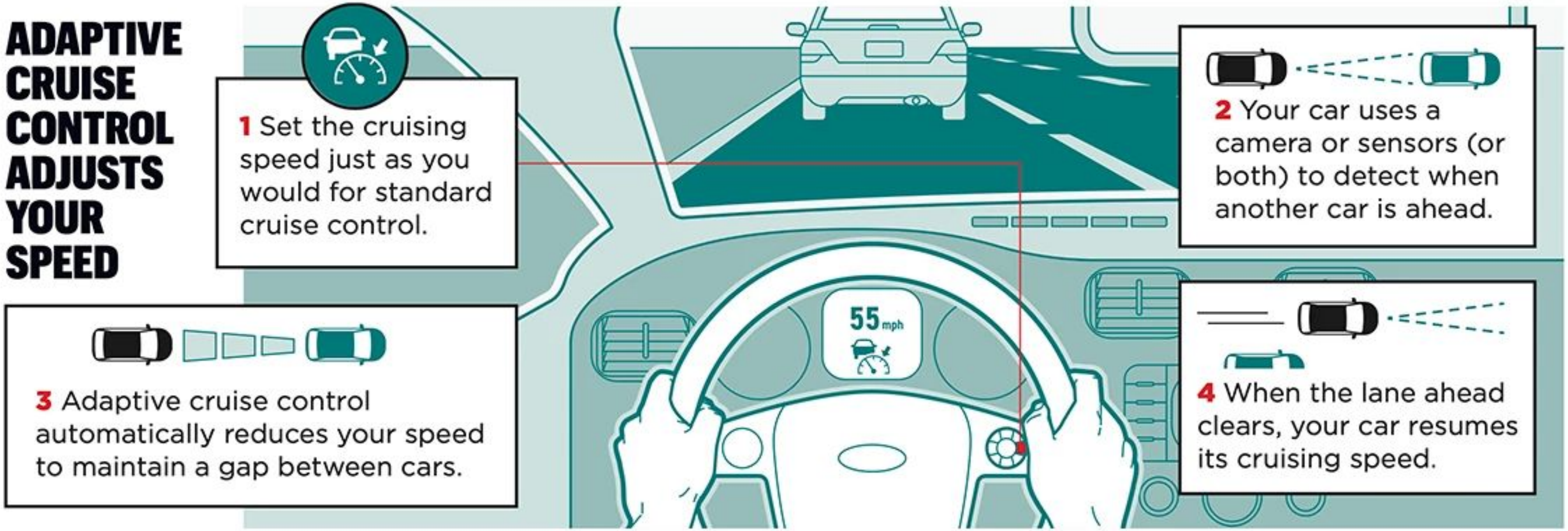
INTRODUCTION

Level-2 Advanced Driver Assistance Systems (ADAS)

- Adaptive Cruise Control (ACC) which controls longitudinal movement
- Automatic Lane Centering (ALC) which controls lateral movement
- Advanced Emergency Braking System (AEBS)
  - Automatic Emergency Braking (AEB)
  - Forward Collision Warning (FCW).



INTRODUCTION



ACC takes as input sensor measurements such as radar, Lidar, or camera and adjusts the speed to maintain a safe following distance to the lead vehicle. At the core of ACC lies the detection and tracking of the lead vehicle.

MOTIVATION

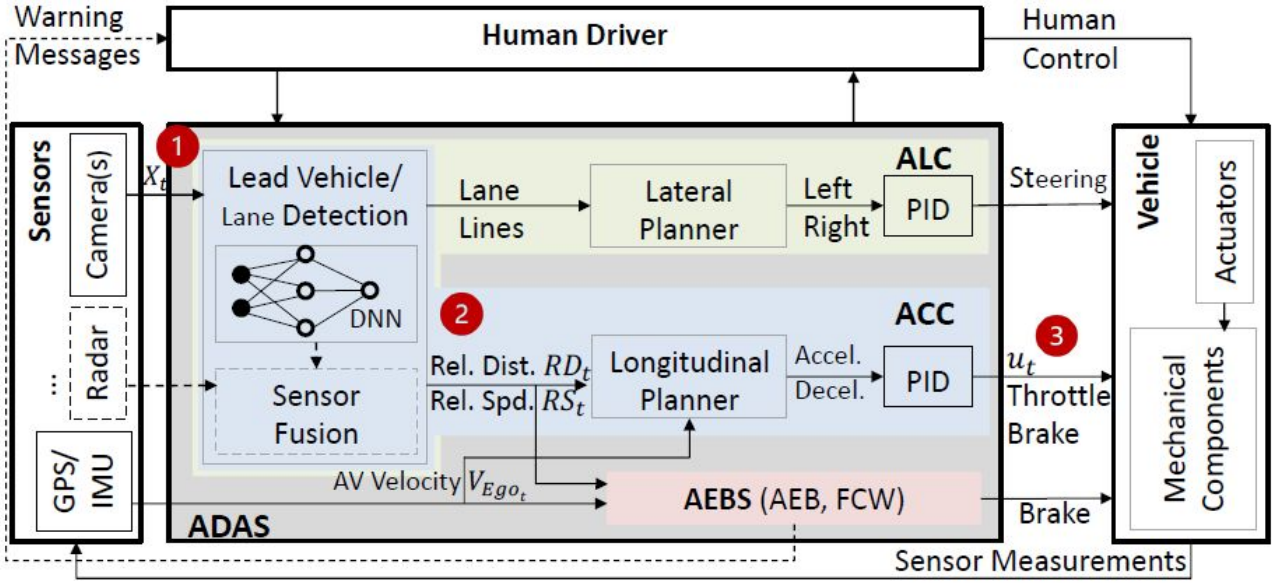
- Critical role of object detection and tracking in ACC
- Offline optimizations.
- Noticeable or preventable by human drivers
- Safety interventions And anomaly detection methods

AIM

Explore vulnerability with human in the loop

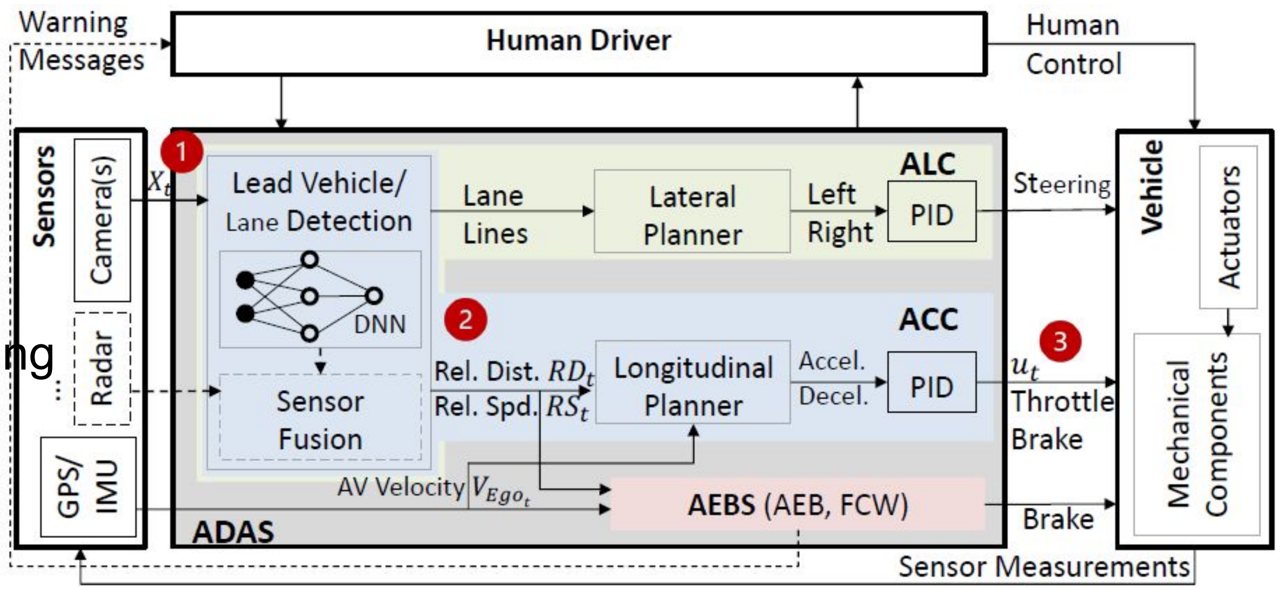
CONTRIBUTIONS

- Determine the best scenario to launch an attack that can lead to collision
- Dynamic optimization-based attack
- Simulation and real world-based evaluation with safety considerations



ADAPTIVE CRUISE CONTROL (ACC)

- Sensors.
  - Cameras, radar, IMU, GPS, LIDAR
- Lead Vehicle Detection
  - relative speed (RS) and distance (RD)
- Longitudinal Planner.
  - LVD outputs: acceleration, deceleration, braking
  - Speed trajectories
- Vehicle Control.
  - lowest speed and risk
  - new state st+1.



ATTACK MODEL

- Focus on DNN inputs to enhance stealthiness
- Attacker Constraints.
  - Modifying live camera feed
- Attacker knowledge
  - Access to ACC system design
  - Intercept and change live camera image frames at runtime
    - Over-air update
    - Remote access
    - Physical attacks via projections

Table 2: Threat models: attacker strength, capability, and impact.

Threat Model	Attacker Strength	Access to ADAS Software	Vehicular Networks	Computation Location	Impact	Examples
Malware	Strong[1]	✓	r/w*	within ADAS	Fleet of Vehicles	[44, 52]
Wireless	Medium[2]		r/w	Local Device, Remote Server	Single Vehicle	[53][54][19] [46][55]
Physical	Weak[3]		r	Remote Server	Single Vehicle	[56][57] [58][59][60]

ATTACK CHALLENGES

- C1 Optimal timing of attacks at runtime to cause safety hazards.
  - no LV is detected
- C2 Generating attack value at runtime to adapt to dynamic changes in the driving environment.
  - Fixed size and vehicle due to offline planning
- C3 Incorporating real-time constraints into the attack optimization process
  - real-time before the next frame

ATTACK DESIGN

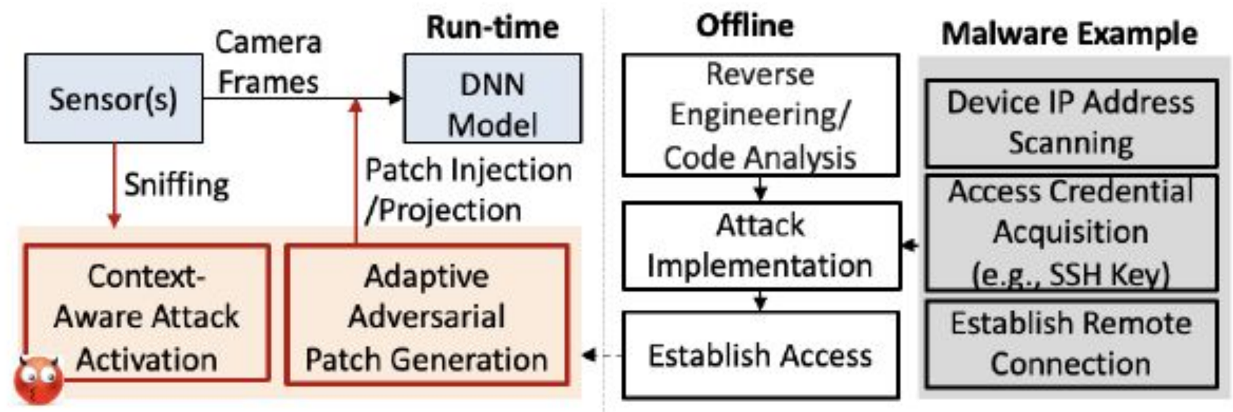


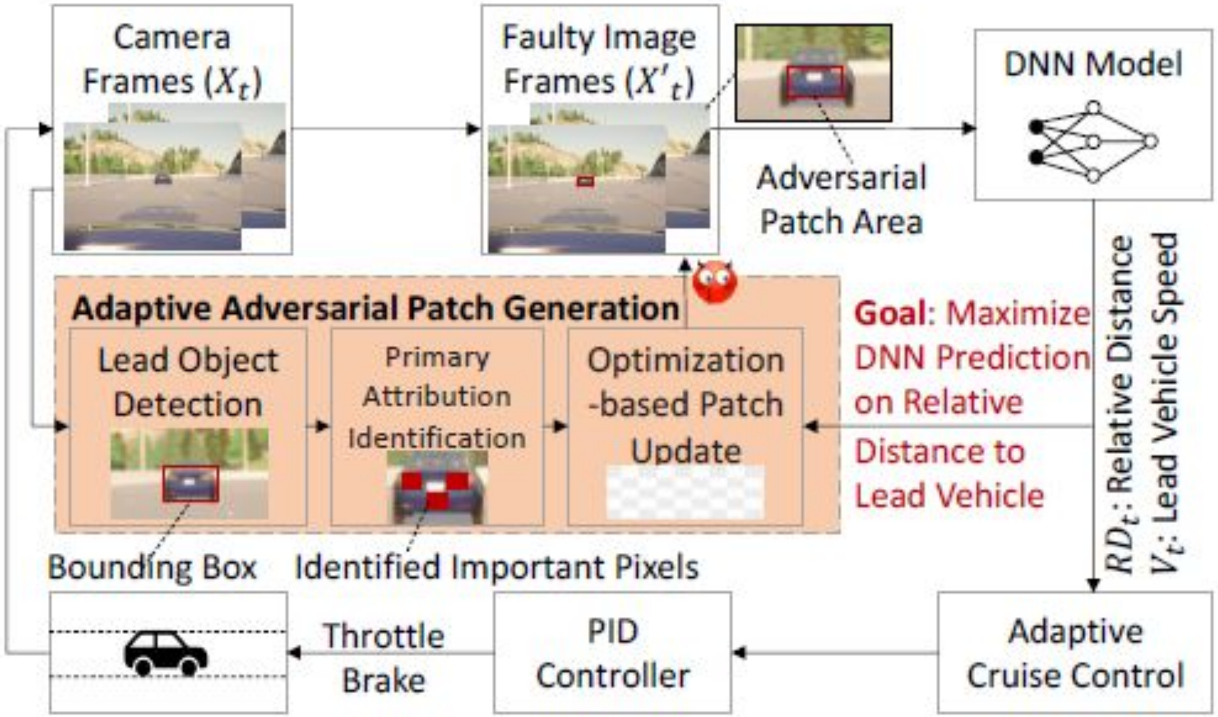
Table 3: Partial safety context table for an ACC system.

Rule	System Context		Control Action	Potential Hazards?
1	$HWT \leqslant HWT_{safe}$	$RS \leqslant 0$	Acceleration	No
2		$RS > 0$		Yes
3	$HWT > HWT_{safe}$	$RS \leqslant 0$		No
4		$RS > 0$		No

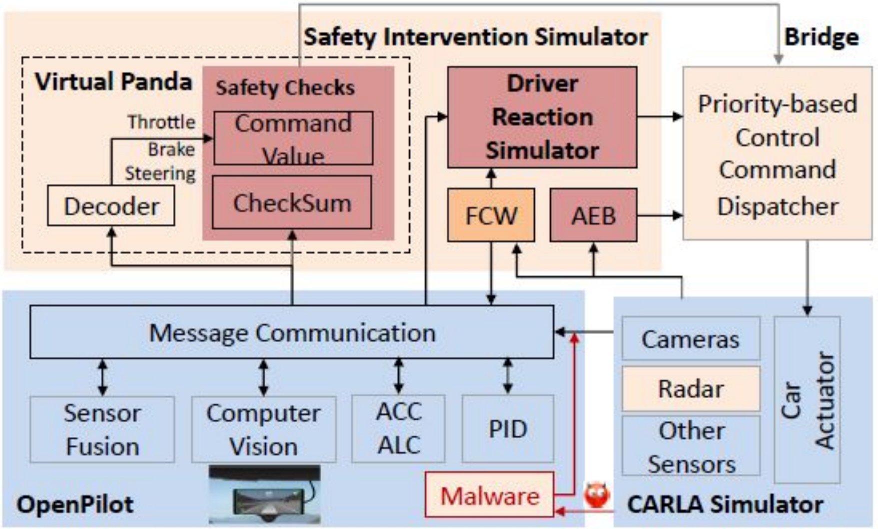
* HWT: Headway Time = Relative Distance/Current Speed;
* RS: Relative Speed = Current Speed (V_{Ego}) - Lead Speed (V_{Lead});

ATTACK DESIGN

$$
\begin{aligned}
\min \sum_{d \in RD_t} & -\nabla g(d,\theta) + \lambda ||\Delta_t||_p \\
s.t.\ Patch_t &= \Delta_t * M_t \\
Patch_t &\in [\mu - \sigma, \mu + \sigma] \\
Area(Patch_t) &\subset BBox(LV)_t \\
X_t^{adv} &= X_t + Patch_t \\
[RD, RS]_t &= LVD_\theta(X_{t-1}^{adv}) \\
u_t &= ACC(s_t, [RD, RS]_t) \\
s_{t+1} &= CarModel(s_t, u_t)
\end{aligned}
$$



SAFETY INTERVENTION SIMULATION



- AEBS is enabled, and AEBS camera data is uncompromised
- AEBS is enabled, but AEBS camera data is compromised
- AEBS is disabled

Table 4: Driver simulator: activation conditions and reactions.

Activation Condition	Driver Reaction	Reaction Time
Alerts (e.g., FCW) Unexpected Acceleration Unsafe Cruise Speed Unsafe Following Distance Obvious Camera Perturbation	Emergency Brake & Zero Throttle No changes in the steering angle	2.5 seconds
Hard Braking	Stop brake and output regular throttle No changes in the steering angle	2.5 seconds

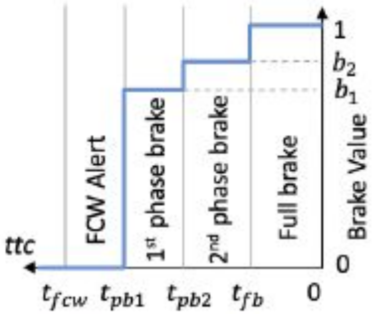


Figure 7: AEBS.

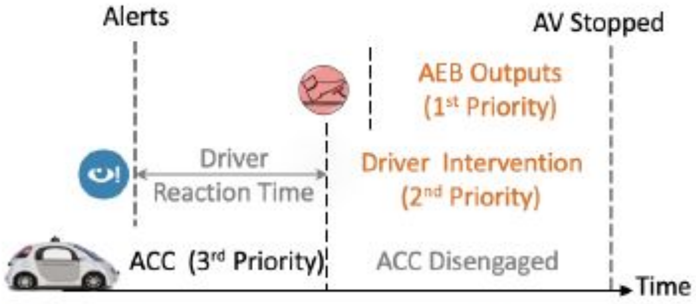
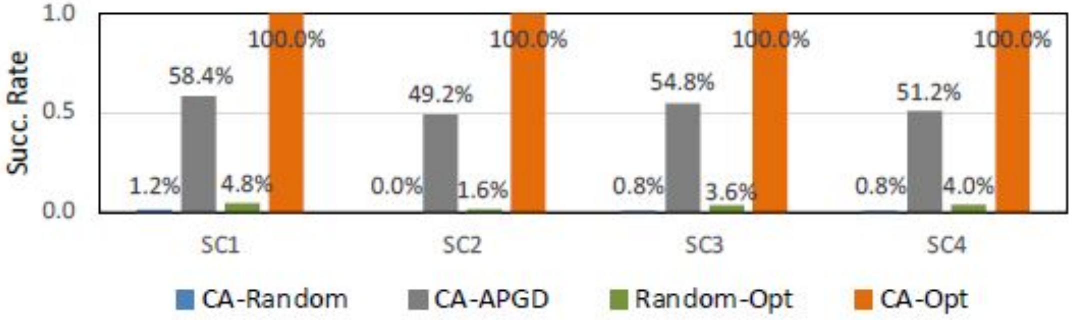


Figure 8: Control command dispatcher.

SIMULATION METHODS AND RESULTS

RQ1: Does strategic selection of attack times and values increase the chance of hazards (forward collisions)?
RQ2: Does stealthiness design help maintain the attack effectiveness in the presence of safety interventions?
RQ3: Does a perception input attack achieve better performance than direct perception and control output attacks?
Baselines: CA-Random and CA-APGD



Attack	Start Time	Duration	Attack Value	#Sim.
CA-Random	Context-Aware	Context-Aware	Random	1000
CA-APGD	Context-Aware	Context-Aware	AutoPGD	1000
Random-Opt	Uniform [5,40]s	Uniform [0.5,2.5]s	Opt-based	1000
CA-Opt (Ours)	Context-Aware	Context-Aware	Opt-based	1000

SIMULATION METHODS AND RESULTS

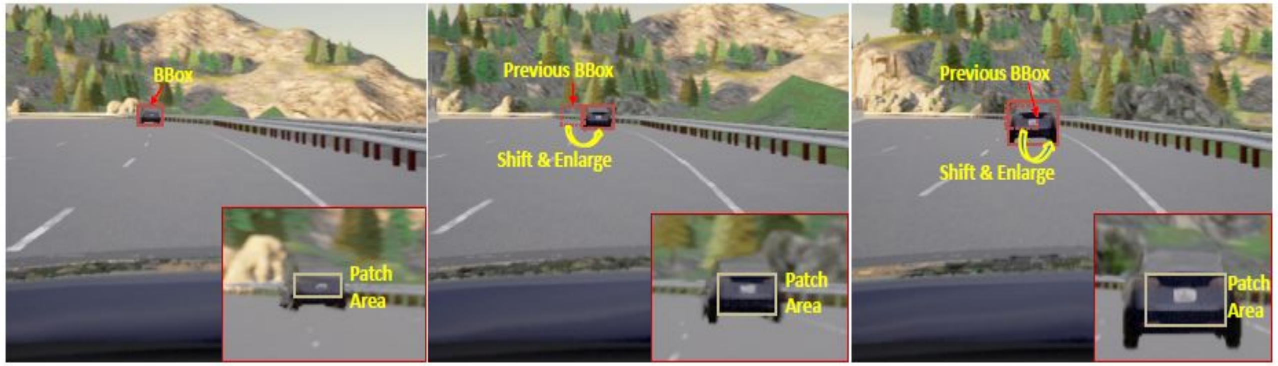


Figure 4: Examples of the shift and adjustment process in the patch generation. Inset figures are the zoomed-in views of the front vehicle with an adversarial patch added around the license plate area.

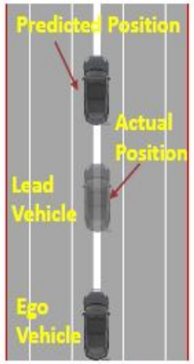


Table 6: Attack success rate with different patch stealthiness levels.

Stealthiness Level λ	Succ. Rate	Perturbation Pixel		Image Similarity	
		L_2	L_∞	RMSE($\times 10^{-5}$)	UIQ
10^{-2}	99.2%	0.086	0.015	1.061	0.993
10^{-3}	100%	0.128	0.015	1.168	0.993
10^{-4}	100%	0.184	0.015	1.319	0.993

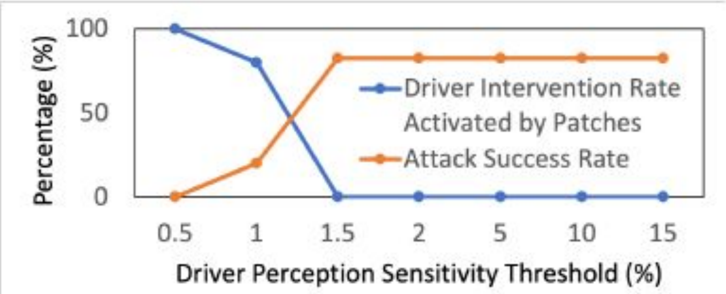
SIMULATION METHODS AND RESULTS

safety interventions are effective
in preventing accidents, and as required for L2 AVs, the human
driver should always be in the loop and actively monitor ADAS
to ensure safety.

CA-Opt attack is more effective than baselines
in keeping perturbations stealthy and causing hazards without
being mitigated by safety interventions.

Table 7: Performance of attacks with all the safety features and different AEBS settings.

Safety Interventions	Attack Method	Intervention Activation Rate	Succ. Rate	Hazard Prevention Rate
All & AEBS Not Compromised (Independent Camera)	CA-Random	27.4%	0	100% (7/7)
	CA-APGD	100%	0	100% (534/534)
	CA-Opt	100%	48.7%	51.3% (513/1000)
All & AEBS Disabled/ Compromised (Shared Camera)	CA-Random	23.8%/ 24.3%	0	100% (7/7)
	CA-APGD	100%	0	100% (534/534)
	CA-Opt	100%	82.6%	17.4% (174/1000)



SIMULATION METHODS AND RESULTS

Table 8: Performance of StrategicOut attack with all the safety features and different AEBS settings (AEBS with Shared Camera).

Safety Interventions	Attack Method	Succ. Rate	Hazard Prevention Rate
All & AEBS Activated	StrategicOut	20.3%	79.7% (797/1,000)
All & AEBS Disabled	StrategicOut	81.9%	18.1%(181/1,000)
All & AEBS Activated	OptOut	34.5%	65.5 (655/1,000)

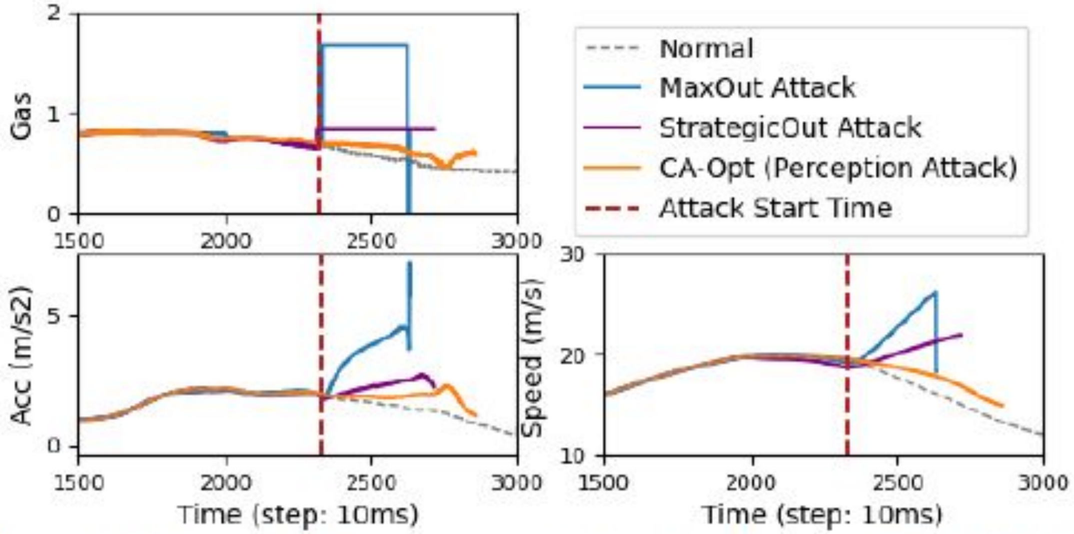


Figure 11: Context-Aware perception attacks vs. output attacks.

REAL WORLD EVALUATION

RQ4: Can our attack transfer well from simulation to real-world implementation?
RQ5: Can our attack evade detection or mitigation by the existing adversarial patch defense methods?

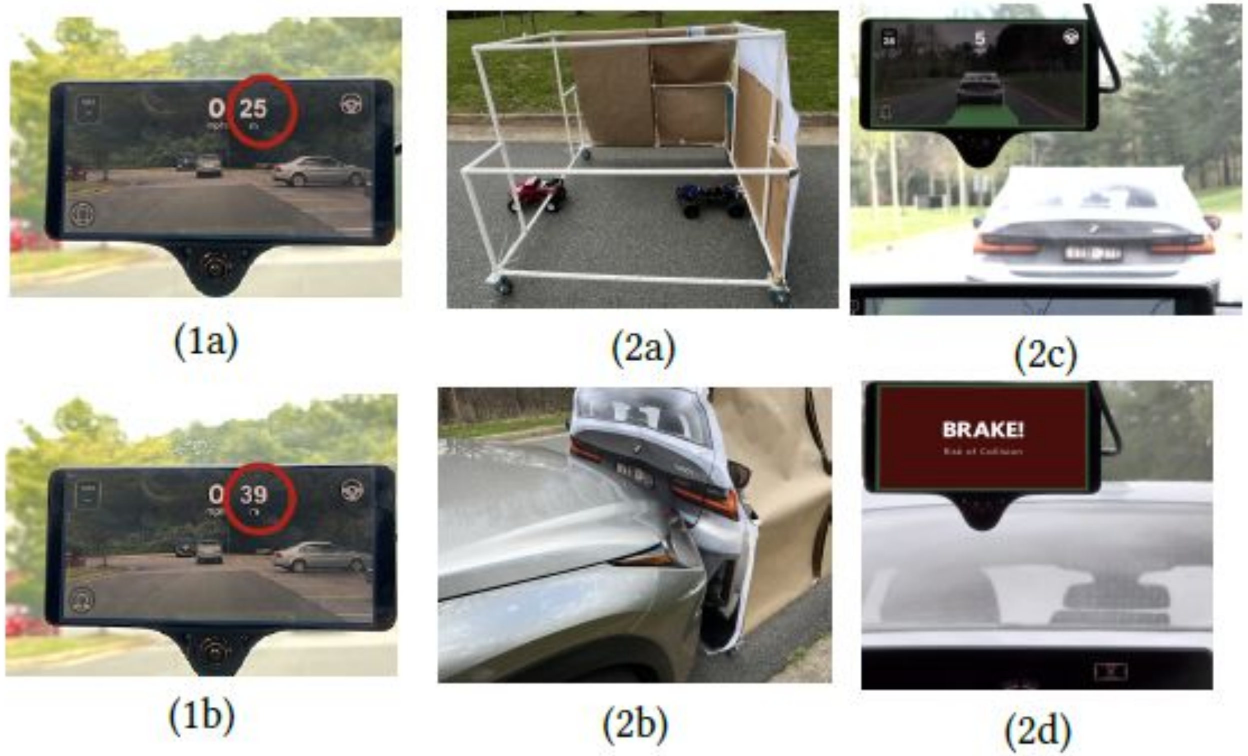


Figure 14: RD predictions w/o (1a) or w (1b) patch on an actual vehicle in a parking lot; (2a) Side view of lead car model; (2b) AV under perception attack collides with the lead car model; (2c) AV follows the car model in a benign scenario; (2d) Driver's view upon collision.

DISCUSSION

According to the U.S. Department of Transportation (DOT), over 42,000 crashes occur in work zones annually, with more than 800 fatalities reported in 2021 alone, a significant portion involving rear-end collisions and large trucks. These are often due to quick lane changes, reduced visibility, and sudden braking—conditions that confuse both human drivers and autonomous systems.

How might these real-world factors in construction zones increase the success rate or stealthiness of such an attack?

Would AVs be safer than humans in this context, or could their dependence on visual DNNs make them even more vulnerable in construction settings?

What elements would you need to include in a simulation or field test to accurately capture these risks (e.g., driver reaction delay, AEBS response)?