Securing Autonomous Vehicles Under Partial-Information Attacks

Spencer Hallyburton and Miroslav Pajic

Cyber-Physical Systems Lab (CPSL)

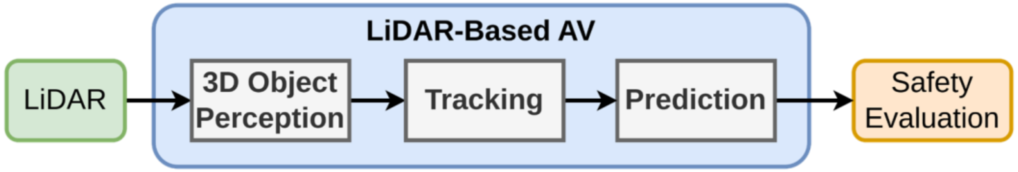
Department of Electrical and Computer Engineering
Pratt School of Engineering
Duke University

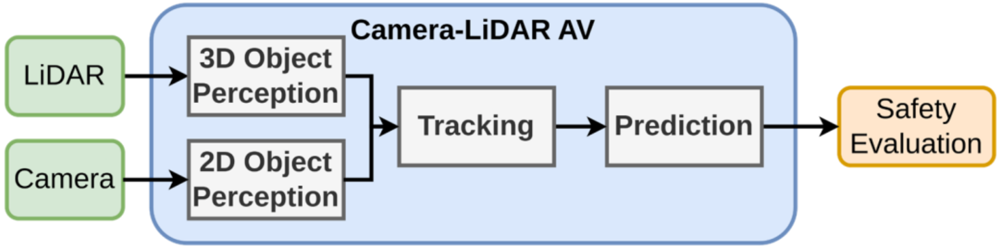
Duke PRATT SCHOOL of ENGINEERING

DUKE ROBOTICS

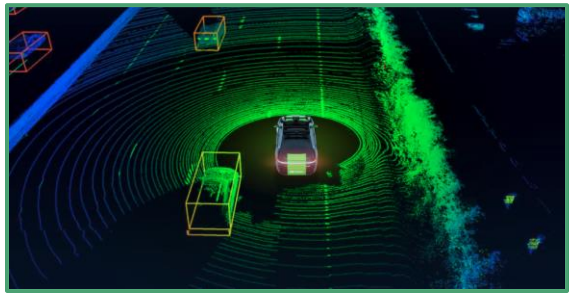
Sensor fusion in autonomous vehicles (AVs)

- Sensors including: LiDAR, camera, radar
- Knowledge of objects in scene
- Prediction of object motion
- Maintaining ego-vehicle safety
- Building situational awareness

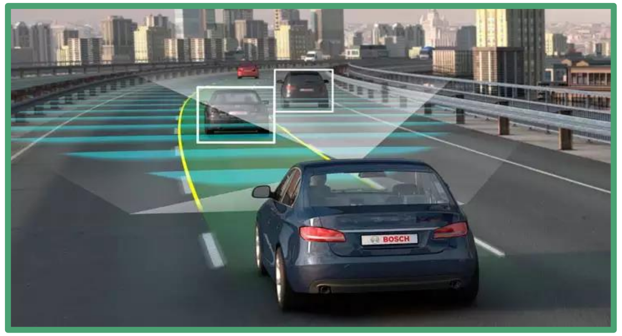


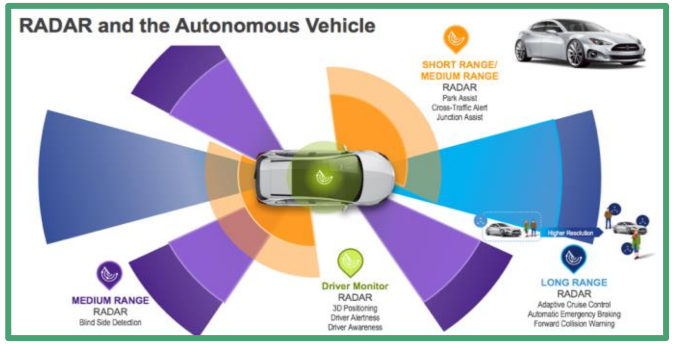


LiDAR provides 3D point cloud



Camera provides dense 2D image

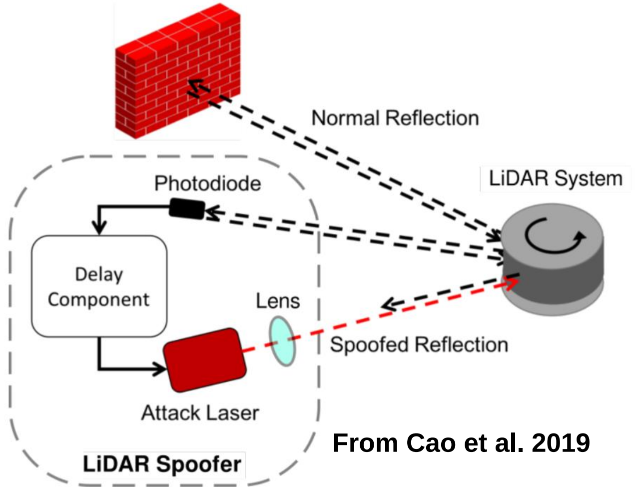




Radar provides sparse position, doppler

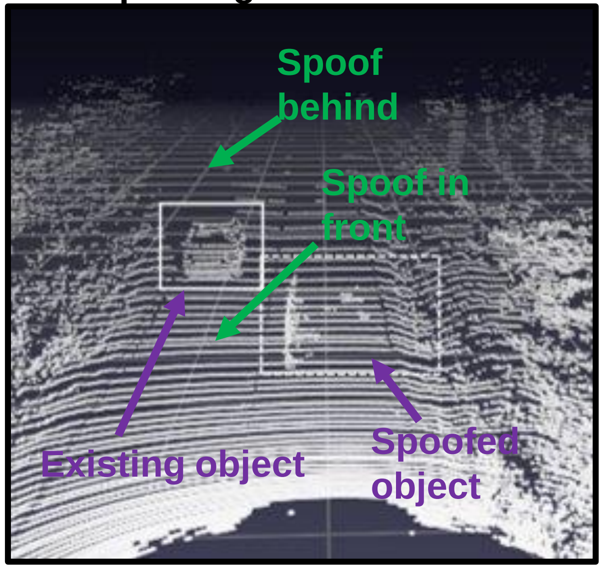
Images from:
https://semiengineering.com/here-comes-high-res-car-radar/
https://roboticsandautomationnews.com/2021/01/29/lidar-sensor-makers-choose-nvidia-drive-for-development/40052/
https://auto.economictimes.indiatimes.com/news/auto-technology/cheaper-infrared-cameras-for-self-driving-cars-phones-in-the-offing/68340483

Recent security analysis:
Structured spoofing and injection attacks



From Cao et al. 2019

Spoofing Attacks at 8m



Threat Model

Attack Model
Road-side attack laser, photodiode

Attacker Knowledge
Line-of-sight to victim to receive and transmit signal

Attacker Capability
Up to 200 spoof points

Challenge: Expensive hardware

Challenge: Moving vehicles

Challenge: Precise aiming, timing

Attack Designs

Naïve Attack
Spoofing in front-near position of victim without contextual information

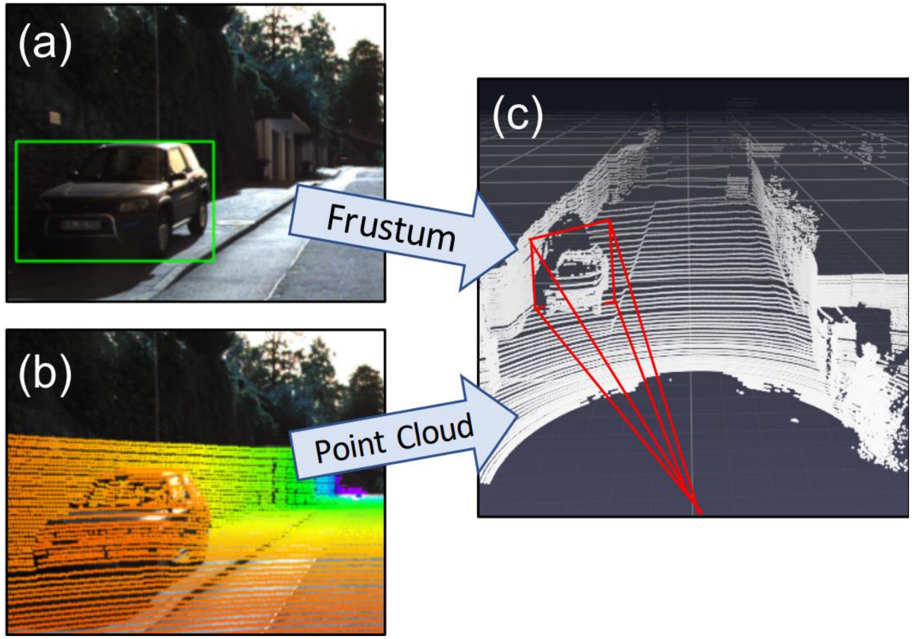
Frustum Attack
Spoofing relative to a "target car" -- in front or behind, relative to victim

Cao, Y., Xiao, C., Cyr, B., Zhou, Y., Park, W., Rampazzi, S., ... & Mao, Z. M. (2019, November). Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM CCS*

Sun, J. S., Cao, Y. C., Chen, Q. A., & Mao, Z. M. (2020, January). Towards robust lidar-based perception in autonomous driving: General black-box adversarial sensor attack and countermeasures. In *USENIX Security Symposium (Usenix Security'20)*.

Hallyburton, R. S., Liu, Y., Cao, Y., Mao, Z. M., & **Pajic, M**. (2022). Security Analysis of {Camera-LiDAR} Fusion Against {Black-Box} Attacks on Autonomous Vehicles. In *31st USENIX Security Symposium (USENIX Security 22)* (pp. 1903-1920).

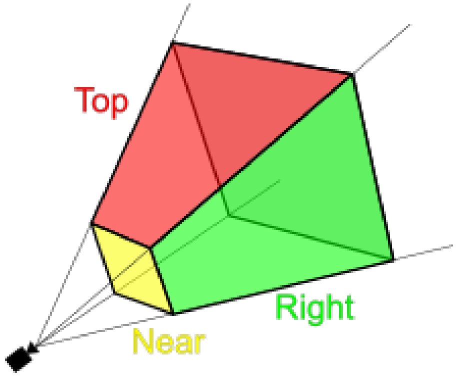
Compromise sensor fusion with “frustum” attack



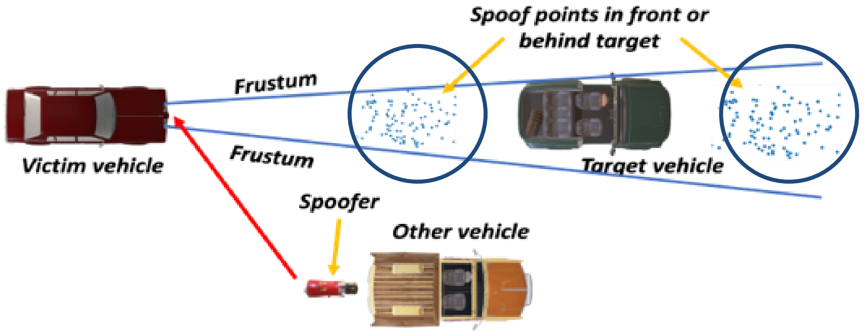
Frustum Vulnerability
3D space <u>in front or behind</u> an existing "target vehicle" is consistent with unaltered 2D image

Shadow Vulnerability
Real 3D objects create a void region of space <u>behind</u> them where no LiDAR points exist

Frustum Definition
2D image unable to resolve range information – leads to 3D "frustum" extruded along range axis



Viewing frustum defined by a camera field-of-view.



Configuration for frustum attack. Adversary spoofs in front or behind target object.

<u>**Hallyburton, R. S.,**</u> Liu, Y., Cao, Y., Mao, Z. M., & <u>**Pajic, M**</u>. (2022). Security Analysis of {Camera-LiDAR} Fusion Against {Black-Box} Attacks on Autonomous Vehicles. In *31st USENIX Security Symposium (USENIX Security 22)* (pp. 1903-1920).

Partial-Information Attacks on LiDAR

Securing Autonomous Vehicles Under Partial-Information Attacks

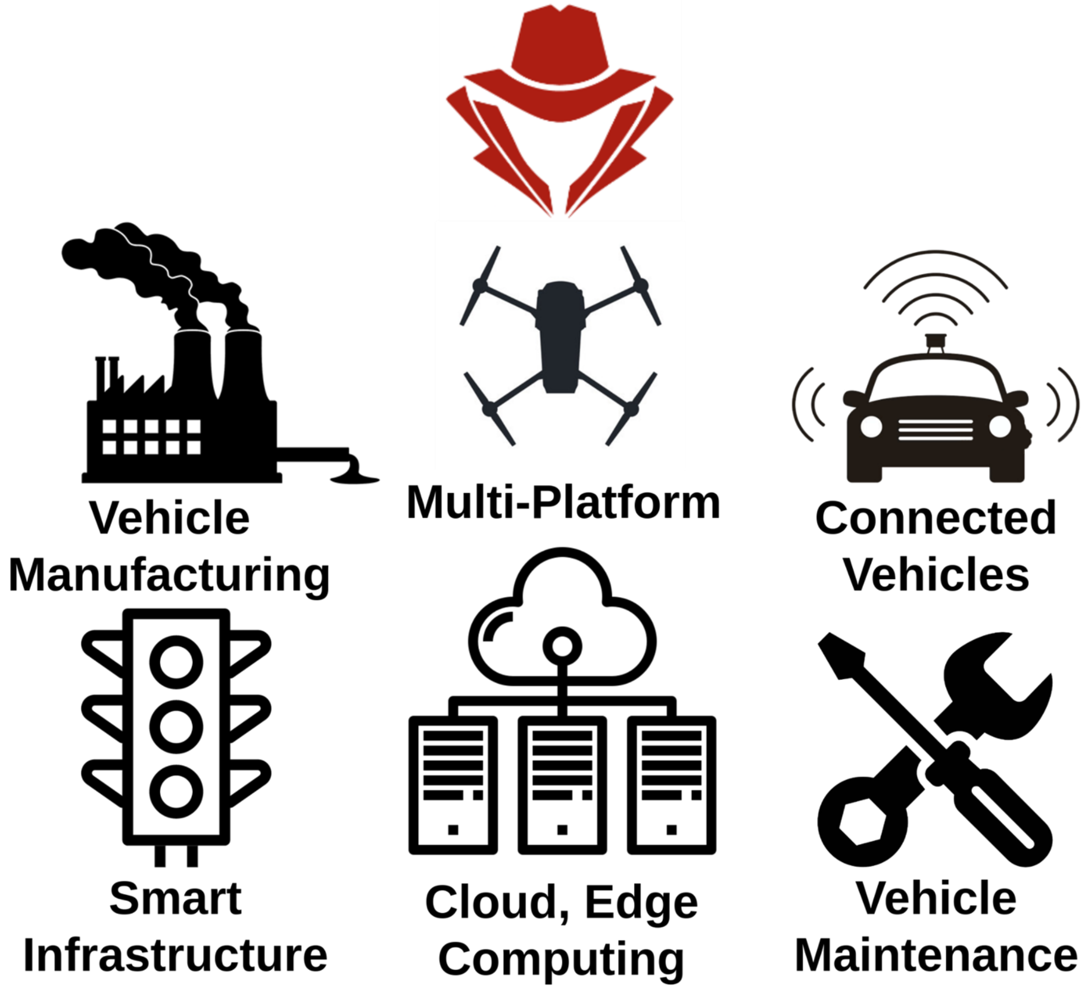
Cyber threats are increasingly likely

Attackers are more ambitious than ever

Connected vehicles, edge computing makes CPS vulnerable

AVs are vulnerable to many attack vectors

Remote attacks on AVs already demonstrated



Cyber attack threat model

<u>Threat Model</u>

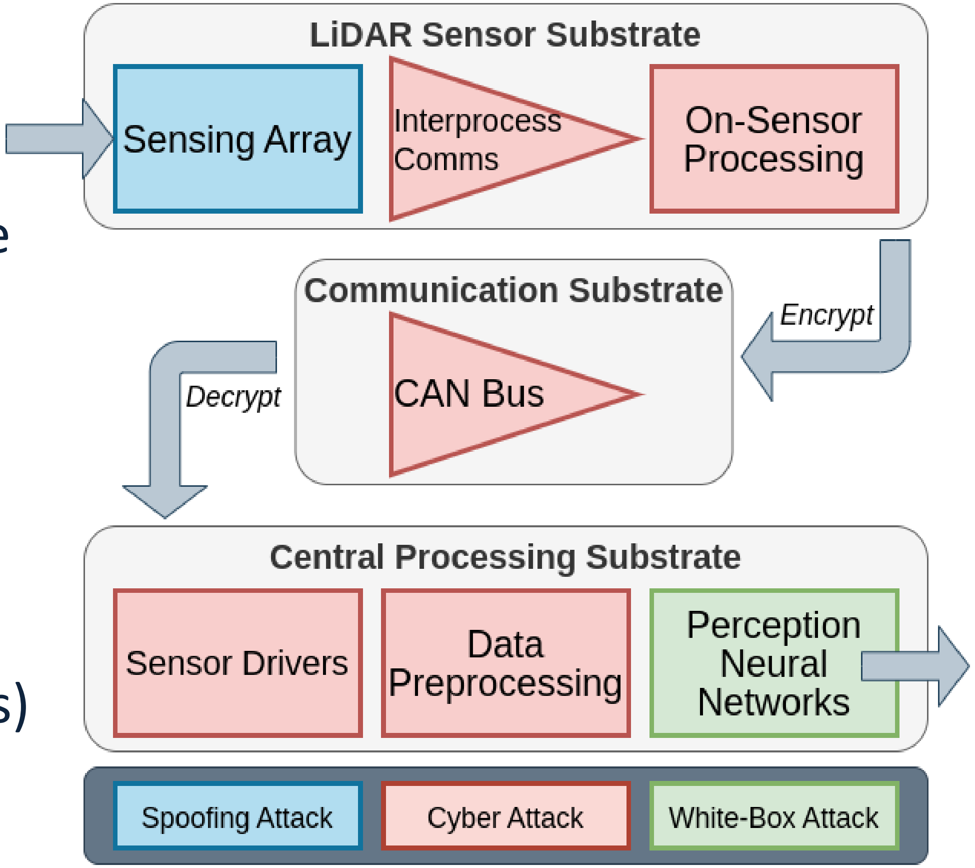
- Compromised sensor (e.g., LiDAR sensor)
- Cyber threat at sensor, comms, or processing substrate

<u>Knowledge Model</u>

- **Limited a-priori information**
- **Only access to raw data at sensor level**

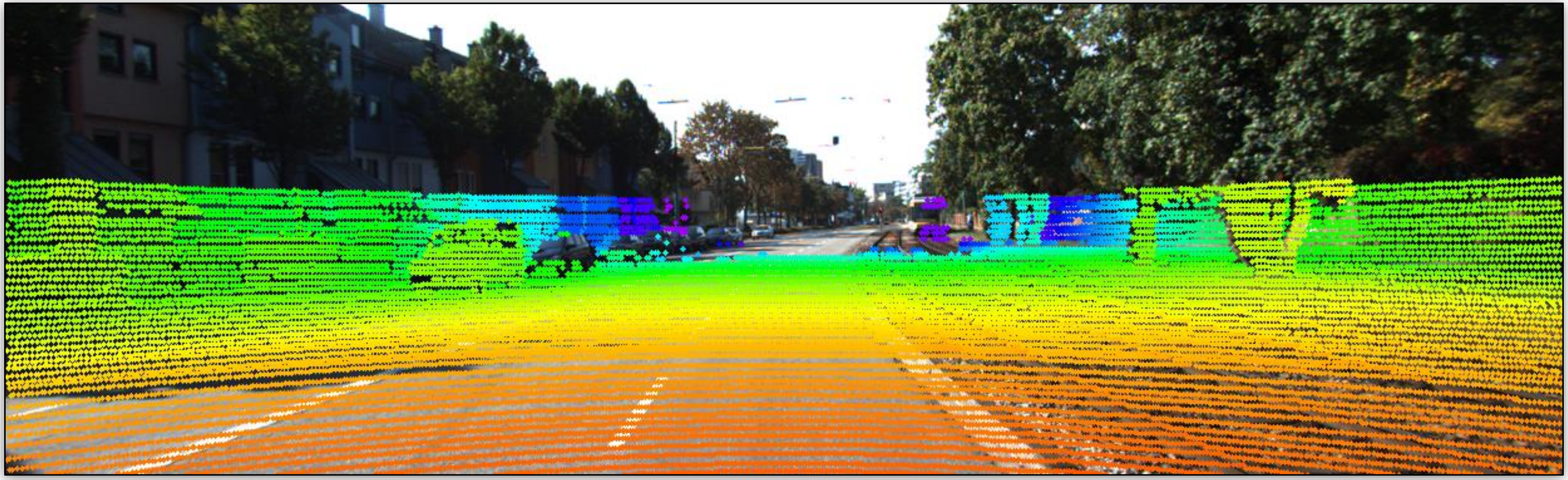
<u>Attacker Capabilities</u>

- Attacker has access to the sensor data (spherical points)
- Range modification → attacker can modify only the range of the points due to LiDAR data structure
- Range nullification → attacker can set range value of points to NULL
- Add/drop LiDAR datagrams
- Attacker <u>cannot</u> modify point angles



Any **of the red elements could be compromised for a cyber attack to be effective**

Understanding the LiDAR point cloud



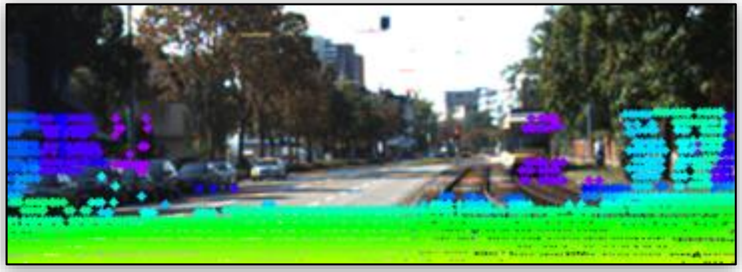
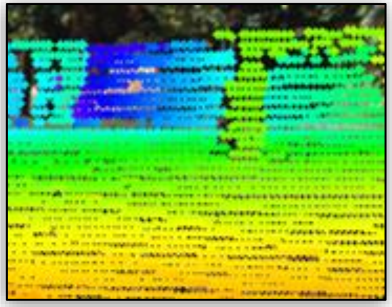
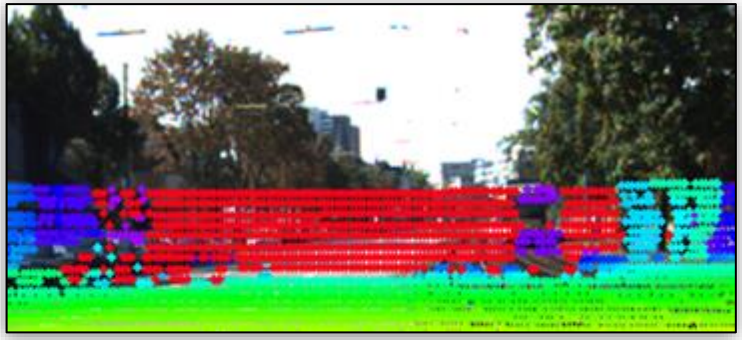
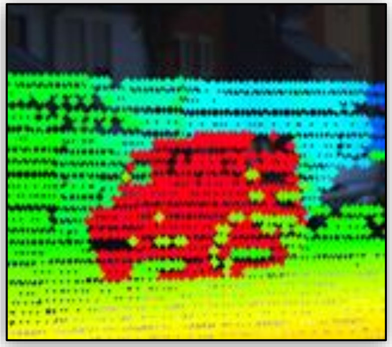
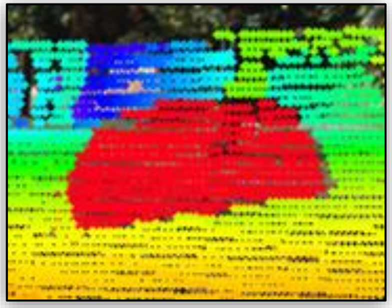
Point cloud projected onto image for visualization purposes

Each point is a 3D return from a laser

Color corresponds to range of the point (distance)

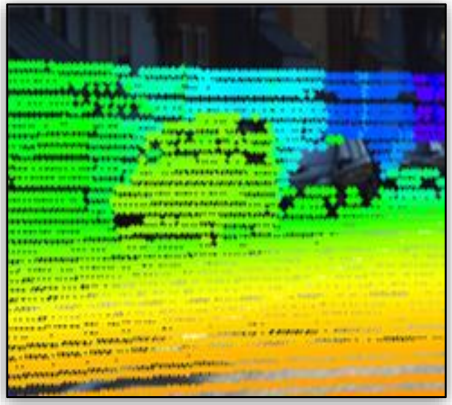
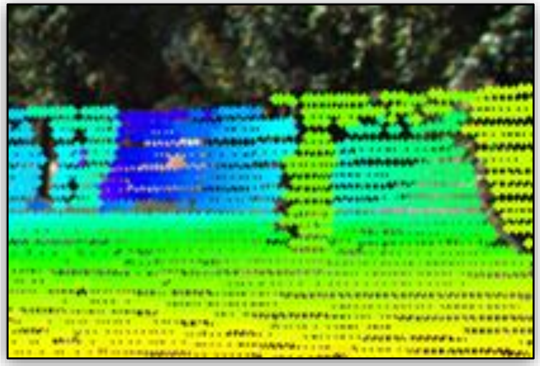
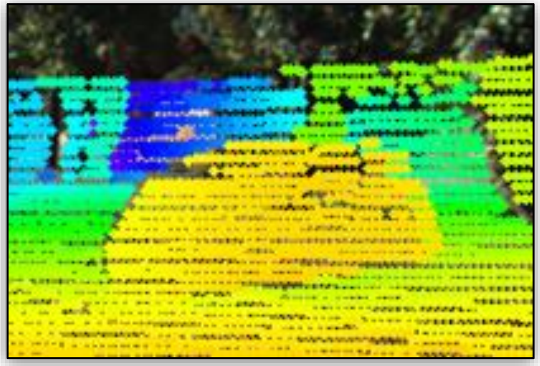
LiDAR has 64 vertical (elevation) channels and many horizontal (azimuth)

Attacker subroutines – “masking”

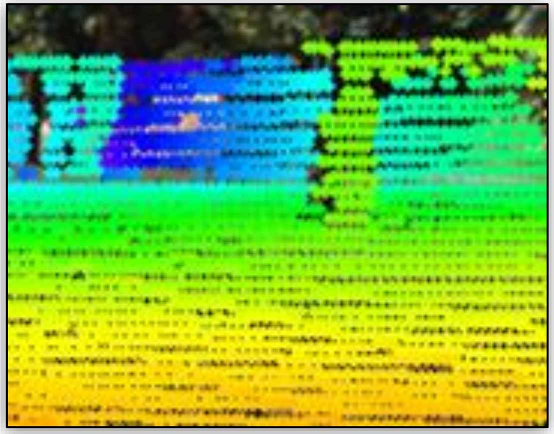
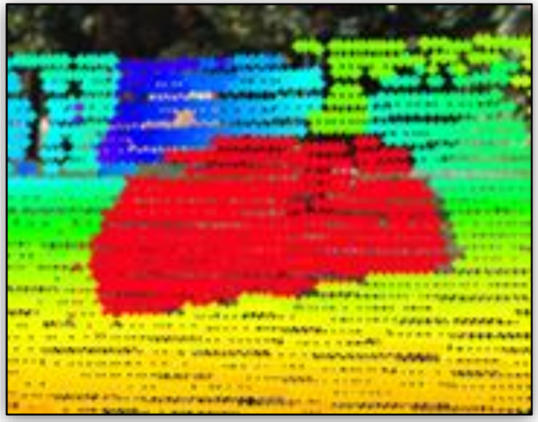
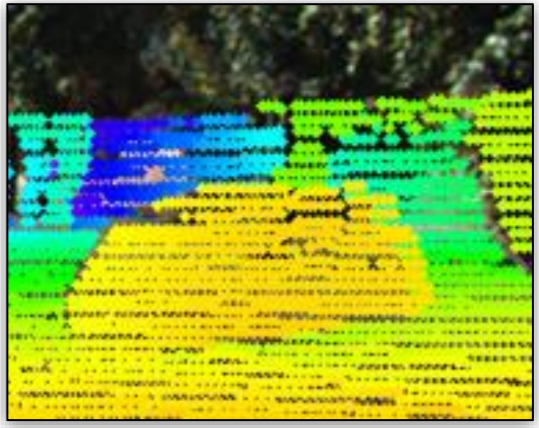
	Mask missing angles	Mask object	Mask trace
Original Point Cloud			
Masked Point Cloud			
	Find angles in the point cloud matrix that originally returned “NULL”	*Mask points pertaining to an existing object*	*Mask points that will be affected by inserting a new “trace”*

***Color overloaded → red means “1” and all others “0” for a binary mask*

Attacker subroutines – “inpainting”

	Inpaint mask as background from context	Inpaint mask as object from trace
Original Point Cloud		
Inpainted Point Cloud		
	Given mask, change ranges to make masked region appear like background	*Given mask, change ranges to make masked region appear like object*

Example: False Positive attack

	Mask trace	Inpaint mask as object from trace
		
Original Point Cloud	*Find Points to Manipulate*	*Manipulate points to look like object*

***Color overloaded → red means "1" and all others "0" for a binary mask*

Building cyber-based attacks

- Attacks built from previous subroutines
- Context-aware: attacker builds awareness in real time
- Attacker only needs to wait for "right moment" to attack.
- Attacks: false positive, replay, object removal

Context Unaware

Context Aware

TABLE II: Attack executions are constructed from subroutines. Frustum-type attacks use other attacks as subroutines.

Num.	Att. Case Name	Subroutines

ATT.1	False Positive	`FindMissingAngles`
		`GetPointMaskFromTrace`
		`InpaintMaskAsObjectFromTrace`

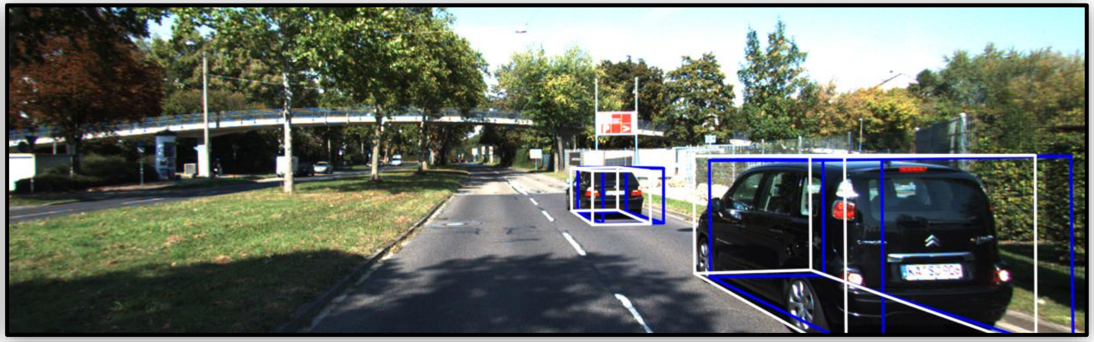
ATT.2	Dual False Positive	`FindMissingAngles`
		`GetPointMaskFromTrace`
		`InpaintMaskAsObjectFromTrace`

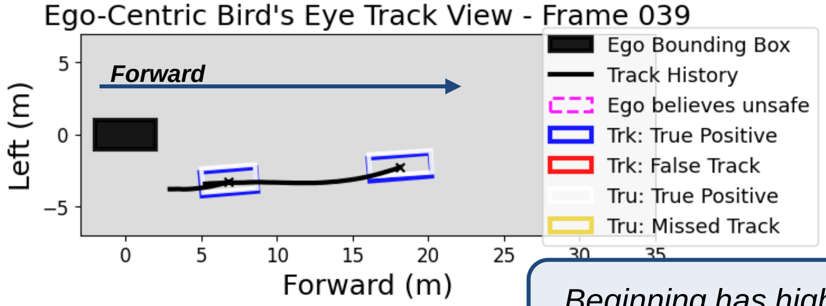
ATT.3	Forward Replay	N/A
ATT.4	Reverse Replay	N/A
ATT.5	Clean Scene	`InpaintMaskAsBackgroundFromContext`

ATT.6	Object Removal	Object Detection, Tracking
		`GetPointMaskFromObject`
		`InpaintMaskAsBackgroundFromContext`

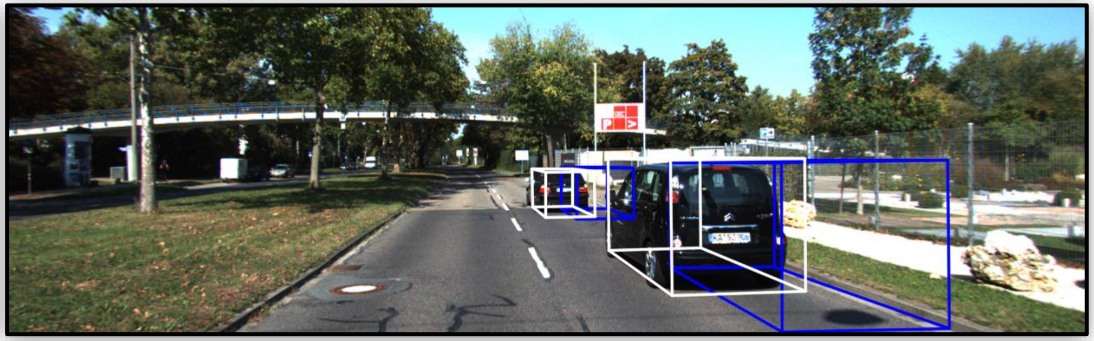
ATT.7	Frustum Translation	Object Removal False Positive
ATT.8	Dual Frustum False Positive	Object Removal False Positive

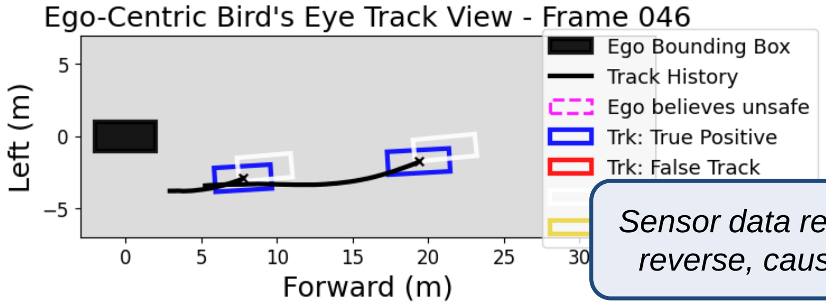
Case study: Reverse Replay attack



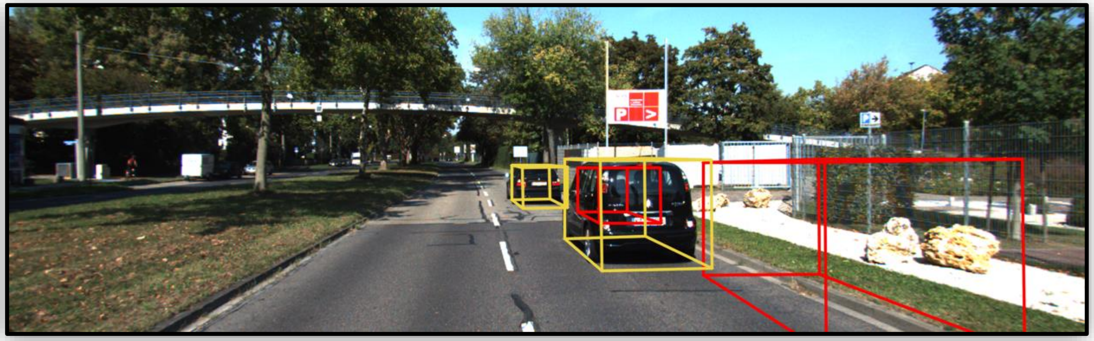


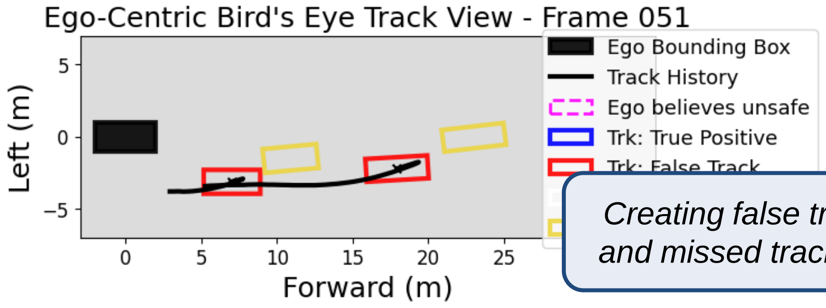
Beginning has high-accuracy situational awareness





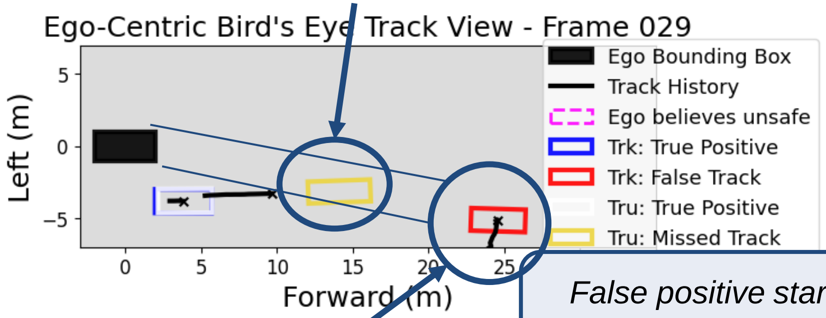
Sensor data replayed in reverse, causing drift





Creating false tracks (red) and missed tracks (yellow)

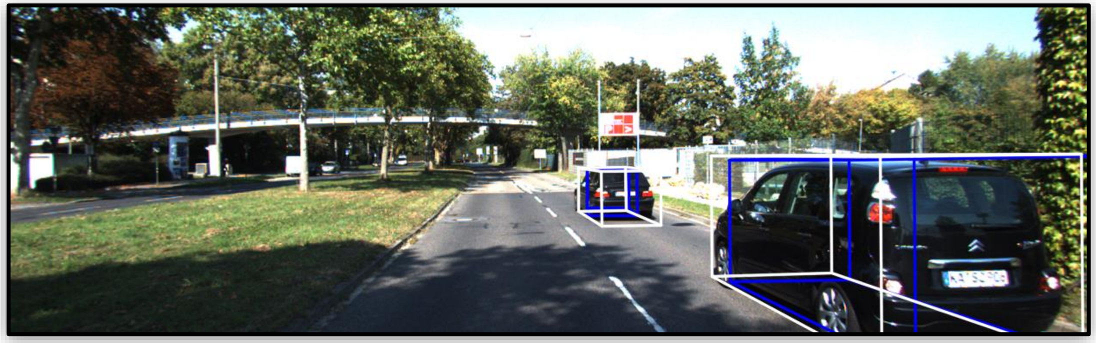
Case study: Frustum Translation attack

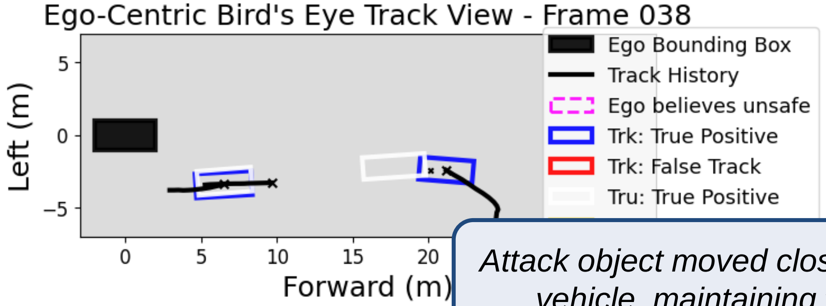


Real object "in line"

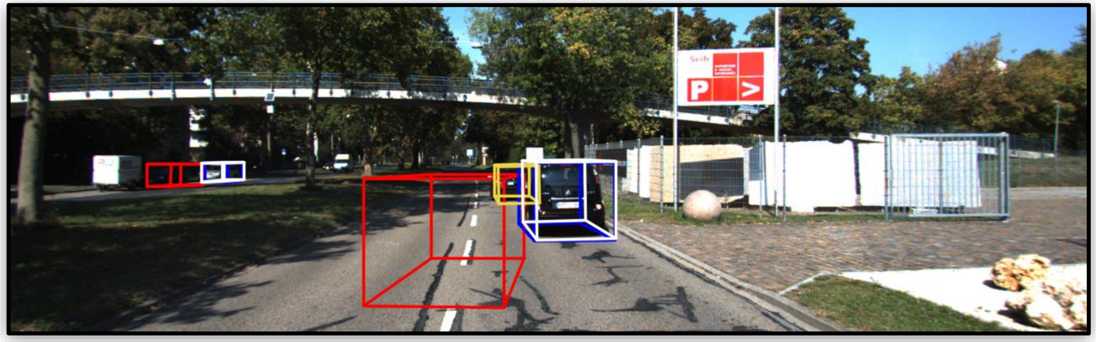
Fake object

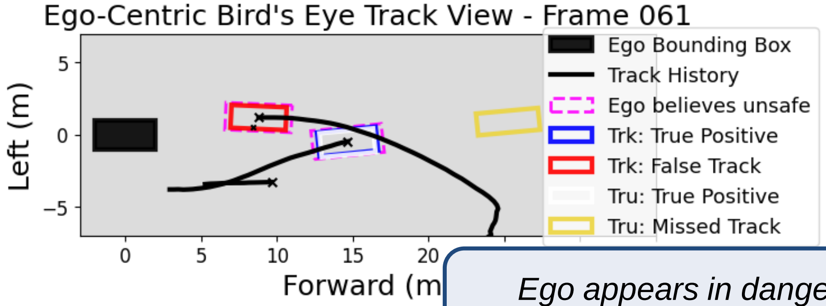
False positive starts in-line with existing object "frustum"





Attack object moved closer to victim vehicle, maintaining frustum

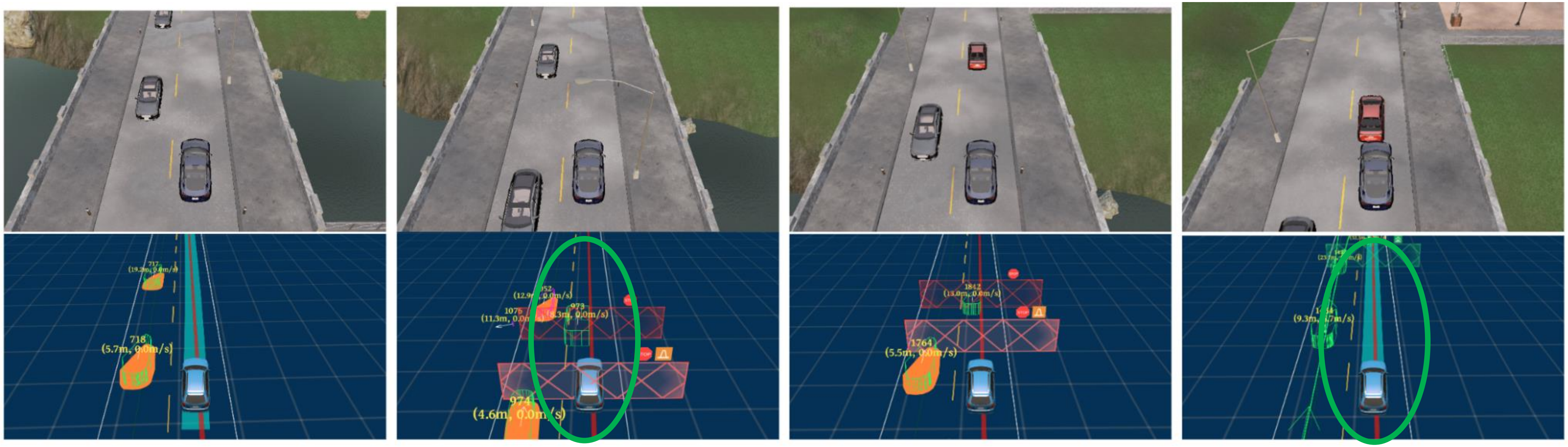




Ego appears in danger of "colliding" with attack object

Demonstrated success on industry-grade AV

Tested on Baidu's Apollo - Level 5, fully autonomous self-driving vehicle



(a) Unattacked. Ego sees lane is clear and plans straight path.

(b) FP attack ATT.1. Ego emergency brakes to avoid fake obj.

(c) Unattacked. Ego stops ahead of existing stopped car

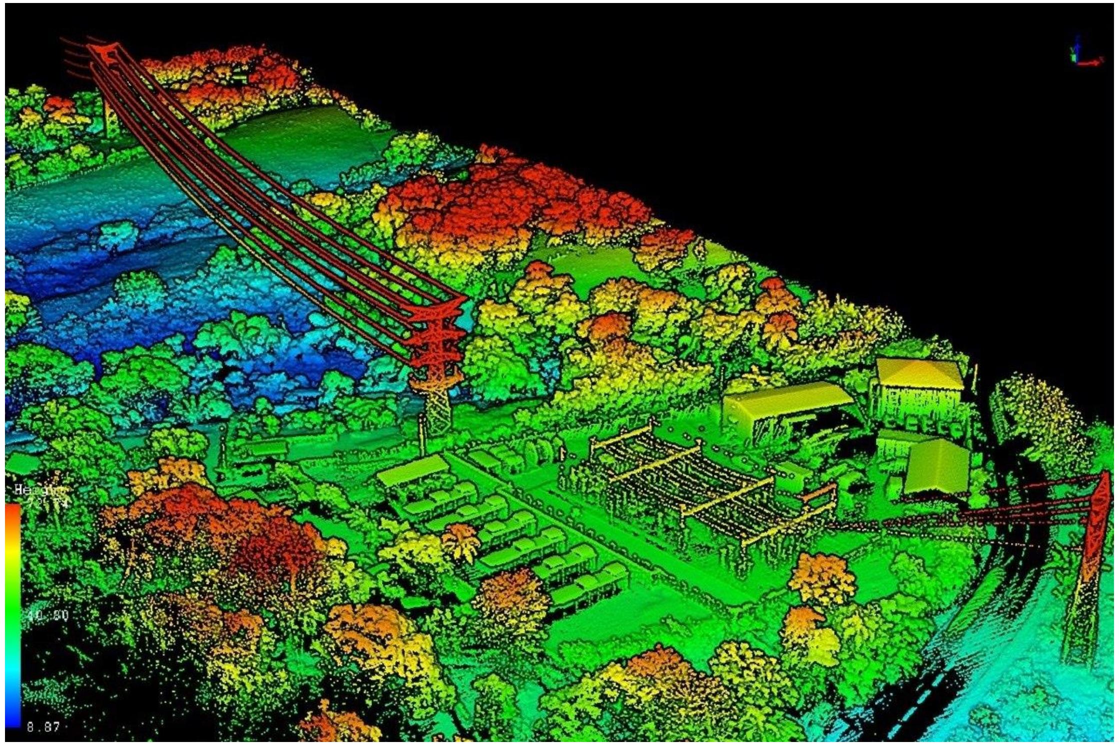
(d) Rev-replay attack ATT.4. Ego crashes into stopped car.

Context-unaware false positive attack

Context-unaware reverse replay attack

Extending attacks into aerial domain

- Attacks are data-source agnostic
- Attacks are platform agnostic
- Moving analysis to AirSim simulator



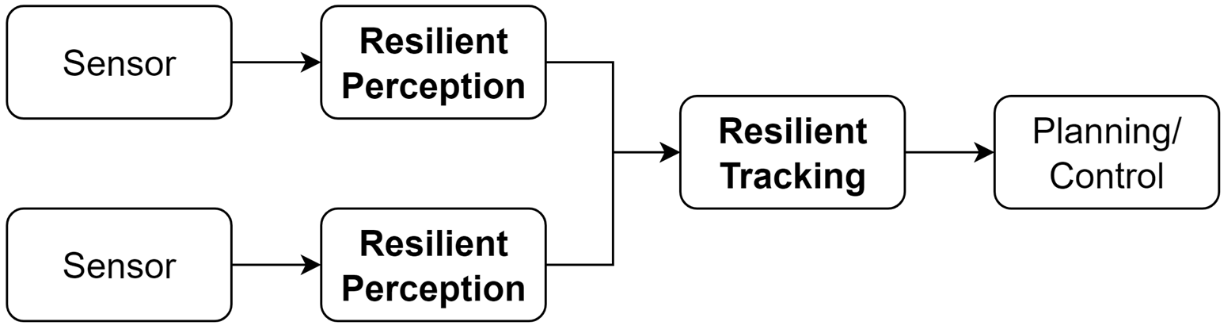
Security-Aware Sensor Fusion

Securing Autonomous Vehicles Under Partial-Information Attacks

Security-aware philosophies

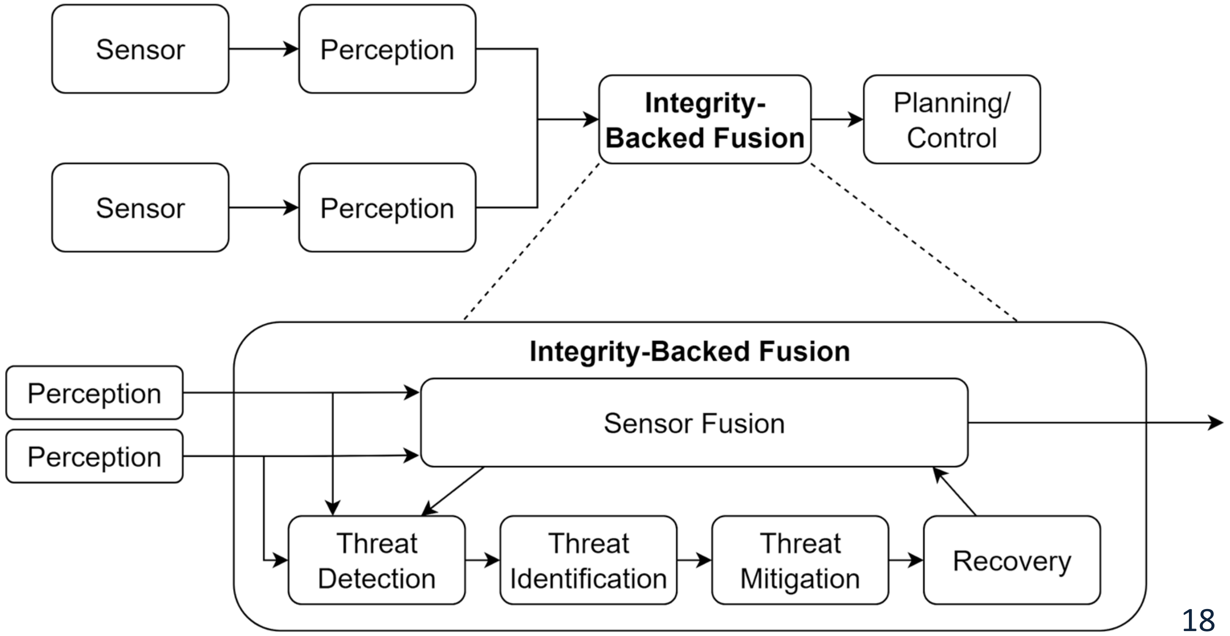
Challenges:

- Degradation in nominal performance
- Complexity of implementation
- Rigidity of estimation structure



Integrity (Detect + Identify + Mitigate + Recover)

- An understanding that some attacks will succeed
- Questions becomes:
  - How do we detect and identify?
  - How do we mitigate and recover?



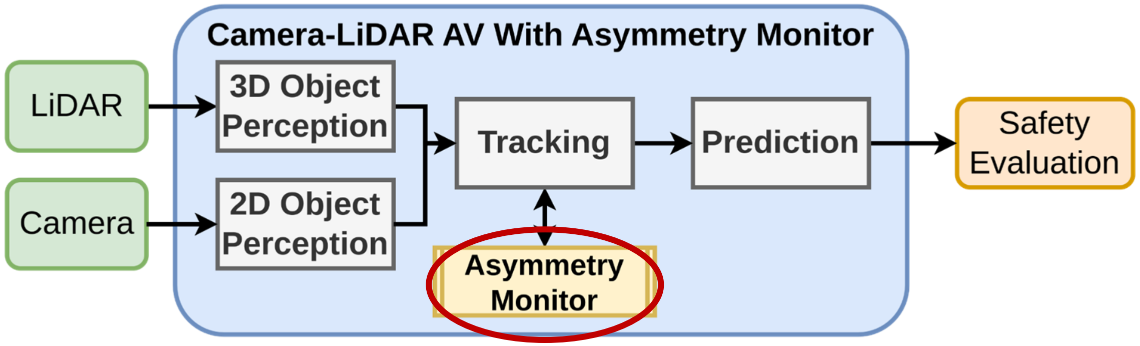
Two approaches to detect and mitigate attacks

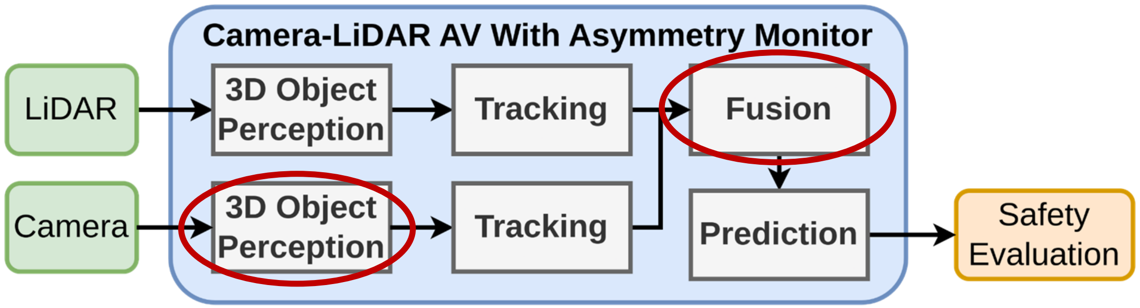
Monitoring data asymmetries

- Centralized object tracking
- Maintaining sensor-specific “scores”
- Scores derived from likelihood ratios

Distributed tracking and fusion

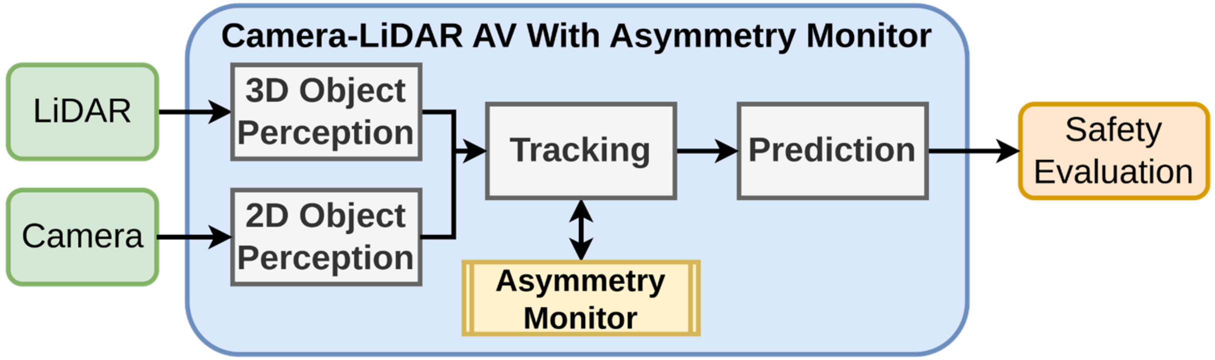
- 3D monocular camera detection
- Distributed object tracking
- Post-tracking fusion





Monitoring for data asymmetries

- Traditional tracking maintains "score"
  - Log likelihood ratio of:
    - H1 → track is a true target
    - H0 → track is a false alarm



$$\mathrm{LR} = \frac{\Pr(D|H_1)\,\Pr_0(H_1)}{\Pr(D|H_0)\,\Pr_0(H_0)} := \frac{P_T}{P_F}$$

True target hypothesis

False alarm hypothesis

$$LLR := L = \log \frac{P_T}{P_F}.$$

Track score

- For <u>n sensors</u>, maintain <u>n+1 scores</u>
  - n per-sensor scores
  - 1 central score (all sensors, same as done in traditional tracking)
  - Central score for track "confirmation"
  - Per-sensor scores for track "validation"

Central track score

User-selected threshold

$$\text{Confirmation} = \begin{cases} \texttt{Confirmed} & L_{k,\text{central}} \geq \tau_c \\ \texttt{Unconfirmed} & \text{otherwise} \end{cases}$$

$$\text{Validation} = \begin{cases} \texttt{Valid} & \begin{array}{l}\texttt{Confirmed}\text{ and} \\ |L_{k,\text{sensor}_1} - L_{k,\text{sensor}_2}| < \tau_v\end{array} \\ \texttt{Invalid} & \text{otherwise} \end{cases}$$

Per-sensor track score

User-selected threshold

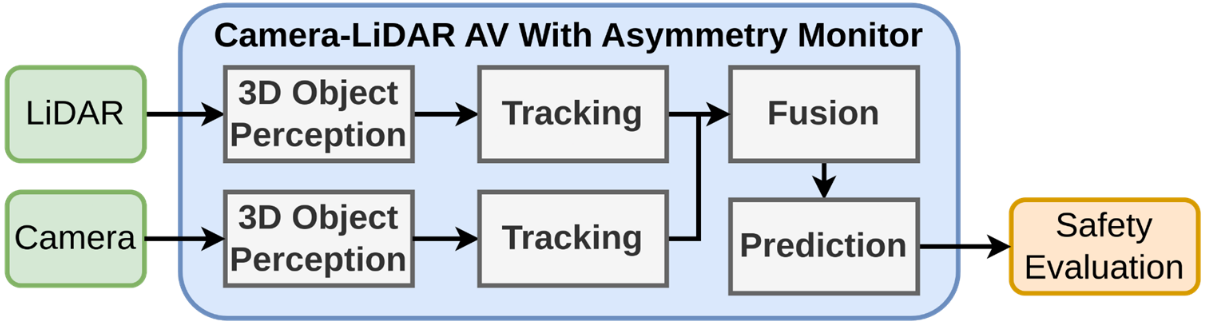
Extending 2d data to 3d with scene context

Monocular Detection

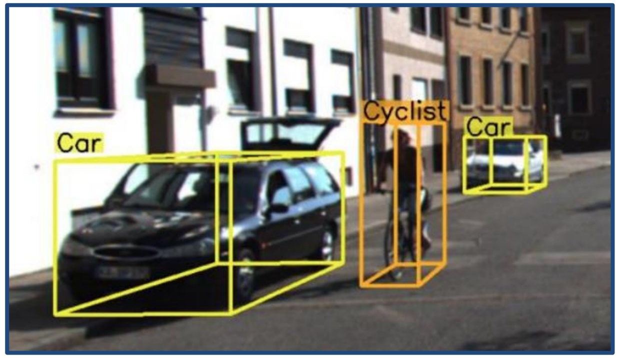
- Motivation: 2D detections from camera are ambiguous when extended into 3D
- Solution: detect 3D objects from 2D images using context directly
- Algorithm: PGD, M3D-RPN

Post-Tracking Fusion

- Motivation: uncompromised sensor compensates for inconsistent dynamics of compensated sensor
- Solution: perform tracking on each sensor and fusion after tracking
- Algorithm: Distributed data fusion (e.g., covariance intersection, conservative Kalman filtering)

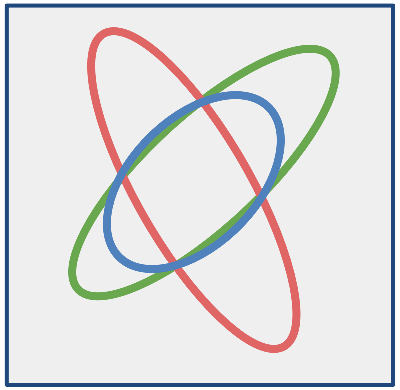


Monocular Detection



Monocular detection extends object detection from 2D data to 3D detections using context and optimization

Covariance Intersection (CI) Fusion



*CI fuses two data sources (**red**, **green**) conservatively to reduce uncertainty of estimate (**blue**). CI useful when data correlations unknown (e.g., same platform)*

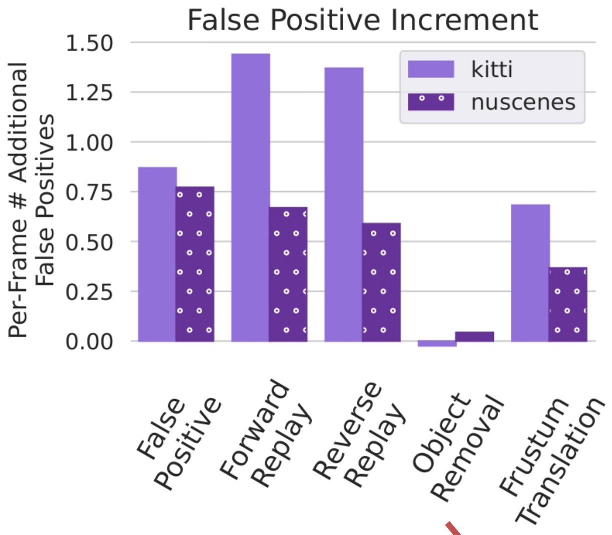
Brazil, G., & Liu, X. (2019). M3d-rpn: Monocular 3d region proposal network for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 9287-9296).

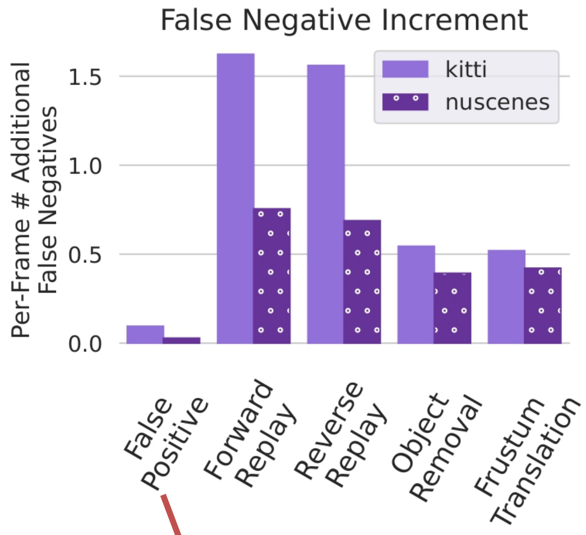
Outcomes at perception

- All AVs tested use same LiDAR perception algorithm
  - Therefore, outcomes at perception are identical for AVs
  - We show difference between attacks

- Metric → "Increment over baseline"
  - (1) run baseline AV
  - (2) run attack on AV
  - (3) compute difference

Attacks successful in creating false positives and false negatives



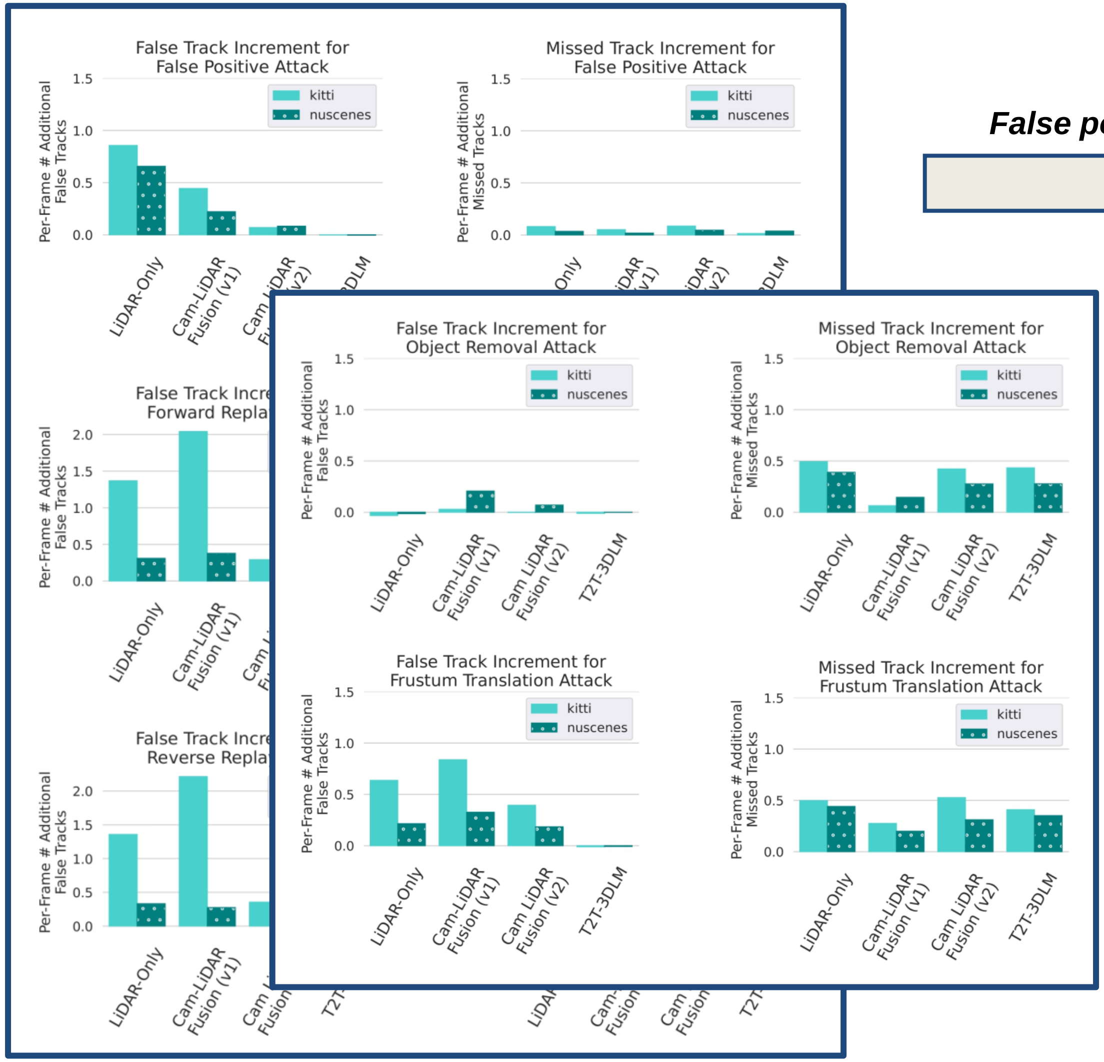


Object removal does not introduce new object

False positive does not remove existing objects

y-axis means: *when we run the attack, we obtain Y more of the metric than we did in the baseline case in each frame of the attack*

Outcomes at tracking



False positive attack

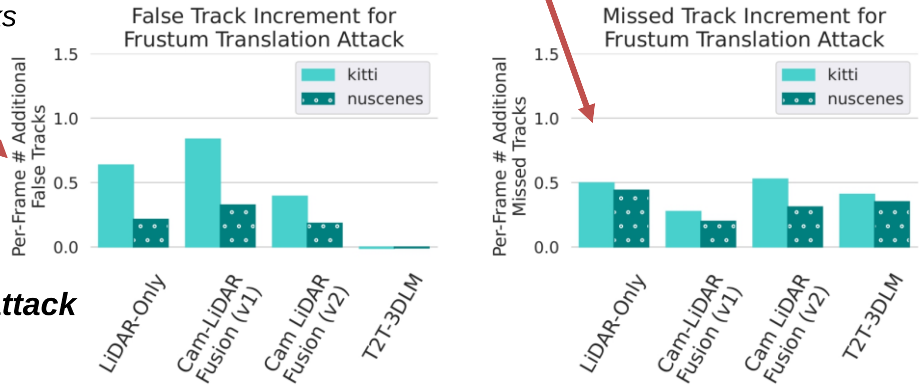
Reduced false tracks for security-aware

Few missed tracks for all

Security-aware fusion defends false positive attacks

Reduced false tracks for security-aware

Similar missed tracks across all



Frustum translation attack

Data asymmetry monitor vulnerable to frustum attack, T2T-3DLM improves performance

In conclusion

PRATT SCHOOL *of* ENGINEERING

- Limited-information cyber attacker can disrupt LiDAR-based AVs
- Attacker gains necessary situational awareness online
- Attacks are successful in many scenarios: KITTI, nuScenes, Apollo
- Basic security-aware architectures can improve assuredness

Thank you

Backup

Securing Autonomous Vehicles Under Partial-Information Attacks

Recent interest in security of AVs

Safety has received much of the attention…but what if data are *adversarially compromised?*

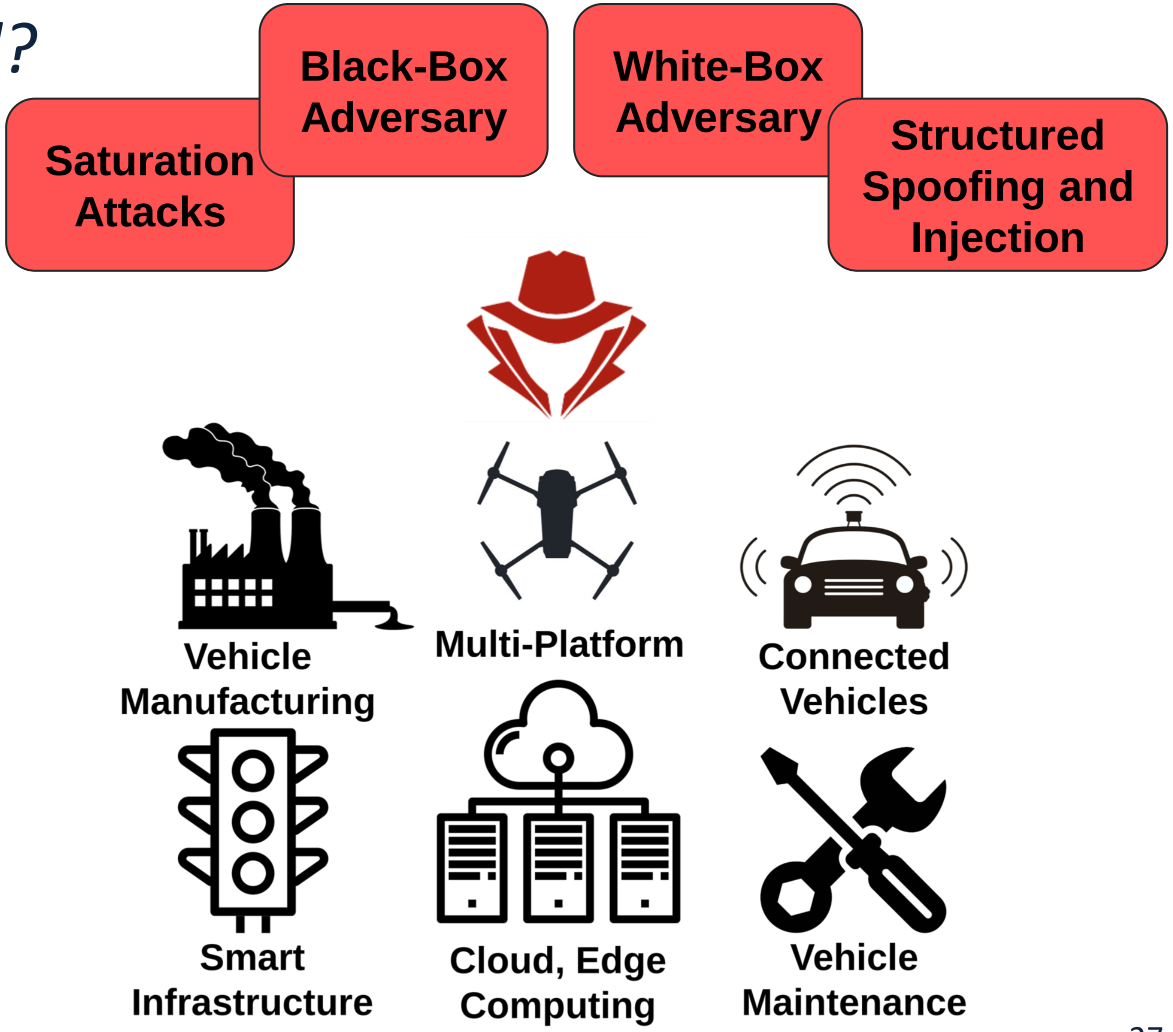
- Remote attacks on AVs
  Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., Savage, S., ... & Kohno, T. (2011, August). Comprehensive experimental analyses of automotive attack surfaces. In *USENIX security symposium* (Vol. 4, No. 447-462, p. 2021).

- Physical attacks
  Cao, Y., Xiao, C., Cyr, B., Zhou, Y., Park, W., Rampazzi, S., ... & Mao, Z. M. (2019, November). Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM CCS* (pp. 2267-2281).

  Hallyburton, R. S., Liu, Y., Cao, Y., Mao, Z. M., & Pajic, M. (2022). Security Analysis of {Camera-LiDAR} Fusion Against {Black-Box} Attacks on Autonomous Vehicles. In *31st USENIX Security Symposium (USENIX Security 22)* (pp. 1903-1920).

- White-box attacks
  Tu, J., Ren, M., Manivasagam, S., Liang, M., Yang, B., Du, R., ... & Urtasun, R. (2020). Physically realizable adversarial examples for lidar object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 13716-13725).

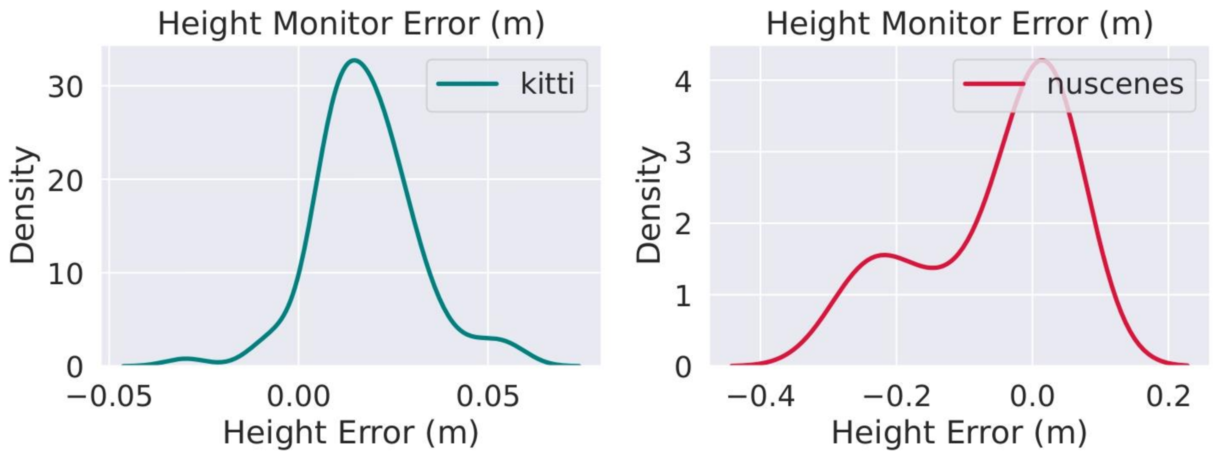
- Cyber attacks
  Hallyburton, R. S., & Pajic, M. (2023). Securing Autonomous Vehicles Under Partial-Information Cyber Attacks on LiDAR Data. *arXiv preprint arXiv:2303.03470*.

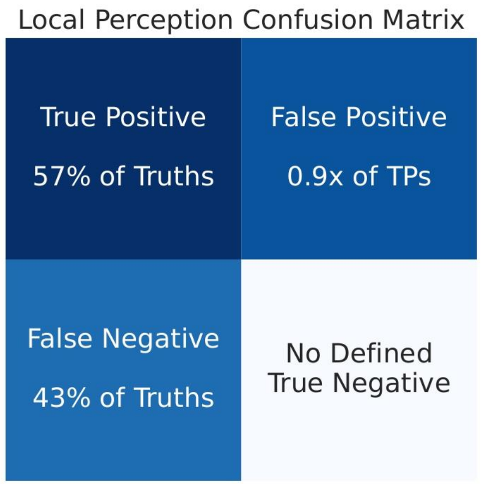


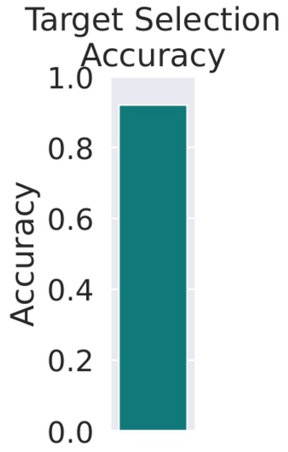
Monitoring and target selection

Monitoring for sensor height

Target selection for situational awareness







Thank you