Fooling Object Detection is Not Enough:
First Adversarial Attack agiaist Multiple Object Tracking

Yunhan Jia[1], Yantao Lu[1], Junjie Shen[2], Qi Alfred Chen[2], Zhenyu Zhong[1], Tao Wei[1]

[1] Baidu X-Lab , [2] University of California Irvine

Introduction

• Adversarial Examples (AE) against object detection models have been studied, and are believed to be a realistic threat to autonomous driving.

E.g., adversarial patch on stop signs (Eykholt et al. [1])

• However, in a visual perception pipeline, detected objects will also be tracked, in a process called Multiple Object Tracking **(MOT)**, to build the moving trajectories of surrounding obstacles.

• We find that existing attacks that blindly target on object detection models are highly ineffective.

• We are the first to study the adversarial learning against complete visual pipeline in autonomous driving, and discover novel attack, **tracker hijacking**, which can move an object in or out of the headway of an autonomous vehicle to cause safety hazards using as few as **one frame**.

MOT Background

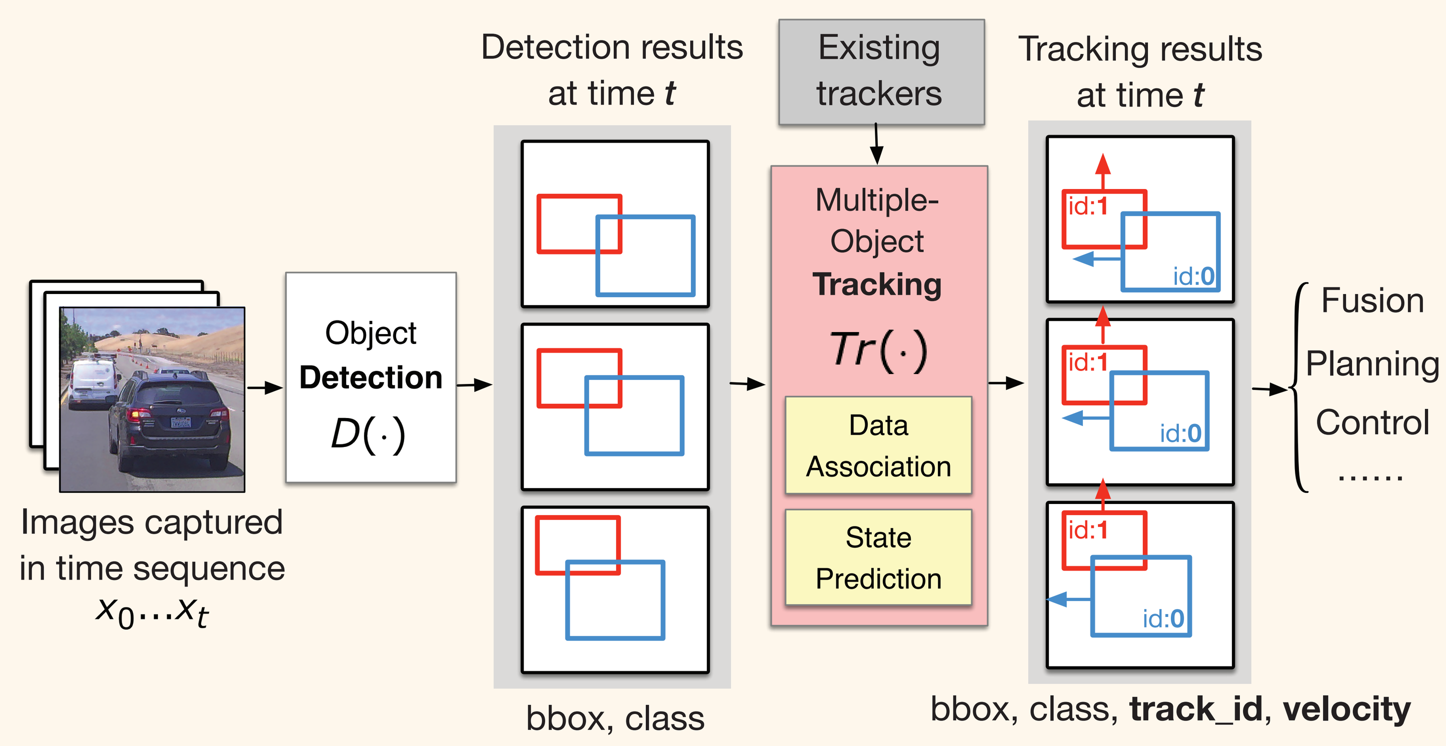
• MOT identifies objects and their trajectories in video frames. Three major components:

• **Data association** between detected objects and existing trackers is formulated as a bipartite matching problem, based on the pairwise similarity between the bounding boxes.

• **State prediction** is performed using a per-track Kalman Filter maintains a velocity model to estimate the locations of the tracked objects in the next frame in order to compensate the motion between frames.

• **Track management** controls the creation and deletion of trackers. A new tracker will be created only when being constantly detected for H frames (Hit Count); A tracker will be deleted only if no objects is associated with for a duration of R frames (Reserved Age).

Figure1. The complete *Track-by-Detection* pipeline of modern autonomous systems.



Recommended setting: R=60, H=6 for 30 fps video [2].

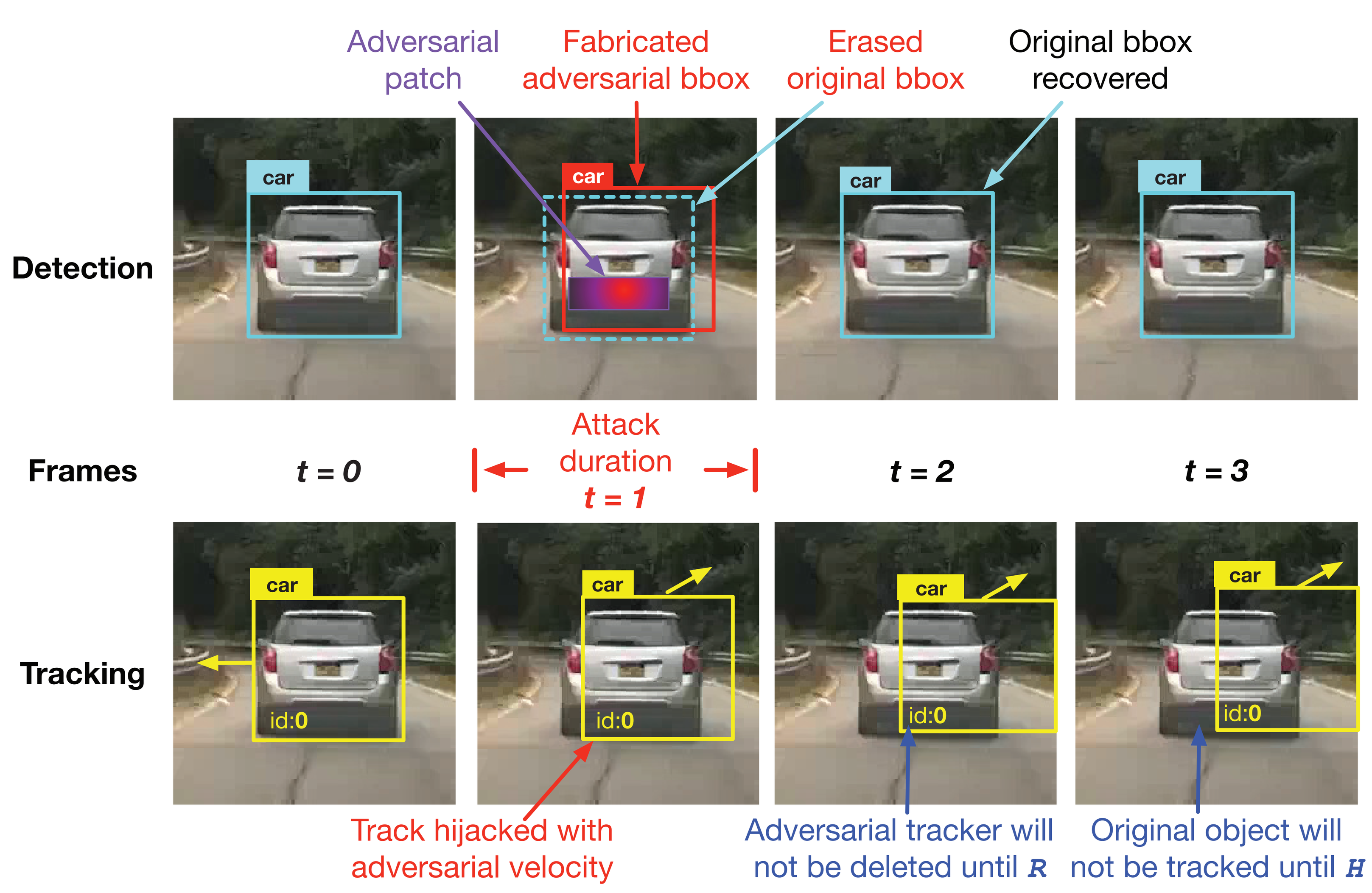


Figure2. One-frame tracker hijacking attack workflow

Tracker Hijacking Attack

• Overview: Generate an adversarial patch to fool the object detector model with two adversarial goals:

1. Erase the bounding box of target object from detection result.
2. Fabricate a bounding box with similar shape that is shifted a little bit towards an attacker-controlled direction

• The fabricated bounding box (red) will be associated with the original tracker, and thus would give a fake velocity towards the attacker-controlled direction.

• In the example, the attack lasts for only one frame, however:
The hijacked tracker will not be deleted until a reserved age **(R)** has passed.

• The target object, though is recovered in the detection result, will not be tracked util a hit count **(H)** has reached. And before that, the object remains missing in the tracking result.

• Causing rear-end crashes in two attack scenarios.

Attack Effectiveness

• Definition of a **successful attack:** the detected bounding box of target object can no longer be associated with any of the existing trackers when attack has stopped.

• Evaluation dataset: 20 video clips from Berkeley Deep Drive (BDD) datasets, 10 for move-in scenario, and 10 for move-out scenario,

• Implementation: MOT implemented based on the one used in OpenCV, Object detection adopts YOLOv3. The number of frames required for a successful attack depends on a parameter, called measurement noise covariance of Kalman filter. We test under different noise level.

Finding optimal position for adversarial bounding box: finding translation δ that minimizes the cost of Hungarian matching $\mathcal{M}(\cdot)$

$$\max_{\delta} \mathcal{M}(detc|_t[K]+\delta, track|_{t-1}[K])$$

$$s.t.\ \mathcal{M} \le \lambda, IoU(detc|_t[K]+\delta, patch) \geqslant \gamma$$

• [Figure 5]: Tracker hijacking attack only requires successful AEs on object detection in **2~3** consecutive frames on average to succeed despite different (R, H) configurations.

Object move-in generally requires less frames compared with object move out,

• [Fugure 6] Tracker hijacking achieves superior (100%) success rate even even by attacking only 3 frames, while detection attack needs to reliably fool at least **R** consecutive frames, which translates to a 98.3% (59/60) AE success rate for a 30 fps video system, which has never been achieved by previous work [1, 3, 4]. Otherwise, object detection attack only has up to **25%** success rate before R.

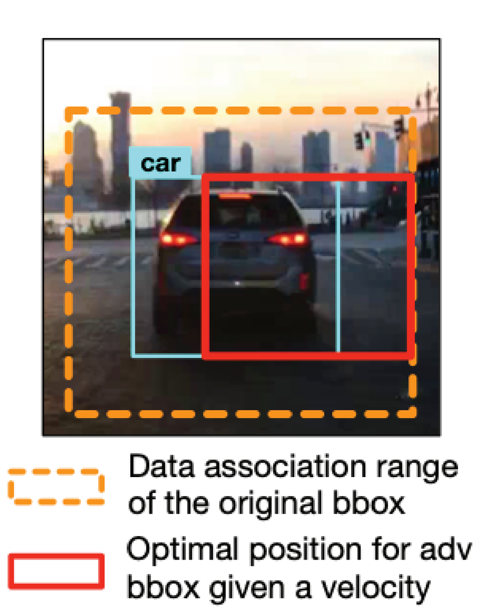


Figure3. Finding position to place fabricated bounding box

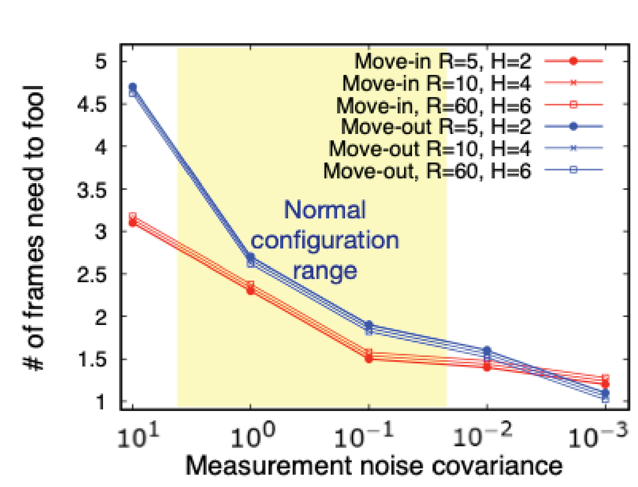


Figure5. Frames required to be fooled for successful attack

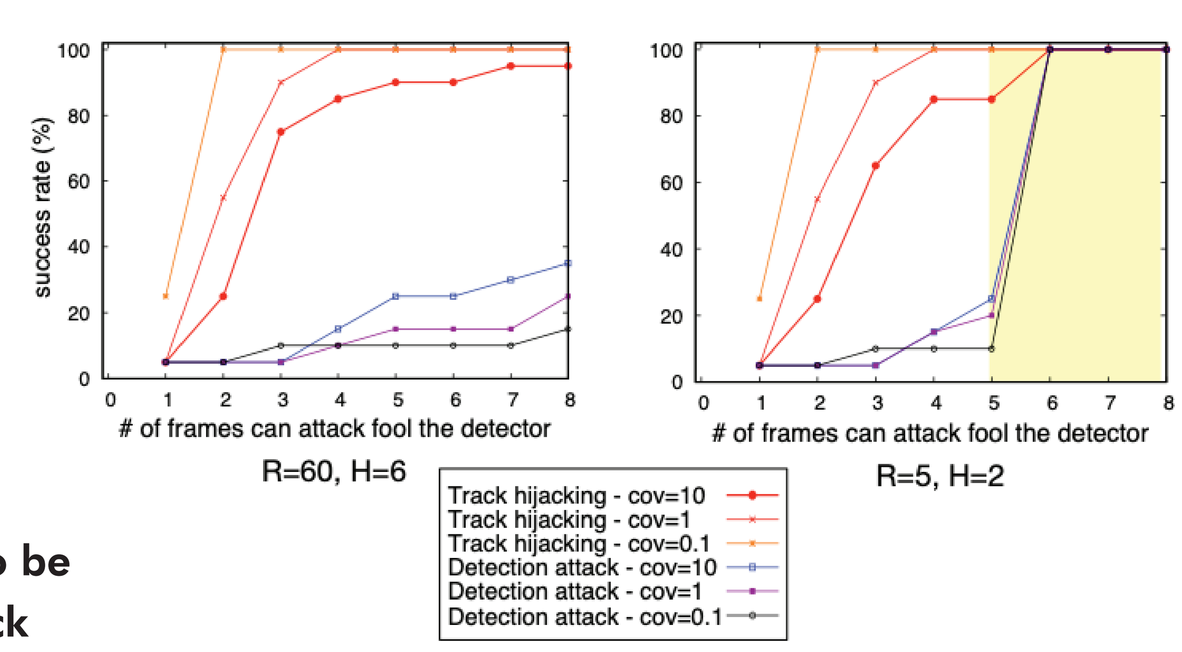
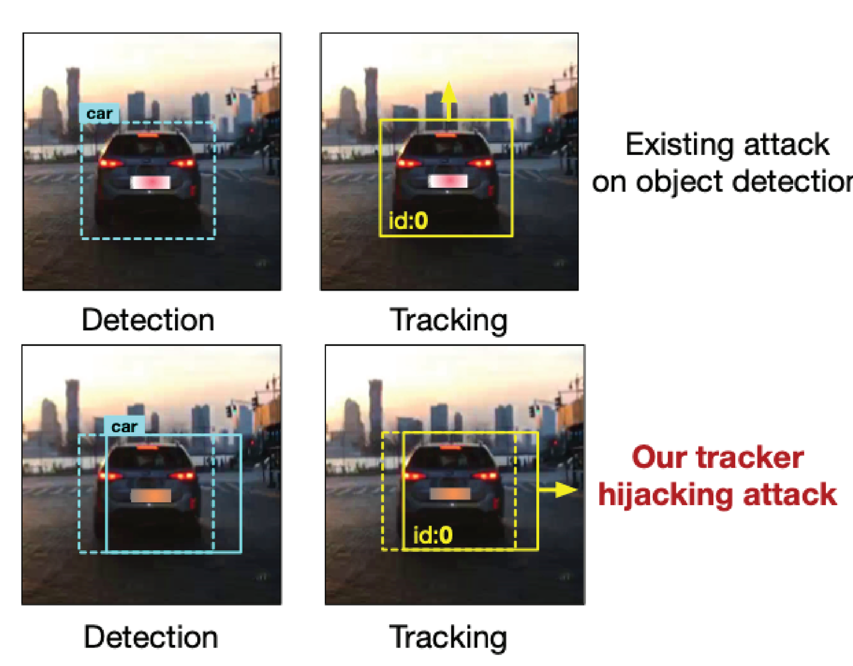


Figure6. Attack success rate at R=60 H=6, and R=5, H=2



References

1. K. Eykholt et al. “Physical adversarial examples for object detectors”. WOOT 18’
2. P. Bergman et al. “Tracking without bells and whostles”. CoRR 19’
3. J.Lu et al. “Adversarial examples that fools detectors” arXiv, 17’
4. Y. Zhao et al. “Practical physical adversarial attack against object detector” arXiv 18’
5. S. Chen, “Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector. ” ECML PKDD 18’
6. Z. Zhong et al. “Perception deception: Physical adversarial attack against object detector” BHEU 18’

Baidu X-Lab UCIRVINE

https://github.com/advboxes/perceptron-benchmark
Welcome to check out our project!