

# CP-FREEZER: Latency Attacks Against Vehicular Cooperative Perception

Chenyi Wang<sup>1</sup>, Ruoyu Song<sup>2</sup>, Raymond Muller<sup>2</sup>, Jean-Philippe Monteuis<sup>3</sup>,  
Z. Berkay Celik<sup>2</sup>, Jonathan Petit<sup>3</sup>, Ryan Gerdes<sup>4</sup>, Ming Li<sup>1</sup>

<sup>1</sup>University of Arizona

<sup>2</sup>Purdue University

<sup>3</sup>Qualcomm

<sup>4</sup>Virginia Tech

{chenyiw, lim}@arizona.edu, {song464, mullerr, zcelik}@purdue.edu

{jmonteui, petit}@qti.qualcomm.com, rgerdes@vt.edu

## Abstract

Cooperative perception (CP) enhances situational awareness of connected and autonomous vehicles by exchanging and combining messages from multiple agents. While prior work has explored adversarial integrity attacks that degrade perceptual accuracy, little is known about CP’s robustness against attacks on timeliness (or availability), a safety-critical requirement for autonomous driving. In this paper, we present CP-FREEZER, the first latency attack that maximizes the computation delay of CP algorithms by injecting adversarial perturbation via V2V messages. Our attack resolves several unique challenges, including the non-differentiability of point cloud preprocessing, asynchronous knowledge of the victim’s input due to transmission delays, and uses a novel loss function that effectively maximizes the execution time of the CP pipeline. Extensive experiments show that CP-FREEZER increases end-to-end CP latency by over 90×, pushing per-frame processing time beyond 3 seconds with a 100% success rate on our real-world vehicle testbed. Our findings reveal a critical threat to the availability of CP systems, highlighting the urgent need for robust defenses.

## Introduction

Cooperative perception (CP) allows connected and autonomous vehicles (CAVs) to share sensor data via V2V messages, fusing them to detect objects beyond the line-of-sight and mitigate risks from limited single-vehicle visibility (Cui et al. 2022). Despite its promise, CP is vulnerable to adversarial attacks. Previous work has focused on integrity attacks that degrade perception accuracy via object spoofing or removal (Tu et al. 2021; Zhang et al. 2024). However, the robustness of CP systems regarding timeliness—a critical *availability* requirement for autonomous driving systems (ADS) (Lin et al. 2018)—remains unexplored. If a CP system fails to produce timely detections, downstream ADS modules will make uninformed decisions, potentially leading to collisions (Wachenfeld et al. 2016). Existing integrity defenses (Li et al. 2023b; Zhang et al. 2024) are ineffective against such threats as they require detection results to be available for their operation.

In this paper, we present CP-FREEZER, the first latency attack against CP systems. As illustrated in Figure 1, CP-

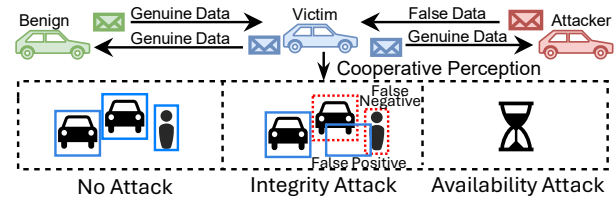


Figure 1: Integrity attacks degrade detection quality, whereas CP-FREEZER *disables* perception by inducing extreme latency, which can lead to severe consequences.

FREEZER maximizes the execution time of CP algorithms, disrupting the *availability* of the perception pipeline and endangering real-time vehicle decisions. Realizing CP-FREEZER involves several challenges: C1 *Non-differentiable pre-processing*: Modern CP involves non-differentiable LiDAR feature extraction, complicating gradient-based optimization. Prior solutions for single-agent systems (Liu et al. 2023) are memory-intensive and too slow for real-time attacks on state-of-the-art (SOTA) point cloud preprocessors (Lang et al. 2019). C2 *Novel latency-inducing loss for CP*: Previous attacks on CP aim to decrease the detection accuracy regarding particular objects (Zhang et al. 2024), which require precise prior knowledge of the ground truth (otherwise the gradient becomes zero) and cannot increase the execution time of CP. Inflating latency poses a fundamentally different objective that needs to be carefully crafted. C3 *Asynchronous attacker knowledge*: In a realistic V2V setting, data is transmitted asynchronously. An attacker cannot rely on a synchronized global state and must craft perturbations with delayed information from the victim.

We address these challenges as follows. For C1, we exploit the trend of intermediate fusion in SOTA CP systems (Xu et al. 2022) by crafting adversarial perturbations at the bird’s-eye-view (BEV) feature level, bypassing the non-differentiable steps and reducing the computational burden. For C2, our analysis of the CP pipeline reveals that Non-Maximum Suppression (NMS) is a key bottleneck due to its quadratic complexity. We design a novel loss function that maximizes the number of persistent and realistic bounding box proposals to exploit this vulnerability. For C3, we use spatial-temporal warping techniques to align the attacker’s

perturbation with the victim’s timestep, compensating for transmission delays and improving attack effectiveness.

Our contributions can be summarized as follows:

- We present CP-FREEZER, the *first* latency attack against CP systems. It exploits feature-level optimization, a novel latency-inducing loss, and spatial-temporal transformation to reveal a critical, overlooked attack vector.
- We show that CP-FREEZER increases end-to-end CP latency by over  $90\times$ , far surpassing adaptations of existing single-vehicle latency attacks (Ma et al. 2024; Liu et al. 2023), and pushes per-frame processing time beyond 3 seconds with 100% success rates on our real-world testbed.
- We provide an extensive evaluation on the benchmark dataset OPV2V (Xu et al. 2022) across multiple hardware platforms and SOTA CP algorithms (Xu et al. 2022; Lu et al. 2023; Li et al. 2023a; Hu et al. 2022). Our code is open-source at <https://github.com/WiSeR-Lab/CP-FREEZER>.

## Background and Related Work

**Connected and Autonomous Vehicles (CAVs).** CAVs are vehicles equipped with advanced wireless communication systems, which facilitate real-time data sharing with other connected vehicles (V2V), infrastructure (V2I), and devices (V2X). Their connectivity enhances situational awareness beyond individual vehicles’ sensing range and enables collaborative driving. The key technology empowering this is Cooperative Perception (CP), which extends the perceptual abilities of CAVs by allowing them to exchange and integrate remote sensor information (Chen et al. 2019a). The inference task (e.g., 3D object detection) is performed on the collective information fused using the ego vehicle’s local data and those received from nearby CAVs.

**Adversarial Attacks against CP Systems.** Despite the advantages, CP systems present a novel threat model in which adversaries exploit the data-sharing mechanism (Monteuuis et al. 2018). The attacker can inject false data into the CP messages, bypassing physical constraints, making the attack more scalable and easier to execute. Previous work has investigated the degradation of detection accuracy and targeted object spoofing/removal by injecting perturbations into shared V2V messages (Tu et al. 2021; Zhang et al. 2024; Wang et al. 2025). However, these attacks often require precise prior knowledge of the ground truth or substantial raw LiDAR data manipulation. Moreover, adversarial defenses for CP have been proposed to validate or restore the integrity of detection results affected by such attacks (Zhang et al. 2024; Li et al. 2023b).

**Availability Attacks and Efficiency Robustness.** Recent works in other domains have explored availability as a new attack surface (Wang et al. 2022; Liu et al. 2023; Muller et al. 2025). These attacks aim to disrupt the timely and reliable access to information in a system by increasing processing time or energy consumption. Shumailov et al. (2021) introduced ‘sponge examples’ that maximize the energy and latency of NLP models, while showing

Method	Attack Objective	Real-time	Apply to SOTA CP	Prac. Feasible	Latency Effect
PGD	Acc. Degrad.	●	○	●	●
Zhang et al.	Spoof/Remove Obj.	●	●	●	○
SlowTrack	Latency Inc.	○	○	○	●
SlowLiDAR	Latency Inc.	○	○	○	●
Ours	Latency Inc.	●	●	●	●

Table 1: Comparison with existing attacks.

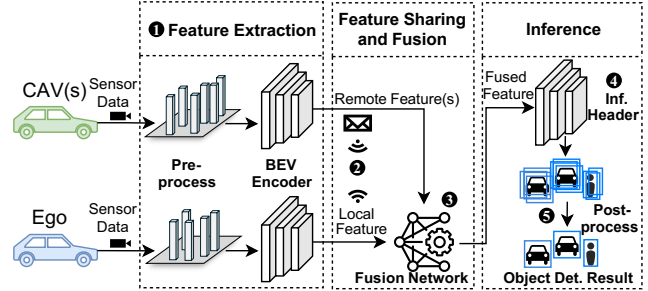


Figure 2: Illustration of the intermediate fusion CP pipeline.

marginal impact on vision models. Subsequent works have explored the availability and efficiency robustness of various models, from vision-language models (Gao et al. 2024) to dynamic deep neural networks (Chen et al. 2023). In the object detection context, previous works have investigated availability attacks on camera-based 2D object detectors (Shapira et al. 2023; Ma et al. 2024; Muller et al. 2025) by perturbing the input images, and on LiDAR-based 3D object detection by manipulating the point cloud input (Liu et al. 2023). Nevertheless, availability attacks on safety-critical CP systems have not been explored, which requires new formulations because of their distinct system pipeline, processing protocols, and the unique challenges introduced by the multi-agent collaborative environment.

## Methodology

### Availability Attack Surface Analysis

**CP Paradigms.** Existing CP algorithms can be categorized into three paradigms based on the stage at which they exchange and fuse data: (1) *Early Fusion*, such as Cooper (Chen et al. 2019b), entails sharing raw sensor data (e.g., LiDAR point clouds) and concatenating the data prior to feature extractions; (2) *Intermediate Fusion*, involves exchanging spatially-aligned BEV features encoded from sensor data, which are then merged using a deep fusion network to form a unified feature for inference (Xu et al. 2022; Chen et al. 2019a; Chen, Shi, and Jia 2023); and (3) *Late Fusion*, combines detections from separate vehicles to create an aggregated result (Rauch et al. 2012).

However, the stringent real-time requirements of ADS (Lin et al. 2018) pose challenges for early fusion, due to the limited V2X communication bandwidth available (e.g., 20 MHz (Abbasi et al. 2022)) and prohibitively

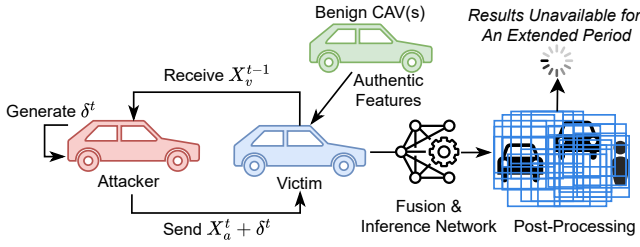


Figure 3: Overview of CP-FREEZER.

large volumes of raw sensor data (e.g., 4 MB per LiDAR frame (Chen et al. 2019a)). In contrast, late fusion suffers from suboptimal detection performance as it solely depends on local information for the inference network (Xu et al. 2022). Therefore, SOTA CP systems (Xu et al. 2022; Lu et al. 2023; Li et al. 2023a; Hu et al. 2022) have adopted intermediate fusion, which achieves superior detection accuracy (Xu 2023) with practical communication and computation overhead. Figure 2 schematically illustrates its core components, and we analyze their potential to be exploited as availability attack surfaces below.

(1) Feature Extraction involves local processing of raw LiDAR points into BEV features. The complexity is linear, and the architecture is fixed, offering no viable attack vector. (2) Message Exchange enables an attacker for wireless jamming (Lin et al. 2025). However, this would cause the victim to revert to single-vehicle perception, which degrades range but not availability. (3) Deep Feature Fusion module widely adopts dot-product attention (Vaswani et al. 2017), which has a complexity of  $\mathcal{O}(N^2 \cdot C \cdot H \cdot W)$  for  $N$  vehicles. An attack would require deploying many malicious CAVs, which is impractical. (4) Inference Network uses fixed CNN layers, which are not influenced by the attacker. (5) Post-processing is the final step that involves Non-Maximum Suppression (NMS), which has a time complexity of  $\mathcal{O}(M^2)$  for  $M$  proposals (Neubeck and Van Gool 2006). Therefore, by adversarially inflating the number of high-confidence, persistent proposals  $M$ , we can trigger a quadratic blow-up in processing time, paralyzing the CP system.

## Problem Formulation

**System Model.** In a multi-vehicle CP system with  $N$  CAVs, each vehicle  $V_i$  computes a BEV feature  $X_i^t$  at time  $t$ . A victim  $V_v$  fuses its feature with those from neighbors  $\mathcal{V}_v$  using a fusion network  $\mathcal{F}(\cdot)$  to get  $\tilde{X}_v^t$ . This is fed to an inference head  $\mathcal{I}(\cdot)$  to generate proposals  $\mathcal{P}_v^t$ , which are finalized by NMS( $\cdot$ ).

**Threat Model.** We assume a white-box attacker controlling a compromised CAV. The attacker can manipulate its outgoing V2V messages to maximize the victim’s end-to-end latency  $\mathcal{T}(\cdot)$ . The goal is to find a perturbation  $\delta^t$  that solves:  $\delta^t = \arg \max_{\delta} \mathcal{T} \left( \text{NMS} \left( \mathcal{I} \left( \tilde{X}_v^t \right) \right) \right)$ , where  $\tilde{X}_v^t$  is the fused feature map containing the attacker’s perturbed feature  $X_a^t + \delta$ .

**Attack Overview.** As shown in Figure 3, CP-FREEZER generates perturbations at the feature level, bypassing non-differentiable preprocessing. The attacker optimizes a perturbation using our latency-inducing loss function and injects it into its shared feature. To handle asynchronous communication, where the attacker only has the victim’s feature from  $t - 1$  when crafting an attack for time  $t$ , we use a feature warping technique to predict the victim’s state and enhance attack effectiveness.

## Adversarial Loss Function Design

To exploit the quadratic time complexity of NMS, the attacker needs to inflate the number of pre-NMS bounding box proposals while ensuring that the resulting proposals remain physically plausible and sparsely distributed to resist efficient filtering. To this end, we combine multiple loss terms that work in tandem: (1) *Confidence Activation*, to produce high-confidence proposals at as many spatial locations as possible; (2) *Shape Regularization*, to prevent the boxes from becoming abnormally large and reduce overlaps; and (3) *Vertical Plausibility*, to ensure that the proposals are within a realistic vertical range (e.g., the ground plane to the height of the car roof). In the following, we outline each term and its interplay in the final joint loss.

**Confidence Activation Loss.** We design a loss to increase the CP model’s objectiveness scores at every anchor box to enable as many high-confidence proposals to be produced as possible. However, pushing scores arbitrarily high wastes gradient steps that could be spent elevating other locations that have not yet passed the confidence filtering threshold. Therefore, we include an activation control parameter  $\tau$ , chosen to match the pre-NMS filtering threshold. Formally, let  $\mathbf{s} \in \mathbb{R}^{B \cdot H \cdot W}$  be the predicted objectiveness scores for the anchor boxes<sup>1</sup>, then the confidence activation loss is  $\mathcal{L}_{\text{conf}} = \frac{1}{BHW} (\mathbf{1}^\top \text{ReLU}(\tau \mathbf{1} - \mathbf{s}))$ , where  $\mathbf{1} = [1, \dots, 1] \in \mathbb{R}^{B \cdot H \cdot W}$ . Minimizing  $\mathcal{L}_{\text{conf}}$  effectively pushes confidence of potential proposals to at least  $\tau$ , but does not spend additional optimization effort to push them beyond  $\tau$ . This mechanism makes the attack more efficient by distributing gradient resources to those boxes that still lie below the threshold, thereby maximizing the total count of high-confidence proposals entering NMS.

**Shape Regularization Loss.** Simply inflating confidence scores could yield proposals that are oddly shaped (excessively large) or overlap each other excessively. Consequently, these fabricated proposals can be efficiently pruned by sanity checks and NMS in CP postprocessing (Xu 2023), undermining the goal of maximizing latency. To address this, let  $M$  be the total number of 3D bounding box proposals and let each proposal be parameterized by  $\mathbf{b}_i = (x_j, y_j, z_j, \ell_j, w_j, h_j, \sigma_j)^2$ . We impose the following shape regularization loss  $\mathcal{L}_{\text{shape}} =$

<sup>1</sup> $B$  is the number of anchor boxes at each feature location.  $H, W$  are the height and width of the feature map, respectively.

<sup>2</sup> $(x_j, y_j, z_j)$  are the center coordinates of the proposal.  $\ell_j, w_j, h_j$  denote the length, width, and height of the proposal, and  $\sigma_j$  is the yaw.

$\frac{1}{M} \sum_{j=1}^M \mathbf{I}[\ell_j > L_{\max} \vee w_j > W_{\max}]$ , where  $\mathbf{I}[\cdot]$  is an indicator function, and  $L_{\max}, W_{\max}$  define upper bounds on length and width (e.g., 5m). By discouraging large boxes, we reduce the IoU between proposal pairs so that they are more sparsely distributed. As more proposals survive each iteration of NMS, the total number of NMS comparisons ( $\mathcal{O}(M^2)$ ) and the overall latency are inflated.

**Vertical Plausibility Loss.** To further maintain realism in 3D placement, we penalize bounding boxes whose vertical coordinate  $z_j$  is implausible (e.g., too high above the road or deep below ground). Let  $\mathcal{Z} = [z_{\min}, z_{\max}]$  be a valid elevation range (e.g., [1, 3] meters). We define  $\mathcal{L}_{\text{vertical}} = \frac{1}{M} \sum_{j=1}^M \mathbf{I}[z_j < z_{\min} \vee z_j > z_{\max}]$ . Penalizing out-of-range boxes deters trivial outliers that CP algorithm implementations typically reject by domain logic or height-thresholding (Xu 2023).

**Unified Loss Formulation.** We combine the above terms into a single objective that the attacker minimizes, with hyperparameters  $\lambda_1$  and  $\lambda_2$  balancing the relative emphasis  $\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{conf}} + \lambda_2 (\mathcal{L}_{\text{shape}} + \mathcal{L}_{\text{vertical}})$ . Minimizing this adversarial loss increases the number of high-confidence, plausible, and persistent proposals. Consequently, a large fraction of these proposals (1) survive the initial confidence-based filtering, (2) are difficult for standard sanity checks to block preemptively, and (3) cannot be efficiently eliminated during NMS.

### Spatial-Temporal Feature Warping

To perform online and real-time optimization for the adversarial perturbation to effectively increase the victim’s CP latency, the attack needs to account for delayed information in a realistic CAV environment. In other words, the attacker must generate perturbations for the current frame  $t$  with access only to the victim’s feature from  $t - 1$  due to asynchronous transmission. To overcome this challenge, we exploit the spatial-temporal consistency property and warping techniques of BEV features, where consecutive feature maps exhibit geometric shifts corresponding to ego-motion and can be approximated by transformation (Yu et al. 2024).

**Affine Transformation Derivation.** Let  $\mathbf{T}_v^{t-1 \rightarrow t} \in \mathbb{R}^{3 \times 3}$  be the 2D affine transform from the victim’s pose at  $t - 1$  to its pose at  $t$ . We derive  $\mathbf{T}_v^{t-1 \rightarrow t}$  by comparing the victim’s reported positions  $\mathbf{p}_v^{t-1}$  and  $\mathbf{p}_v^t$  through standard pose subtraction (Yu et al. 2024; Lu et al. 2023). Similarly, the adversary estimates  $\mathbf{T}_a^{t-1 \rightarrow t}$  for its own motion. These transformations are then normalized according to the BEV resolution, mapping physical distances to feature pixels (e.g., 0.4 m/grid cell).

**Warping and Online Optimization.** Given a prior feature map  $X_v^{t-1} \in \mathbb{R}^{C \times H \times W}$ , we warp it into the new frame  $t$  via  $\tilde{X}_v^t = \mathcal{W}(X_v^{t-1}, \mathbf{T}_v^{t-1 \rightarrow t})$ , where  $\mathcal{W}(\cdot, \cdot)$  denotes a spatial transformation that applies  $\mathbf{T}_v^{t-1 \rightarrow t}$  to align the feature map with the updated coordinate frame. The adversary performs the same operation for its own feature  $\mathbf{X}_a^{t-1}$ , resulting in  $\tilde{\mathbf{X}}_a^t$ , which is used to optimize the adversarial perturbation.

## Evaluation

### Experimental Setup

**Dataset and Algorithms.** We evaluate CP-FREEZER on the widely used benchmark CP dataset OPV2V (Xu et al. 2022), which is a large-scale V2V perception dataset, aligning with existing CP robustness work (Zhang et al. 2024; Wang et al. 2025). The dataset contains 11,464 LiDAR point cloud frames from 2-5 CAV agents and includes 232,913 annotated 3D vehicle bounding boxes. We regard the CAVs with the lowest and second-lowest indices as the victim and attacker vehicles, respectively. We assess CP-FREEZER on four SOTA CP algorithms: AttFusion (Xu et al. 2022), Where2comm (W2C) (Hu et al. 2022), CoAlign (Lu et al. 2023), and V2VAM (Li et al. 2023a), which are known for their high detection accuracy and real-time computational efficiency (Xu 2023).

**Comparison Baselines.** As the first CP latency attack, we compare CP-FREEZER against two baselines that are both optimized at the feature level for a fair comparison: (1) Projected Gradient Descent (PGD) (Madry et al. 2017), which though primarily aims to degrade accuracy untargetedly, often inflates the number of false positives as a side effect, thereby contributing to post-processing overhead. (2) Prior art in single-vehicle latency attacks proposed by SlowLiDAR (Liu et al. 2023) and SlowTrack (Ma et al. 2024). We use their loss function and adapt to CP with multi-agent input for comparison.

**Evaluation Metrics.** We measure the average end-to-end computation latency of the victim’s CP pipeline, both in benign and attacked scenarios (i.e.,  $\bar{T}_{\text{benign}}$  and  $\bar{T}_{\text{attack}}$ ). To measure the variability of the CP latency, we use the relative standard deviation (%RSD) defined as the ratio of the latency standard deviation over the average latency. A lower %RSD indicates that the measured latency values concentrate more consistently around the mean. Also, since the execution time is hardware-dependent, we calculate the Rate-of-Increase of pre-NMS proposal counts (RoI-P) and latency of the whole CP pipeline (RoI-L) defined as  $\text{RoI-L} = \frac{(T_{\text{attack}} - T_{\text{benign}})}{T_{\text{benign}}}$ , and  $\text{RoI-P} = \frac{(|\mathcal{P}|_{\text{attack}} - |\mathcal{P}|_{\text{benign}})}{|\mathcal{P}|_{\text{benign}}}$ . The RoIs provide hardware-independent measurements that capture how much slower the CP system runs and the increase in the number of bounding box proposals when attacked, relative to the benign scenario (Liu et al. 2023; Ma et al. 2024). Furthermore, to assess whether the increased latency poses a significant safety risk, we consider an attack as successful if it pushes the end-to-end CP latency beyond the threshold of 1.5 seconds, which is widely considered the worst-case time-to-collision threshold in autonomous driving literature (Wachenfeld et al. 2016; Kusano and Gabler 2011). The attack success rate (ASR) is the ratio of successful attacks to the total number of frames. Also, we show ASR under different latency thresholds in Figure 5.

**Testing Hardware.** We evaluate CP-FREEZER across three different hardware platforms: (1) RTX 2080 Super: a high-performance GPU that is equipped on our real-world vehicle testbed running a NUVO-8208GC computer. We emu-

Model	No Attack	PGD	Prior Art	CP-FREEZER
AttFusion	44.02	321.64 (12.28 $\times$ )	432.92 (13.26 $\times$ )	<b>1235.53</b> (46.19 $\times$ )
CoAlign	46.52	446.24 (12.93 $\times$ )	626.36 (17.41 $\times$ )	<b>1602.80</b> (54.84 $\times$ )
V2VAM	21.07	52.32 (3.70 $\times$ )	171.06 (9.58 $\times$ )	<b>340.46</b> (21.09 $\times$ )
Where2comm	19.44	5.58 (-0.62 $\times$ )	45.53 (2.02 $\times$ )	<b>506.20</b> (37.81 $\times$ )

\* Average RoI is computed per frame and then averaged. It differs from the RoI of  $|\mathcal{P}|$ .

Table 2: CP-FREEZER effectively inflates NMS workload. It has the highest average number of pre-NMS proposals  $|\mathcal{P}|$  and RoI in proposals RoI-P (parentheses).

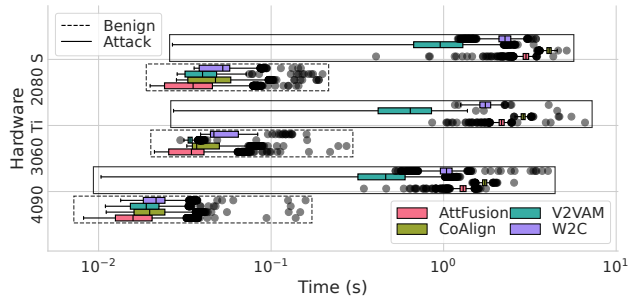


Figure 4: Latency distribution boxplot. Each CP algorithm mostly runs below 100 ms without attack. When CP-FREEZER is applied, the latency increases and concentrates around several seconds.

late sensor inputs and V2V communication by loading the dataset through the vehicle developer I/O. (2) RTX 3060 Ti: a more recent GPU with better performance-per-watt efficiency than RTX 2000 series. (3) RTX 4090: a top-tier GPU, representing a high-end system with ample computing power and memory bandwidth.

**Implementation Details.** We use the public implementations and pre-trained weights of the four CP algorithms released for OPV2V (Xu et al. 2022). For CP-FREEZER, we set  $\lambda_1 = 0.1$  and  $\lambda_2 = 1$ . We employ the BIM optimization scheme (Kurakin, Goodfellow, and Bengio 2018) with 10 gradient descent steps, each having a step size of 0.1. The same optimization parameters are applied to both baseline attacks. All model hyperparameters follow the official default configuration (Xu 2023).

## Evaluation Results

**Proposal Count Inflation.** Table 2 shows how CP-FREEZER sharply increases the average number of bounding box proposals  $|\mathcal{P}|$  across all four CP algorithms. In contrast, PGD and prior art induce only moderate or even negative ( $-0.62\times$ ) growth for certain models (e.g., W2C). For instance, PGD induces an average RoI-P of 12.28 $\times$ , and prior art of 13.26 $\times$ , whereas CP-FREEZER achieves a notably

higher 46.19 $\times$ . Similar proposal count increase patterns are observed on the other CP models.

**Latency across Hardware.** Figure 4 illustrates the distribution of the end-to-end latency of the victim’s CP pipeline across different hardware. Table 3 summarizes the latency of the victim’s CP pipeline and compares across different attack methods. We observe that PGD rarely exceeds a 1.3-2.0 $\times$  overhead on algorithms like V2VAM and W2C, sometimes failing to increase latency at all (e.g., W2C on RTX 2080 Super yields  $-0.2\times$ ). Prior art achieves moderate increases, but remains below the dramatic slowdowns caused by CP-FREEZER. Higher computing power, as shown by the RTX 4090, reduces the latency effect, yet CP-FREEZER consistently yields the highest RoIs in latency—often surpassing 80 $\times$ , with the lowest %RSD.

### Attack Success under Different Latency Thresholds.

Figure 5 shows the ASR as a function of the latency threshold, which represents different levels of real-time responsiveness required by the system. CP-FREEZER achieves higher ASRs across different thresholds and hardware than the prior art baseline. For example, on RTX 2080 Super, CP-FREEZER can push the latency of V2VAM beyond 1 s with around 40% ASR, on which the prior art can only introduce a latency less than 0.1 s with less than 20% ASR. Even at more strict thresholds around 2 s, CP-FREEZER maintains an ASR above 80–100% for the other models.

### Ablation Study

We further investigate the effectiveness of CP-FREEZER under more restrictive hyperparameter settings in CP post-processing. Specifically, we vary the confidence threshold (for filtering out low-confidence pre-NMS proposals), the IoU threshold in NMS, and the maximum number of proposals allowed into NMS. By default, these values are set to (0.2, 0.15, 1000) respectively (Xu 2023).

**Confidence Threshold.** Figure 6 shows that raising the confidence threshold only marginally reduces the RoI-P and RoI-L. Higher thresholds admit fewer borderline proposals, thereby reducing NMS overhead. However, even at higher thresholds (e.g., 0.35), CP-FREEZER still causes detection to be tens of times slower, indicating that simply tightening this threshold cannot neutralize the threat.

### IoU Threshold and Maximum Number of Proposals.

Raising the IoU threshold allows inflated boxes to persist longer, as fewer proposals are eliminated per iteration. As shown in Figure 7a, the similar slowdown effect at various IoU thresholds achieved by CP-FREEZER reflects the high persistence of the small and distributed proposals created. Figure 7b shows that capping the number of proposals below 1,000 partially alleviates the slowdown but still leaves a substantial latency increase (up to 20 $\times$ ).

**Spatial-Temporal Warping.** Table 4 highlights the impact of our spatial-temporal feature warping on the performance of CP-FREEZER. In most cases, warping significantly increases both RoI-P and RoI-L on RTX 4090, illustrating how aligning feature maps across frames, despite

Attack		No Attack		PGD			Prior Art			CP-FREEZER		
Hardware	Model	Avg	%RSD	ASR↑	Avg (RoI-L)	%RSD↓	ASR↑	Avg (RoI-L)	%RSD↓	ASR↑	Avg (RoI-L)	%RSD↓
RTX 2080 Super	AttFusion	0.039	53.85%	1.27%	0.806 (23.8×)	31.89%	18.82%	1.076 (26.9×)	50.37%	<b>99.61%</b>	<b>2.981 (89.4×)</b>	<b>7.98%</b>
	CoAlign	0.050	48.00%	50.98%	1.646 (33.5×)	37.85%	76.44%	2.364 (47.1×)	44.46%	<b>100%</b>	<b>4.078 (91.3×)</b>	<b>4.39%</b>
	V2VAM	0.042	38.10%	0.00%	0.090 (1.3×)	113.33%	0.00%	0.302 (6.0×)	55.30%	<b>15.26%</b>	<b>1.011 (22.8×)</b>	<b>48.86%</b>
	W2C	0.051	33.33%	0.00%	0.041 (-0.2×)	24.39%	0.00%	0.070 (0.3×)	55.71%	<b>96.93%</b>	<b>2.251 (46.5×)</b>	<b>13.95%</b>
RTX 3060 Ti	AttFusion	0.033	106.06%	0.78%	0.667 (22.2×)	61.02%	7.74%	0.797 (22.8×)	48.81%	<b>99.15%</b>	<b>2.164 (64.6×)</b>	<b>8.78%</b>
	CoAlign	0.037	127.03%	6.36%	1.043 (28.4×)	27.33%	38.83%	1.374 (36.7×)	42.94%	<b>100%</b>	<b>2.898 (69.7×)</b>	<b>5.24%</b>
	V2VAM	0.034	64.71%	0.00%	0.065 (1.0×)	90.77%	0.00%	0.149 (3.3×)	71.81%	<b>0.33%</b>	<b>0.636 (17.2×)</b>	<b>49.69%</b>
	W2C	0.045	62.22%	0.00%	0.035 (-0.2×)	5.71%	0.00%	0.044 (0.0×)	13.64%	<b>92.38%</b>	<b>1.754 (33.3×)</b>	<b>10.95%</b>
RTX 4090	AttFusion	0.018	72.22%	0.00%	0.382 (24.6×)	30.89%	0.00%	0.483 (27.2×)	49.48%	<b>1.07%</b>	<b>1.290 (81.1×)</b>	<b>8.06%</b>
	CoAlign	0.021	76.19%	1.07%	0.719 (34.9×)	35.88%	14.24%	1.000 (47.4×)	45.60%	<b>99.76%</b>	<b>1.728 (86.9×)</b>	<b>4.51%</b>
	V2VAM	0.023	69.57%	0.00%	0.035 (0.8×)	114.29%	0.00%	0.142 (6.0×)	58.45%	0.00%	<b>0.478 (23.5×)</b>	<b>47.70%</b>
	W2C	0.026	46.15%	0.00%	0.015 (-0.4×)	26.67%	0.00%	0.031 (0.2×)	80.65%	<b>0.59%</b>	<b>1.040 (48.3×)</b>	<b>19.62%</b>

Table 3: Results of average latency (s), relative standard deviation (%RSD), rate-of-increase in latency (RoI-L), and attack success rate (ASR) under 1.5 s threshold. CP-FREEZER significantly increases latency more reliably than PGD (Madry et al. 2017) and prior art (Liu et al. 2023; Ma et al. 2024).

Metric	RoI-P↑		RoI-L↑	
Model	w/o warp	w/ warp	w/o warp	w/ warp
AttFusion	26.14×	<b>46.19×</b>	63.91×	<b>81.10×</b>
CoAlign	30.19×	<b>54.84×</b>	<b>98.74×</b>	86.90×
V2VAM	9.97×	<b>21.09×</b>	6.35×	<b>23.50×</b>
W2C	3.47×	<b>37.81×</b>	0.70×	<b>48.30×</b>

Table 4: Spatial-temporal warping increases attack effectiveness with higher RoIs in both proposal counts and latency.

asynchronous knowledge, effectively creates more adversarial proposals, hence slowing down the victim’s CP system.

### Attack Robustness Against Defenses

**Adversarial Training.** We retrained models for five epochs while exposing them to CP-FREEZER. This defense was ineffective. For most models, it caused catastrophic collapse (0 mAP), destroying their utility. For V2VAM, which remained functional (0.89 mAP), CP-FREEZER still achieved a potent 18.7× RoI-L. This failure occurs because the objectives of robustness in integrity and computational efficiency are orthogonal. Training against latency attacks does not teach the model to be faster, but instead can corrupt its learned feature representations.

Metric	mAP@0.5↑		RoI-L↑	
Model	No Defense	ROBOSAC	No Defense	ROBOSAC
AttFusion	0	<b>0.71</b>	81.10×	<b>777.85×</b>
CoAlign	0	<b>0.72</b>	86.90×	<b>785.54×</b>
V2VAM	0	<b>0.77</b>	23.50×	<b>221.44×</b>
W2C	0	<b>0.74</b>	48.30×	<b>451.93×</b>

Table 5: ROBOSAC guards the integrity of cooperative perception, restoring mAP, yet amplifies the latency effect caused by CP-FREEZER.

**ROBOSAC.** We tested CP-FREEZER against ROBOSAC (Li et al. 2023b), a SOTA integrity defense that uses a RANSAC-like approach to identify malicious agents. As Table 5 shows, ROBOSAC restores detection integrity (mAP). However, its iterative design is fundamentally incompatible with defending against latency attacks. By repeatedly running the already-slowed perception pipeline, ROBOSAC inadvertently *amplifies* the latency, increasing the RoI-L from 81.1× to 777.8× for AttFusion. This demonstrates that reactive defenses designed for integrity can exacerbate availability threats.

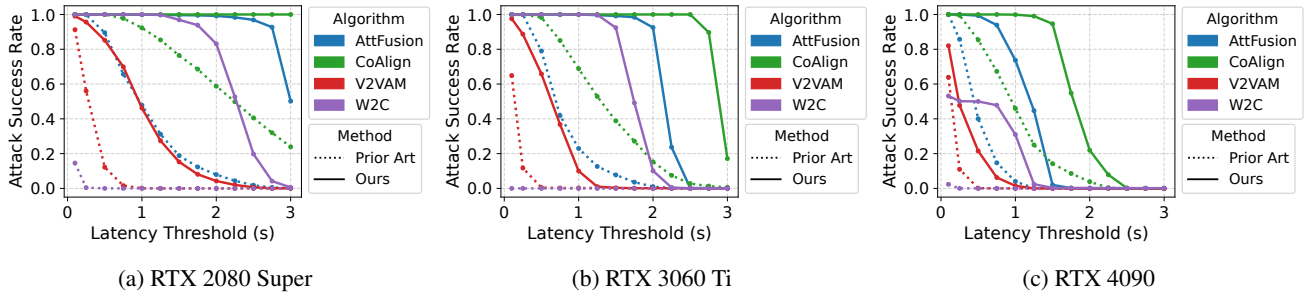


Figure 5: Attack success rates (ASRs) under varying latency thresholds, across different hardware. For each CP algorithm, CP-FREEZER achieves higher ASRs compared to the attack adopting the adversarial loss function used in the prior art on single-vehicle latency attacks.

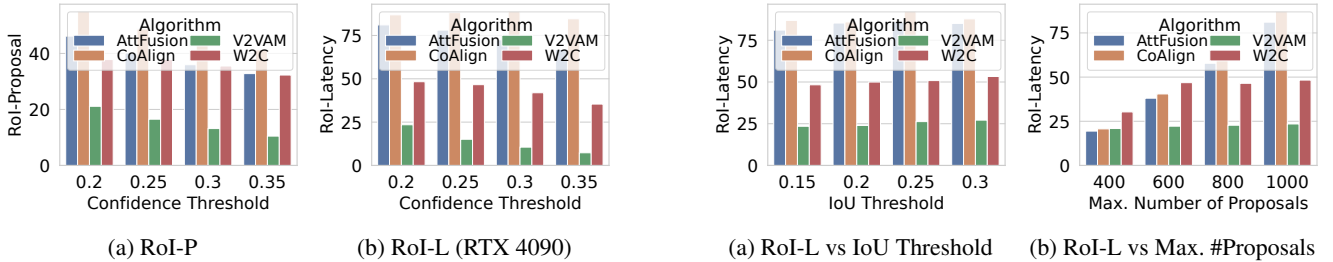


Figure 6: Attack effectiveness under different pre-NMS confidence thresholds. CP-FREEZER maintains high proposal count increase and slow-down effects for higher confidence thresholds.

Figure 7: RoI-L on RTX 4090 for different NMS hyper-parameters. CP-FREEZER achieves similar slowdown effects across different IoU thresholds, indicating high persistence of the fabricated proposals. Lowering the threshold for the maximum number of proposals can only partially alleviate the latency effect.

## Discussion

**Practical Impact.** Our work presents the first latency attack on CP, wherein a practical attacker can inflate the victim’s perception latency effectively by injecting adversarial perturbations into the shared messages. On our real-world vehicle testbed, the attacker can generate online perturbations in real-time with 50 ms per iteration, demonstrating its practical feasibility. We further confirm the applicability of CP-FREEZER on an NVIDIA Jetson Orin SoC platform. Although the public benchmark CP implementation (Xu 2023) runs more slowly on Orin at around 0.3 s per benign frame (potentially due to lack of official software hardware-co-design), CP-FREEZER nonetheless achieves similar Rate-of-Increase in latency, highlighting the hardware-agnostic nature of our approach. Since a single compromised CAV can degrade perception for multiple vehicles, our results highlight an urgent need for more robust CP system designs.

**Potential Defenses.** Current CP defenses (Li et al. 2023b; Zhang et al. 2024) require available final detection results to verify or repair, which makes them susceptible to CP-FREEZER. Tuning hyperparameters can reduce the number of proposals allowed and cap the worst-case scenario, but at the cost of a higher risk of overlooking true detections under benign cases and an inability to fully mitigate the slow-down effect. Therefore, more proactive defenses are needed to counter the availability threats posed by CP-FREEZER.

**Black-box Attack.** Our attack assumes white-box knowledge of the CP pipeline, in line with existing work (Liu et al. 2023; Ma et al. 2024). However, emerging heterogeneous CP systems (Lu et al. 2024) create a new system model where collaborators do not use the same CP algorithm. This translates into a black-box attack setting. As future work, we plan on investigating query-based black-box attacks, where the adversary only observes its own CP algorithm output and timing.

## Conclusion

We present CP-FREEZER, the first latency attack against vehicular cooperative perception (CP) by injecting adversarial perturbations via V2X messages, threatening the availability and timeliness guarantees of the safety-critical autonomous driving systems. Through extensive experiments on the benchmark dataset OPV2V with various SOTA CP algorithms and hardware platforms, CP-FREEZER shows slowdowns of  $90\times$  over benign operation, causing the single-frame latency to exceed 3 s with 100% success rates on our vehicle testbed. Our work highlights the need for new defenses to counter such availability threats.

## Acknowledgments

This work has been partially supported by the FY25 Eighteenth Mile TRIF Funding and a Thomas R. Brown Endowed Fellowship from the University of Arizona. This work was also supported by the National Science Foundation under grant numbers CNS-2144645, IIS-2229876, and in part by the National Center for Transportation Cybersecurity and Resiliency (TraCR), a U.S. Department of Transportation National University Transportation Center headquartered at Clemson University, Clemson, South Carolina, USA. Any opinions, findings, conclusions, and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of TraCR, and the U.S. Government assumes no liability for the contents or use thereof.

## References

- Abbasi, H. I.; Gholmieh, R.; Nguyen, T. V.; Patil, S.; and Misener, J. 2022. LTE-V2X (C-V2X) performance in congested highway scenarios. In *IEEE International Conference on Communications*.
- Chen, Q.; Ma, X.; Tang, S.; Guo, J.; Yang, Q.; and Fu, S. 2019a. F-cooper: feature based cooperative perception for autonomous vehicle edge computing system using 3D point clouds. In *ACM/IEEE Symposium on Edge Computing*.
- Chen, Q.; Tang, S.; Yang, Q.; and Fu, S. 2019b. Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. In *IEEE International Conference on Distributed Computing Systems (ICDCS)*.
- Chen, S.; Chen, H.; Haque, M.; Liu, C.; and Yang, W. 2023. The dark side of dynamic routing neural networks: Towards efficiency backdoor injection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Chen, Z.; Shi, Y.; and Jia, J. 2023. Transiff: An instance-level feature fusion framework for vehicle-infrastructure cooperative 3d detection with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Cui, J.; Qiu, H.; Chen, D.; Stone, P.; and Zhu, Y. 2022. Coopernaut: End-to-end driving with cooperative perception for networked vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Gao, K.; Bai, Y.; Gu, J.; Xia, S.-T.; Torr, P.; Li, Z.; and Liu, W. 2024. Inducing high energy-latency of large vision-language models with verbose images. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Hu, Y.; Fang, S.; Lei, Z.; Zhong, Y.; and Chen, S. 2022. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in Neural Information Processing Systems (NeurIPS)*.
- Kurakin, A.; Goodfellow, I. J.; and Bengio, S. 2018. Adversarial examples in the physical world. In *Artificial Intelligence Safety and Security*. Chapman and Hall/CRC.
- Kusano, K. D.; and Gabler, H. 2011. Method for estimating time to collision at braking in real-world, lead vehicle stopped rear-end crashes for use in pre-crash system design. *SAE International Journal of Passenger Cars-Mechanical Systems*, 4(2011-01-0576).
- Lang, A. H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; and Beijbom, O. 2019. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Li, J.; Xu, R.; Liu, X.; Ma, J.; Chi, Z.; Ma, J.; and Yu, H. 2023a. Learning for Vehicle-to-Vehicle Cooperative Perception Under Lossy Communication. *IEEE Transactions on Intelligent Vehicles*.
- Li, Y.; Fang, Q.; Bai, J.; Chen, S.; Juefei-Xu, F.; and Feng, C. 2023b. Among Us: Adversarially Robust Collaborative Perception by Consensus. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Lin, S.-C.; Zhang, Y.; Hsu, C.-H.; Skach, M.; Haque, M. E.; Tang, L.; and Mars, J. 2018. The Architectural Implications of Autonomous Driving: Constraints and Acceleration. In *International Conference on Architectural Support for Programming Languages and Operating Systems*. ACM.
- Lin, Z.; Xiao, L.; Chen, H.; Lv, Z.; Zhu, Y.; Zhang, Y.; and Liu, Y.-J. 2025. Edge-Assisted Collaborative Perception Against Jamming and Interference in Vehicular Networks. *IEEE Transactions on Wireless Communications*.
- Liu, H.; Wu, Y.; Yu, Z.; Vorobeychik, Y.; and Zhang, N. 2023. SlowLiDAR: Increasing the latency of lidar-based detection using adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Lu, Y.; Hu, Y.; Zhong, Y.; Wang, D.; Wang, Y.; and Chen, S. 2024. An Extensible Framework for Open Heterogeneous Collaborative Perception. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Lu, Y.; Li, Q.; Liu, B.; Dianati, M.; Feng, C.; Chen, S.; and Wang, Y. 2023. Robust collaborative 3d object detection in presence of pose errors. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*.
- Ma, C.; Wang, N.; Chen, Q. A.; and Shen, C. 2024. Slow-track: Increasing the latency of camera-based perception in autonomous driving using adversarial examples. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*.
- Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards Deep Learning Models Resistant to Adversarial Attacks. *CoRR*.
- Monteuuis, J.-P.; Petit, J.; Zhang, J.; Labiod, H.; Mafrika, S.; and Serval, A. 2018. Attacker model for connected and automated vehicles. In *ACM Computer Science in Car Symposium*.
- Muller, R.; Song, R.; Wang, C.; Zhan, Y.; Monteuuis, J.-P.; Man, Y.; Li, M.; Gerdes, R.; Petit, J.; and Celik, Z. B. 2025. Investigating physical latency attacks against camera-based perception. In *IEEE Symposium on Security and Privacy (SP)*.

Neubeck, A.; and Van Gool, L. 2006. Efficient Non-Maximum Suppression. In *International Conference on Pattern Recognition (ICPR)*.

Rauch, A.; Klanner, F.; Rasshofer, R.; and Dietmayer, K. 2012. Car2x-based perception in a high-level fusion architecture for cooperative perception systems. In *IEEE Intelligent Vehicles Symposium*.

Shapira, A.; Zolfi, A.; Demetrio, L.; Biggio, B.; and Shabtai, A. 2023. Phantom sponges: Exploiting non-maximum suppression to attack deep object detectors. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

Shumailov, I.; Zhao, Y.; Bates, D.; Papernot, N.; Mullins, R.; and Anderson, R. 2021. Sponge examples: Energy-latency attacks on neural networks. In *EuroS&P*. IEEE.

Tu, J.; Wang, T.; Wang, J.; Manivasagam, S.; Ren, M.; and Urtasun, R. 2021. Adversarial attacks on multi-agent communication. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Wachenfeld, W.; Junietz, P.; Wenzel, R.; and Winner, H. 2016. The worst-time-to-collision metric for situation identification. In *IEEE Intelligent Vehicles Symposium*.

Wang, C.; Muller, R.; Song, R.; Monteuiis, J.-P.; Petit, J.; Man, Y.; Gerdes, R.; Celik, Z. B.; and Li, M. 2025. From Threat to Trust: Exploiting Attention Mechanisms for Attacks and Defenses in Cooperative Perception. In *34th USENIX Security Symposium (USENIX Security 25)*.

Wang, J.; Li, A.; Li, H.; Lu, C.; and Zhang, N. 2022. RT-TEE: Real-time System Availability for Cyber-physical Systems using ARM TrustZone. In *IEEE Symposium on Security and Privacy (SP)*.

Xu, R. 2023. OpenCOOD. <https://github.com/DerrickXuNu/OpenCOOD>.

Xu, R.; Xiang, H.; Xia, X.; Han, X.; Li, J.; and Ma, J. 2022. OPV2V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*.

Yu, H.; Tang, Y.; Xie, E.; Mao, J.; Luo, P.; and Nie, Z. 2024. Flow-based feature fusion for vehicle-infrastructure cooperative 3d object detection. *Advances in Neural Information Processing Systems (NeurIPS)*.

Zhang, Q.; Jin, S.; Zhu, R.; Sun, J.; Zhang, X.; Chen, Q. A.; and Mao, Z. M. 2024. On data fabrication in collaborative vehicular perception: Attacks and countermeasures. In *33rd USENIX Security Symposium (USENIX Security 24)*.