General Black-box Adversarial Sensor Attack and Countermeasures

Robust LiDAR-based Perception in Autonomous Driving

(Charles) Chengzeng You
Connected and autonomous vehicles
Department of computing
Email: chengzeng.you19@imperial.ac.uk

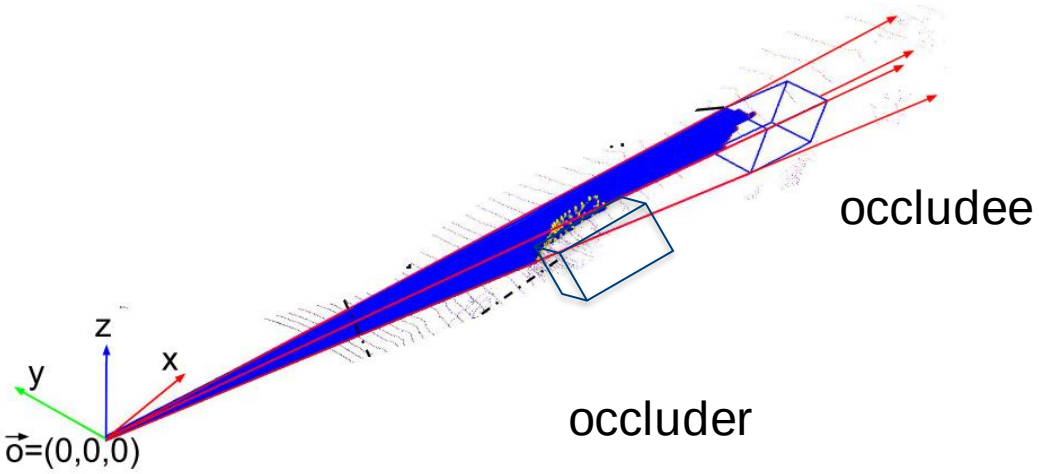
Imperial College London

Content

Imperial College London

6.1 Occlusion patterns

- state-of-the-art 3D object detection model designs generally ignore the occlusion patterns in LiDAR point clouds.



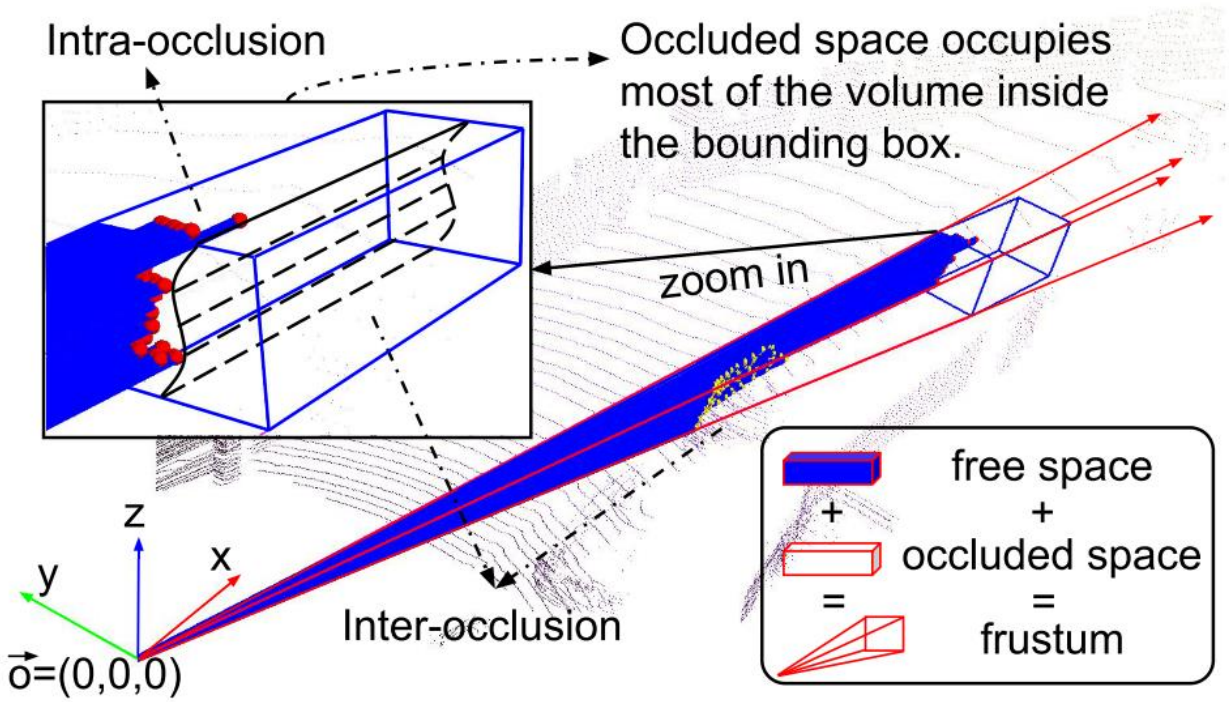
Imperial College London

Content

Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.1 Free Space Detection(FSD)

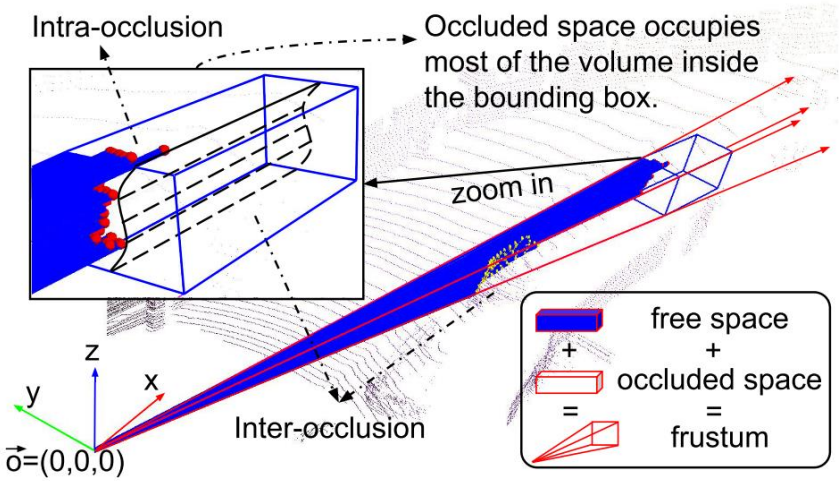


Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.1 Free Space Detection(FSD)

Observation:
f = the volume of FS / the volume of a detected bounding box
∃b ∈ (0,1), implying f ∈ (0,b]
For fake vehicles:
∃a ∈ (0,1) such that f' ∈ [a,1).



Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.1 Free Space Detection(FSD)

$$f_{\mathrm{B}} = \frac{\sum_{c \in \mathrm{B}} \mathbb{1} \cdot FS(c)}{|\mathrm{B}|}$$

FS(c) indicates whether the cell c is free or not

|B| denotes the total number of cells in the bounding box B

Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.1 Free Space Detection(FSD)

100 ms/bounding box

Setup

cell size: 0.25

Dataset: KITTI training set

600 generated attack traces

cumulative distribution function:

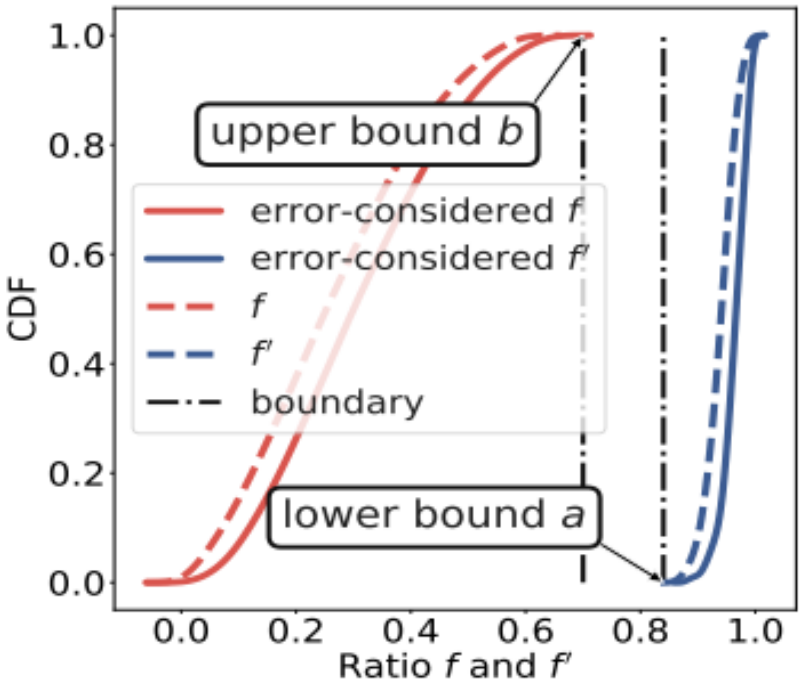
$$F_X(x) = P(X \leq x)$$

$F_X(x)$ = function of X

X = real value variable

P = probability that X will have a value less than or equal to x

cumulative distribution function of f



Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.2 Laser Penetration Detection (LPD)

three spaces of one frustum:

1) the space between the LiDAR sensor and the bounding box B ↑
2) the space inside the bounding box B
3) **the space behind the bounding box B ↓**

Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.2 Laser Penetration Detection (LPD)

5 ms/bounding box

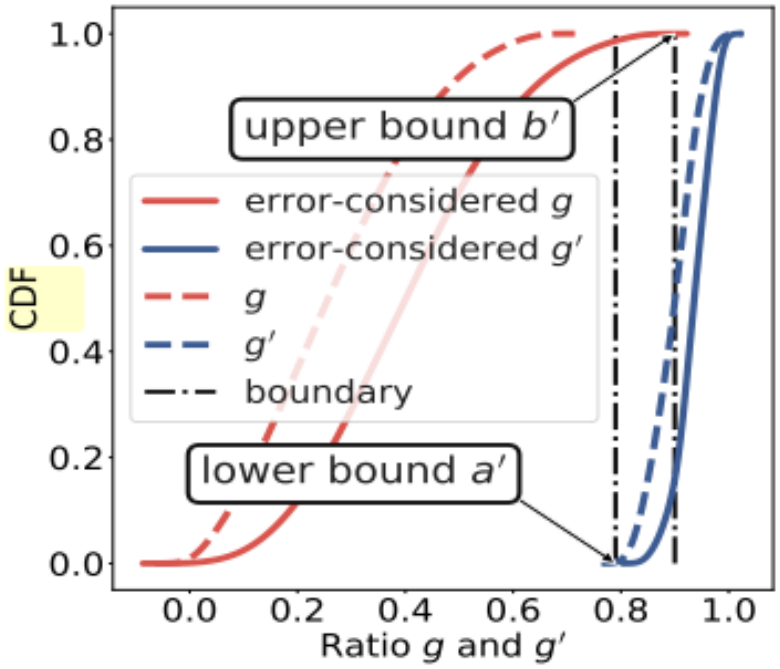
$$g_{\mathrm{B}} = \frac{\sum_{\vec{p} \in \mathrm{B}\downarrow} \mathbb{1}(\vec{p})}{\sum_{\vec{p} \in \mathrm{B} \cup \mathrm{B}\downarrow \cup \mathrm{B}\uparrow} \mathbb{1}(\vec{p})}$$

For benign vehicles:
g should be upper bounded by ∃b' ∈ (0,1).
For spoofed vehicles:
g' should be lower bounded by ∃a' ∈ (0,1).

Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.2 Laser Penetration Detection (LPD)



b' > a'
Might cause erroneous detection of potential anomalies

Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.3 Hierarchy Design

around 8.5 ms / each vehicle

Algorithm 1: CARLO

input: Detected bounding boxes $\boldsymbol{B} = \{\mathsf{B}\}$;
LiDAR laser ray directions $\boldsymbol{L} = \{\mathsf{L}\}$;
3D point cloud $\boldsymbol{X} = \{\vec{p}\}$;
1 Threshold of FSD $\frac{a+b}{2}$;
Thresholds of LPD $b' + \varepsilon, a' - \varepsilon$;
output: Valid bounding boxes $\boldsymbol{B}_{\text{valid}} = \{\mathsf{B}\}$;
Adversarial bounding boxes $\boldsymbol{B}_{\text{adv}} = \{\mathsf{B}\}$;

2 **Initialization** : $\boldsymbol{B}_{\text{valid}} \leftarrow \emptyset, \boldsymbol{B}_{\text{adv}} \leftarrow \emptyset, g \leftarrow 0, f \leftarrow 0$;
/* Initiate parameters. */

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.3 Hierarchy Design

```
for B ∈ B do
    /* Initiate parameters, where FS(·) is the
       free space and F_B is the frstum of B.   */
    F_B ← ∅, FS(·) ← ∅;
    for L ∈ L do
        /* Predict whether L will intersect with
        B.                                       */
        if L ∩ B then
            p⃗_L ← L;
            F_B.append([L, p⃗_L]);
        /* Extract the frustum F_B of B.        */
    end
    g ← Equation 6;
    /* Calculate g by F_B for B (LPD).          */
```

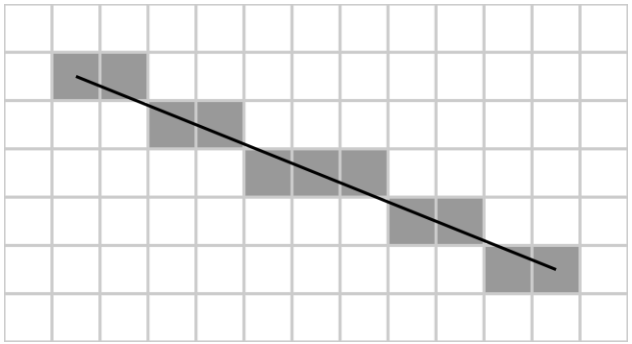
Imperial College London

7.1 CARLO(oCclusion-Aware hieRarchy anomaLy detectiOn)

7.1.3 Hierarchy Design

```
    if g < a' − ε then
        B_valid.append(B);
        /* Certainly valid vehicles.                    */
    else if g > b' + ε then
        B_adv.append(B);
        /* Certainly spoofed vehicles.                  */
    else
        /* Calculate f by F_B for B (FSD).              */
        for [L, p_L] ∈ F_B do
            FS(L) ← Bresenham([L, p_L])[16];
            FS(B) ← FS(B) ∪ FS(L) ;
        end
        f ← Equation 5;
        if f < (a+b)/2 then
            B_valid.append(B);
        else
            B_adv.append(B);
end
Return : B_valid, B_adv;
```

Illustration of the result of Bresenham's line



Imperial College London

7.2 CARLO Evaluation

7.2.1 Experimental setup

Dateset K: existing real-world traces
Dateset R: simulated attack traces
Adversarial model: Adv-LiDAR [17] on Apollo 5.0
Target location: 5-8 meters in front of the victim
CARLO-guarded models: CARLO(M(·))

Defense goal: successfully detect the spoofed fake vehicles from the output bounding boxes without hurting the original performance of the target models

Imperial College London

7.2 CARLO Evaluation

7.2.2 Evaluation metrics

$$ASR = \frac{\text{\# of successful attacks}}{\text{\# of total point cloud samples}}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$\mathrm{AP} = \int_0^1 p(r)dr$$

TP = True positive

TN = True negative

FP = False positive

FN = False negative

Imperial College London

7.2 CARLO Evaluation

7.2.2 Evaluation metrics

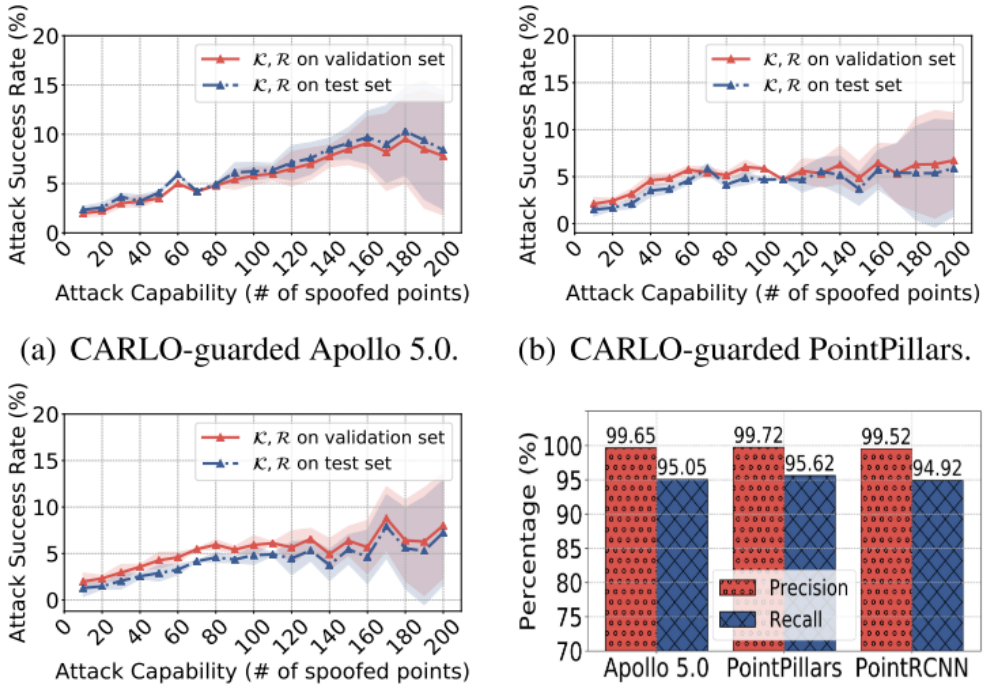


Figure 12: Attack success rates (*ASR*s) of proposed black-box spoofing attacks on three CARLO-guarded models.

reduce the mean ASR to around 5.5%.

the remaining 5.5% comes from the detection errors .

Imperial College London

7.2 CARLO Evaluation

7.2.2 Evaluation metrics

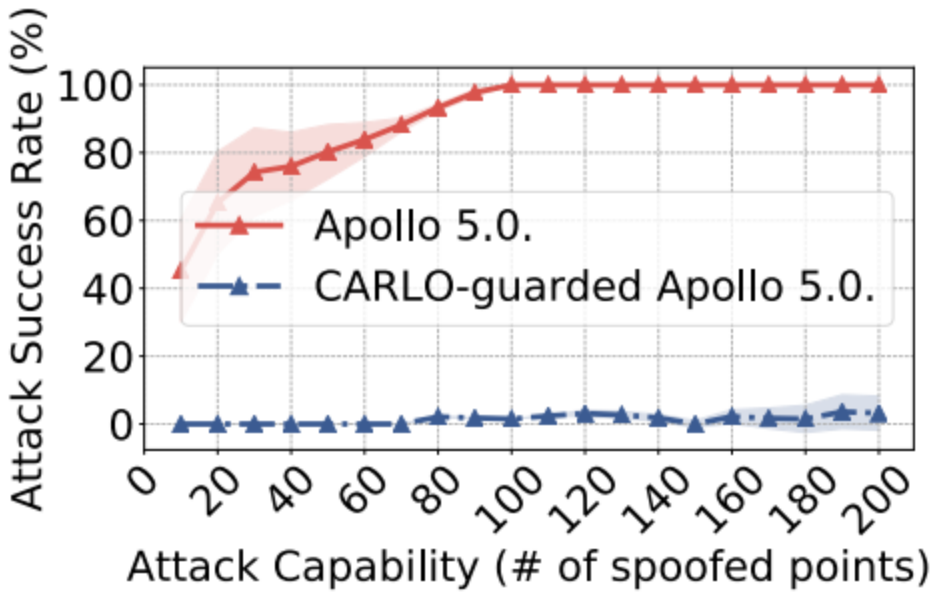
Table 1: PointPillars' and PointRCNN's APs (%) of 3D car detection on the KITTI validation set.

Model	PointPillars			PointRCNN		
	Easy	Mod.	Hard	Easy	Mod.	Hard
Original	86.56	76.87	72.09	88.80	78.58	77.64
Attack	74.06	56.69	53.98	84.51	71.17	68.06
CARLO	86.57	78.60	73.55	88.91	78.61	77.63

Imperial College London

7.2 CARLO Evaluation

7.2.3 Defense against White-box Attacks



Imperial College London

7.2 CARLO Evaluation

7.2.4 Defense against Adaptive Attacks

$$f_{\mathrm{B}} = \frac{\sum_{c \in \mathrm{B}} \mathbb{1} \cdot FS(c)}{|\mathrm{B}|}$$

absolute free space

Imperial College London

7.2 CARLO Evaluation

7.2.4 Defense against Adaptive Attacks

Attack goal:

1.to spoof a vehicle at target locations

2.minimize the size of the bounding box

$$\min_{\theta,\tau} \quad \mathcal{L}(x \oplus V \cdot H(\theta,\tau)^T) + \lambda \cdot \mathcal{V}_B(V \cdot H(\theta,\tau)^T)$$

L(.): the adversarial loss, given a detector

x : the corresponding input feature matrix

V : adversarial spoofed input feature matrix

$\oplus(\cdot)$: merging function

$H(\theta,\tau)$: transformation matrix

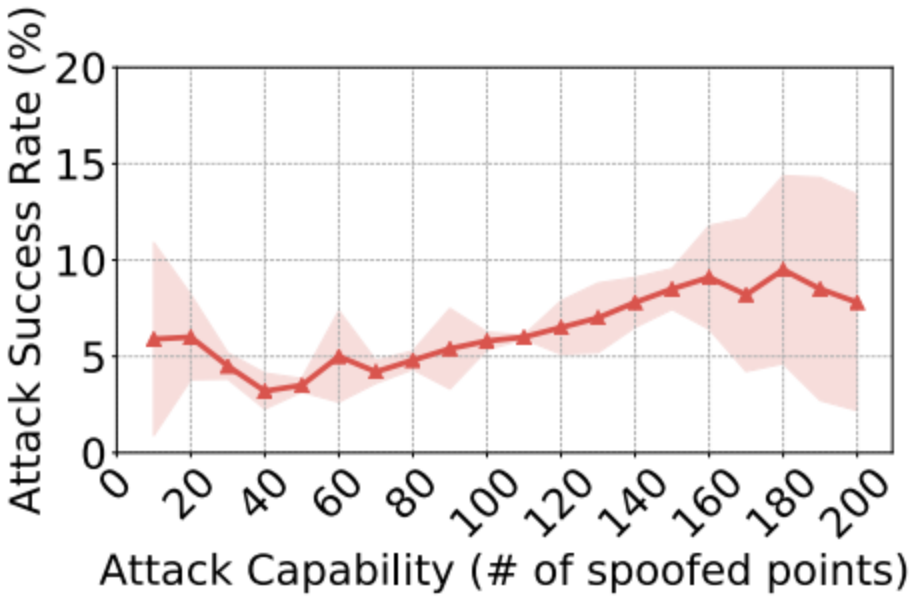
$V_B(\cdot)$: the volume of the target bounding box B,

λ : a hyper-parameter

Imperial College London

7.2 CARLO Evaluation

7.2.4 Defense against Adaptive Attacks



Imperial College London

Content

Imperial College London

8.1 Sequential View Fusions(SVF)

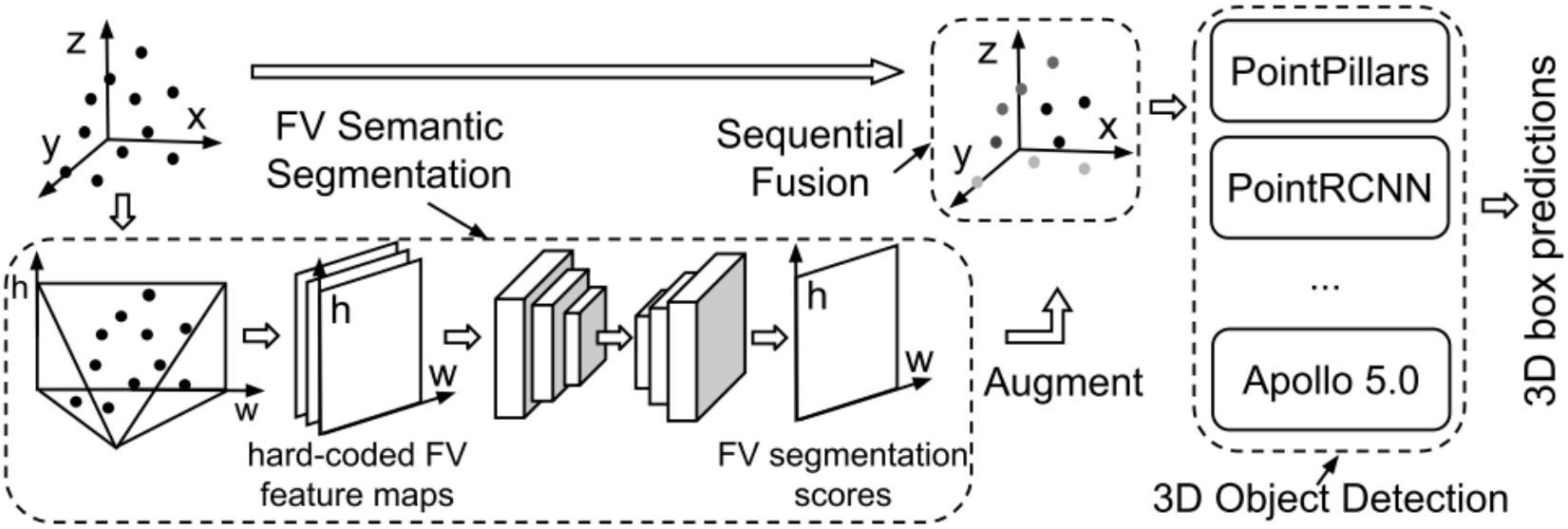


Figure 17: Sequential view fusion (SVF) architecture.

Imperial College London

8.2 SVF Evaluation

8.2.1 Experimental setup

Target model: trained SVF-PointPillars and SVF- PointRCNN
Dataset : KITTI training set.
Evaluate SVF against Adv-LiDAR [17] on Apollo 5.0 and the adaptive attacks.

Imperial College London

8.2 SVF Evaluation

8.2.2 Evaluation metrics

leverage ASR to test their robustness against LiDAR spoofing attacks

Table 3: SVF-PointPillars' and SVF-PointRCNN's APs (%) of 3D car detection on the KITTI validation set.

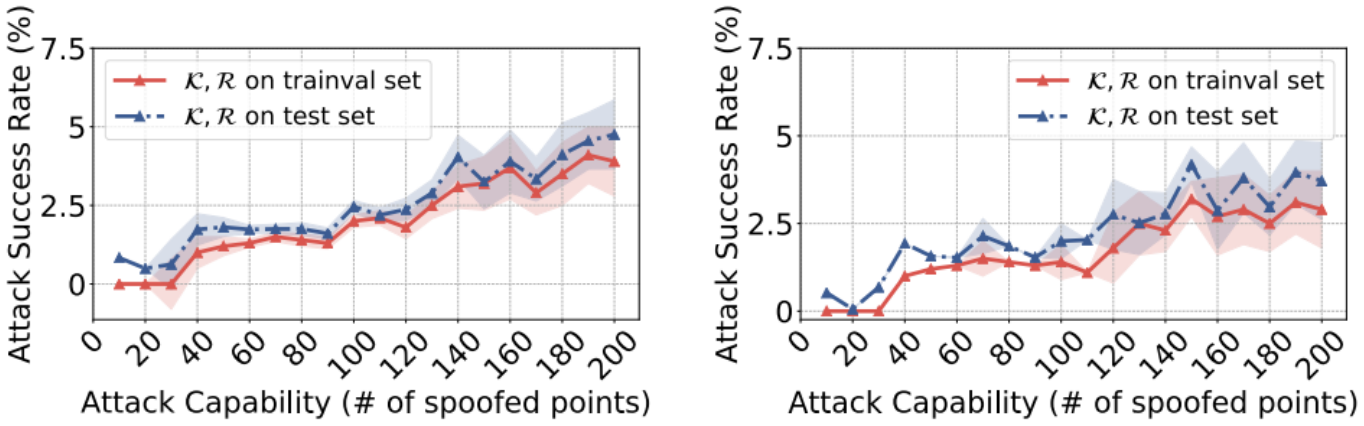
Model	Car Detection		
	Easy	Moderate	Hard
SVF-PointPillars	85.93	74.12	70.19
SVF-PointRCNN	88.12	76.56	74.81

Imperial College London

8.2 SVF Evaluation

8.2.2 Evaluation metrics

leverage ASR to test their robustness against LiDAR spoofing attacks



(a) *ASR* of SVF-PointPillars. (b) *ASR* of SVF-PointRCNN.

Figure 18: Attack success rates (*ASR*s) of proposed black-box spoofing attack on SVF models.

Imperial College London

8.2 SVF Evaluation

8.2.3 Defense against White-box Attacks

Since SVF requires re-training for the model, cannot directly evaluate Adv-LiDAR on SVF-Apollo

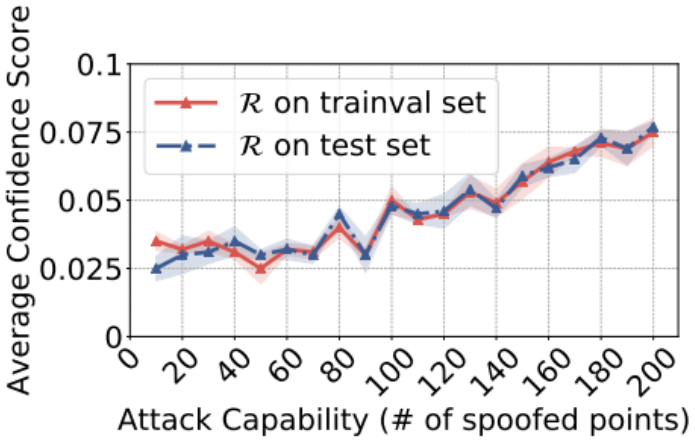


Figure 19: Average confidence score of Adv-LiDAR on the segmentation network.

Imperial College London

8.2 SVF Evaluation

8.2.4 Defense against Adaptive Attacks

Assume the adversaries are aware of the SVF architecture.
Attack goal: both fool the semantic segmentation and 3D object detection modules.

$$\min_{\theta,\tau} \quad -\mathcal{L}_{seg}\left(x \odot V \cdot H(\theta,\tau)^T\right)$$

Imperial College London

8.2 SVF Evaluation

8.2.4 Defense against Adaptive Attacks

Assume the adversaries are aware of the SVF architecture.
Attack goal: both fool the semantic segmentation and 3D object detection modules.

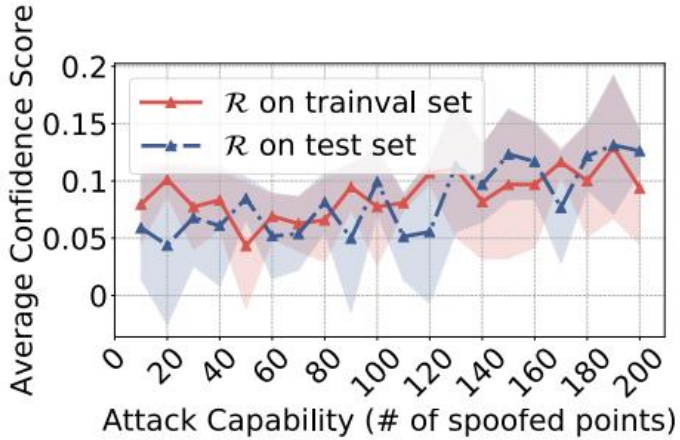


Figure 20: Average confidence score of the adaptive attack on the segmentation network.

Imperial College London

Content

Imperial College London

9 Limitations

1.**same pattern** at the **target location**, whether this defense strategy is transferable to other LiDAR spoofing attacks (e.g.other patterns) still remains unexplored.

2.the identified vulnerability does not provide completeness, there may exist other potential vulnerabilities hidden in the AD systems to be discovered and exploited

3. Lack of attacks on other types of objects, such as perdestrains

4. Lack of real-world validations

Imperial College London

Thank you