Robustness of Trajectory Prediction Models Under Map-Based Attacks: Supplementary Materials

Zhihao Zheng
Lehigh University
zhzc21@lehigh.edu

Xiaowen Ying
Lehigh University
xiy517@lehigh.edu

Zhen Yao
Lehigh University
zhy321@lehigh.edu

Mooi Choo Chuah
Lehigh University
chuah@cse.lehigh.edu

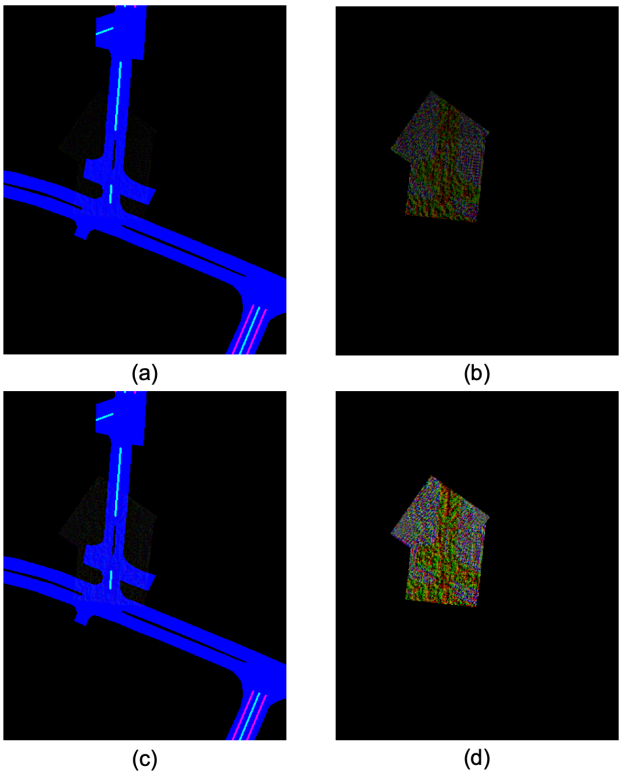


Figure 1. Visualization of maps created using different constraints for continuous perturbation: (a) and (b): perturbation with $\epsilon = 0.03$; (c) and (d): perturbation with $\epsilon = 0.06$. (b) and (d) show the perturbation that are magnified 10 times larger for a clearer view.

1. Imperceptible Perturbation

In this section, we present the discussion and analysis of perturbation constraints we explore in the main paper.

1.1. Image-based Map

Continuous Perturbation. In this paper, we set the constraint $\epsilon = \frac{8}{255} = 0.03$ to satisfy the imperceptible requirement. Here, we investigate what happens if we set ϵ to 0.06 (equal to $\epsilon = 16$ for images [0,255]). As shown in Figure 1, perturbation on *Trajectron++* with $\epsilon = 0.06$ is more noticeable than the one with $\epsilon = 0.03$, considering the map we attack is a binary image in nuScenes dataset.

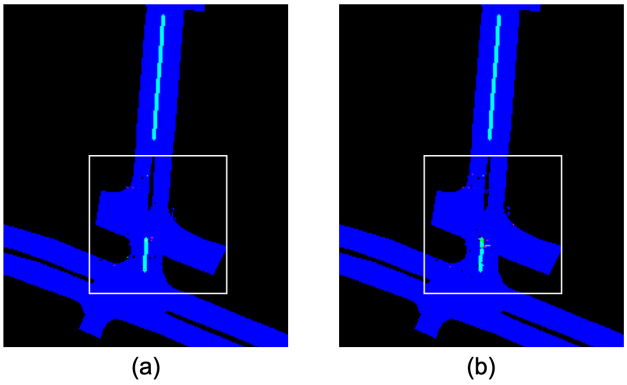


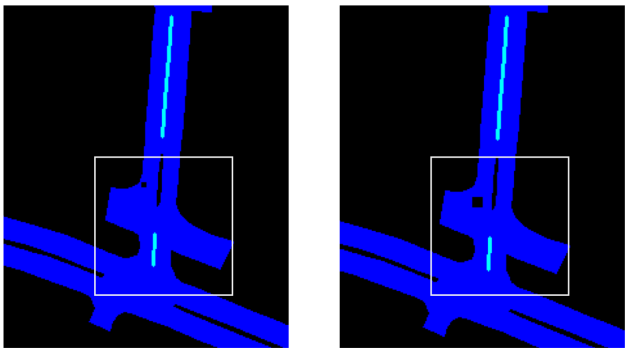
Figure 2. Visualization of attack outcomes with different constraints for sparse pixel binary perturbation. (a): perturbation with $\epsilon = 50$; (b): perturbation with $\epsilon = 75$. The white bounding box in each figure shows the local map area (100×100).

Binary Perturbation. For binary perturbation, we set $\epsilon = 50$ as the maximum number of pixels that can be modified by the attack within a local map of size 100×100. We visualize the sparse pixel binary perturbation that are generated by white box attacks on *Trajectron++* with different ϵ in Figure 2. The white bounding box in each figure shows the local map area. Perturbation with $\epsilon = 50$ is more natural than the one with $\epsilon = 75$.

With black box attacks on image-based map models, we denote each particle as a list of patches to generate binary perturbation with PSO. Thus, we not only restrict the binary perturbation with constraint $\epsilon = 50$ but also limit the patch size to 6×6 for the imperceptible requirement. Figure 3 demonstrate the difference between two patches sizes: 6×6 and 8×8. It is obvious that a patch of size 8×8 is too noticeable compared with the smaller one.

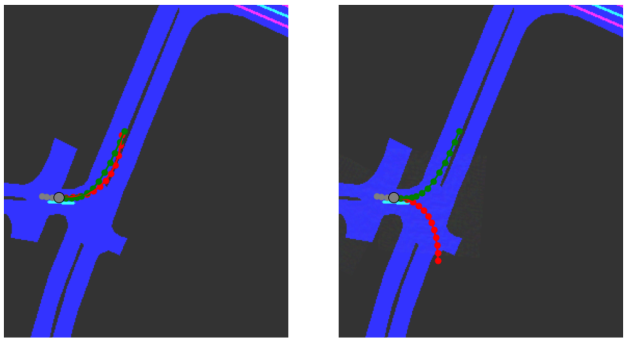
1.2. Node-based Map

With our proposed attacks on node-based map models, we generate adversarial perturbation that can change the x-



(a) (b)

Figure 3. Visualization of outcomes with different constraints for sparse pixel binary perturbation. (a): perturbation with $\epsilon = 50$; (b): perturbation with $\epsilon = 75$. The white bounding box in each figure shows the local map area (100×100).



(a) (b)

Figure 4. Visualization of white box attacks on Trajectron++. (a): original trajectory prediction; (b): attacked trajectory prediction. Green points are ground-truth future trajectory, red points are predicted future trajectory, gray points are history trajectory

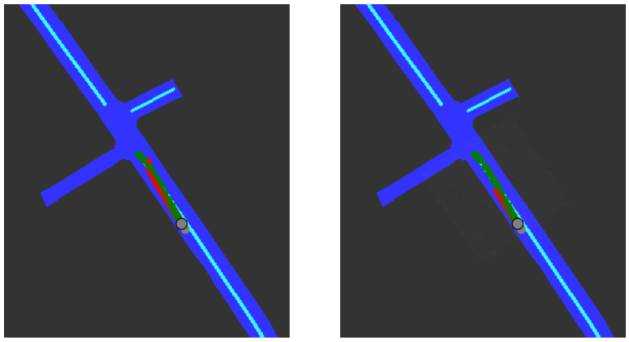
y coordinates in the map nodes. Thus, we set the constraint of perturbation based on the limits of physical properties. Given that the urban lane width is around 4 meters in the nuScenes dataset, we set 1 meter as the maximum bound for the perturbation as all nodes will stay in their lane even with perturbation.

2. Qualitative Analysis.

We provide two scenarios in the main paper to reveal the impact of our proposed attacks on both image-based and node-based map encoding models. In this section, we show three more scenarios to demonstrate attack impacts under various situations.

2.1. Turning at the intersection.

As we mentioned in the main paper, we observe that the TP models are very vulnerable to our proposed attacks at the intersection, which is a much more complicated road situation. Here, we provide two scenarios under such a circumstance, one from *Trajectron++* and the other from *PGP*.



(a) (b)

Figure 5. Visualization of white box attacks on Trajectron++. (a): original trajectory prediction; (b): attacked trajectory prediction. Green points are ground-truth future trajectory, red points are predicted future trajectory, gray points are history trajectory

In Figure 4, the vehicle is going to turn left at the intersection in the future trajectory (green points). The TP model makes a correct turning prediction with ground truth map representation. However, after the attack, adversarial perturbation provides the model with wrong map features and fools the model to predict a right turn in the future, maximizing the prediction errors. Similarly, in Figure 6, the model correctly predicts the vehicle driving straight forward along its lane in the future. But after adding perturbation to the map nodes, the vehicle is predicted to make a left turn at the intersection.

In both scenarios, the victim models make a totally wrong turning prediction in the future trajectory after our proposed attacks. Our experimental results show that turning at the intersection is the most common road situation where map-based attacks cause high prediction errors.

2.2. Driving along the lane

In the main paper, we show one scenario where adversarial perturbation causes a large deviation along the lane and makes the victim model predict a fake lane shift in the future trajectory. Except for the deviation of trajectory, speed changes can also cause high prediction errors along the lane by our proposed attacks.

As shown in Figure 5, the vehicle is driving forward along the lane at a steady speed in the future. Without perturbation, the model can make a proper prediction of the vehicle's future trajectory in Figure 5(a). However, the model predicts that the vehicle will slow down in the near future after the attack in Figure 5(b). Without changing the direction of the prediction trajectory, such an attack can potentially cause serious danger or result in an accident.

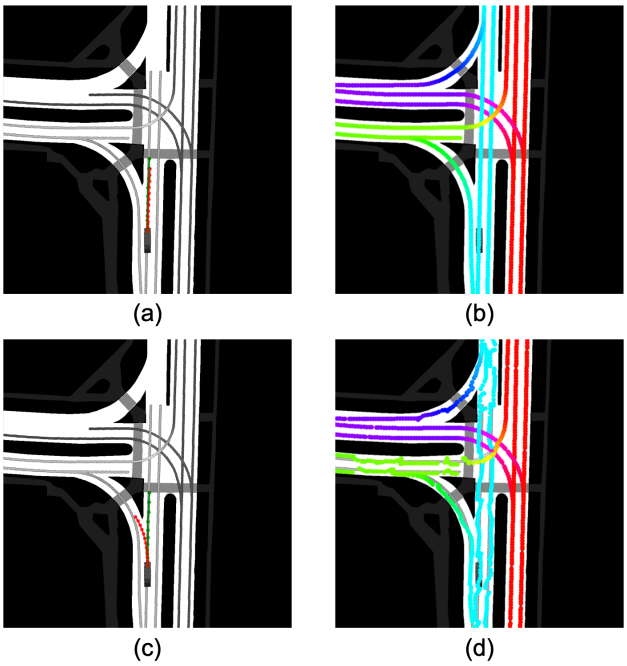


Figure 6. Visualization of white box attacks on PGP. (a) and (b): original trajectory prediction and map; (c) and (d): attacked trajectory prediction and map. Green points are ground-truth future trajectory, red points are predicted future trajectory in the left column. In the right column, each point is a map node and the color of each node is based on its rotation angle.