SENSORAD: CROSS-MODAL PHYSICAL CONSISTENCY FOR SENSOR ATTACK DETECTION IN AUTONOMOUS VEHICLES

Anonymous authors
Paper under double-blind review

ABSTRACT

Autonomous vehicles fuse cameras, LiDAR, event cameras and GPS/IMU to perceive and act safely. Each of these sensors is individually attackable, but an adversary who compromises one cannot easily forge physically consistent signals across the rest. **SensorAD** turns this asymmetry into a detector: on normal driving data we mask an entire sensor modality and train a unified transformer to predict it from the remaining ones, so that the error minimized in training with a modality masked is the score read at deployment and no attack labels are used for training or threshold calibration. We pre-train on real-world nuScenes data, fine-tune on CARLA to incorporate the event camera, and evaluate on a 132-episode benchmark spanning ten attack types along six axes. Against nine baselines from five paradigms, SensorAD is strongest on physics-grounded attacks, averaging 0.843 AUROC versus 0.738 for the best cross-modal baseline; it localizes the compromised sensor at 67.0% top-1 / 79.5% top-2 accuracy, and its timing advantage over that baseline is specific to stealthy attacks: it flags a gradual GPS drift after $3.4_{\pm 0.4}$ m of accumulated offset against $12.6_{\pm 1.4}$ m. Detection degrades gradually under coordinated, spatially aligned and white-box adaptive attackers and costs at most 0.028 AUROC on unseen towns and weather. On real nuScenes streams with the same threat models injected, the CARLA-fine-tuned detector reaches 0.735 mean AUROC on the five expressible attacks (0.793 in CARLA), though its threshold must be recalibrated on real data. Per-sensor visual detectors remain stronger on weakly cross-modal attacks: the two families fail on largely disjoint attack classes, and a simple late-fusion stack outperforms either alone on average.

1 INTRODUCTION

Autonomous vehicles (AVs) depend on heterogeneous sensor suites to perceive their environment: RGB cameras, LiDAR, dynamic vision sensors (DVS), GPS, and inertial measurement units (IMU). Each sensor is individually vulnerable. GPS coordinates can be spoofed by commodity software-defined radios (Humphreys, 2012; Tippenhauer et al., 2011); LiDAR point clouds can be injected with phantom obstacles via synchronized laser pulses (Cao et al., 2019; Sun et al., 2020; Shin et al., 2017); cameras can be corrupted with adversarial patches or saturated by directed light (Boloor et al., 2020; Eykholt et al., 2018; Petit et al., 2015; Yan et al., 2016). Several have been demonstrated on commercial sensors or real vehicles with off-the-shelf hardware, so they are not a hypothetical threat.

Existing defenses largely follow a per-sensor paradigm. Anomaly detectors such as EfficientAD (Batzner et al., 2024), PatchCore (Roth et al., 2022), and PaDiM (Defard et al., 2021) are trained on normal data from one sensor stream and flag statistical deviations, as Sakhai et al. (2025) do with EfficientAD in CARLA. This paradigm has three limitations. (i) Some attacks remain locally plausible and are detectable only when compared with other modalities (a gradual GPS drift that stays within natural variance is the canonical example). (ii) Per-sensor detectors cannot observe cross-modal disagreement: a spoofed GPS position can be inconsistent with the IMU trajectory and visual scene, but no per-sensor detector sees this gap. (iii) Each sensor needs its own model, with no shared representation.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We argue that much of AV sensor security is a cross-modal problem. The intuition is simple: an attacker who compromises one sensor cannot, in general, produce physically consistent signals across the others. This is not a claim that real attacks are confined to one sensor: attacks rendering mutually consistent camera and LiDAR observations are documented (Tu et al., 2022; Cao et al., 2021). The claim is narrower, that forging consistency across physically separated data paths is harder than corrupting one of them, and we test it against coordinated, spatially aligned and white-box adaptive attackers in Section 4.4. A car in the camera image must also yield LiDAR returns at the correct depth, generate correlated DVS events at object boundaries, and stay consistent with the ego vehicle's GPS/IMU-derived motion. We treat this cross-modal physical consistency as a learnable invariant.

SensorAD (Sensor Attack Detection for autonomous driving) learns this invariant via masked pre-training: on normal driving data we mask an entire modality and train a unified transformer to predict its features from the others; at deployment we repeat this for each modality k and read the cross-modal prediction error as an anomaly score, which attacks on k inflate (Section 3.1).

Training such a model requires both real-world physical grounding and controlled attack injection. No single dataset offers both. Real driving data provides realistic sensor noise but has no DVS modality and no ground-truth attacks; high-fidelity simulators provide controlled attacks but lack real-world variability. We therefore use a three-phase real-to-sim pipeline: pre-train on real-world nuScenes (Caesar et al., 2020) (cameras, LiDAR, GPS/IMU), fine-tune on normal CARLA (Dosovitskiy et al., 2017) data with the DVS modality added, and evaluate on a held-out CARLA attack benchmark of 132 episodes covering ten attack types. Simulation gives exact attack onset, target and magnitude; we treat CARLA as a controlled benchmark rather than a substitute for deployment validation, and check transfer on real nuScenes streams (Sec. 4.5).

We make four contributions.

1. **Problem formulation.** We formalize AV sensor attack detection as cross-modal consistency verification, give a simple distribution-free separability condition (Proposition 1), and state the training-deployment correspondence that makes label-free training and calibration possible (Observation 1).
2. **Detection protocol.** A leave-one-modality-out detector that conditions each modality jointly on *all* remaining sensors, uses ego-motion physical invariants as prediction targets, and attributes the anomaly to a specific sensor. The components are standard; what is new is their combination, and that the full-modality-dropout term of the training objective is the same quantity read as the score at deployment.
3. **Benchmark.** 132 CARLA episodes over ten attack types, evaluated along six axes: detection, localization, delay, robustness to stronger attackers, false alarms under benign and environmental shift, and downstream damage.
4. **Empirical findings.** SensorAD leads on physics-grounded attacks (0.843 vs. 0.738 AUROC, significant under a paired per-episode test) and *loses* to per-sensor visual detectors on weakly cross-modal ones; the families fail on largely disjoint attack classes, and a simple late-fusion stack outperforms either alone on average (0.750 vs. 0.722 and 0.579). Modality masking earns almost all of its value on attacks that remain locally plausible (+0.200 AUROC on gradual drift, $\approx$0 on DoS, RGB, DVS and depth), and detection declines gradually as an adaptive attacker's budget grows (0.851 to 0.632 at twice the benchmark budget).

2 Related Work

AV Sensor Security. Beyond the GPS, LiDAR and camera attacks above, Shen et al. (2020) showed that fusion-based localization can drift by tens of meters under GPS spoofing even with LiDAR. Existing defenses are largely per-sensor anomaly detectors: Sakhai et al. (2025) train EfficientAD on benign data for each stream in CARLA. Attack-resilient sensor fusion (Ivanov et al., 2016) bounds the influence of a compromised subset of redundant sensors that measure the same variable, but does not model heterogeneous modalities or realistic sensor variability. Attacks can also preserve cross-modal consistency: Tu et al. (2022) and Cao et al. (2021) craft adversarial 3D objects that deceive camera and LiDAR at once, and the frustum attack of Hallyburton et al. (2022) spoofs LiDAR points inside a camera frustum so that camera and LiDAR semantics stay consistent. These are the hardest cases for any consistency-based defense; we treat them as explicit stress tests (Sec. 4.4) rather

than as excluded cases. To our knowledge, no prior work formulates AV sensor security as learned end-to-end cross-modal consistency verification.

Multi-Modal Anomaly Detection. Industrial inspection has driven much recent work on multi-modal anomaly detection. Crossmodal Feature Mapping (CFM; Costanzino et al., 2024) learns pairwise feature mappings between RGB and 3D for static objects, the closest prior work to ours; Li et al. (2025) add a third sensor with decision-level fusion; the Anomaly Transformer (Xu et al., 2022) uses association discrepancy for time series; and MambaAD (He et al., 2024), ADPretrain (Yao et al., 2025) and ResAD (Yao et al., 2024) study state-space models, anomaly-aware pre-training and residual features. None combines heterogeneous sensors with temporal dynamics: the industrial methods inspect static objects and the Anomaly Transformer models a single time series. Our setting differs: four heterogeneous modalities (images, point clouds, events, time series) with temporal dynamics, exploitable physics and safety-critical false-alarm requirements.

Self-Supervised Multi-Modal Pre-Training. MAE (He et al., 2022), BEiT (Bao et al., 2022), data2vec (Baevski et al., 2022) and CAV-MAE (Gong et al., 2023) mask patches within a modality (and, in the audio-visual case, occasionally across modalities) for representation learning; we use modality-level masking as the mechanism that aligns training with deployment for detection. BEV-Fusion (Liu et al., 2023), TransFusion (Bai et al., 2022) and UniAD (Hu et al., 2023) fuse sensors for better world prediction; we invert the goal and predict sensors from each other to detect when one is wrong. CAV-MAE learns shared representations rather than detection scores. CFM, like SensorAD, scores with the cross-modal mappings it trains on normal data, but predicts a modality from one other modality at a time; SensorAD conditions jointly on all remaining sensors, adds ego-motion targets, and attributes the anomaly to a sensor.

3 SensorAD: Cross-Modal Consistency Framework

3.1 Problem Formulation

Definition 1 (Multi-modal sensor observation)**.** *At time t, the vehicle observes M sensor modalities $\mathcal{S}_t = \{s_t^1, \ldots, s_t^M\}$, where each s_t^m lies in a modality-specific input space. We consider $M = 4$: RGB (four cameras; the co-located depth map is folded into the front-camera representation, Appendix G.4), LiDAR, DVS, and GPS/IMU.*

Definition 2 (Single-sensor attack)**.** *An attack on modality k replaces the true observation s_t^k with $\hat{s}_t^k = \mathcal{A}_k(s_t^k)$. We assume sensors are physically separated, so attacking k does not affect other modalities, which continue to reflect the true state.*

This assumption scopes the single-sensor detectability analysis below. It is not a claim about which attacks exist in the wild: coordinated, spatially aligned, and white-box adaptive multi-sensor attacks are evaluated in Section 4.4. RGB and DVS share an optical path in our simulated sensor rig and so do not satisfy it by construction, which is one reason those stress tests matter.

Threat model. The adversary controls the data path of one sensor (driver, ISP or upstream bus) and may replace its output arbitrarily in timing and magnitude, but cannot alter the physical scene or the other sensors' data paths, and has no access to the detector's weights or scores. Section 4.4 relaxes each assumption in turn (multi-sensor, spatially aligned and white-box attackers); adversarial physical scenes (Tu et al., 2022; Cao et al., 2021) lie outside this model.

Let $f_\theta^{k|\bar{k}}$ denote a cross-modal predictor of modality k from the remaining modalities. When trained on normal data, such a predictor incurs low error; under attack, the inconsistency between $\hat{s}_t^k$ and the unaffected modalities leads to elevated prediction error.

Proposition 1 (Detectability lower bound, in expectation)**.** *Let $\varepsilon_k^{\text{normal}} = \mathbb{E}\big[\,\big\|\tilde{z}^k - z^k\big\|^2\,\big]$ be the expected* squared *cross-modal prediction error on benign data, and let $\bar{\delta}_k = \big(\mathbb{E}\big[\,\big\|\hat{z}^k - z^k\big\|^2\,\big]\big)^{1/2}$ be the root-mean-square feature-space perturbation induced by an attack on modality k alone. Then*

$$\varepsilon_k^{\text{attack}} \geq \left(\bar{\delta}_k - \sqrt{\varepsilon_k^{\text{normal}}}\right)_+^2, \tag{1}$$

where $(x)_+ = \max(x, 0)$. *Consequently, any attack with* $\bar{\delta}_k > 2\sqrt{\varepsilon_k^{\text{normal}}}$ *yields an expected attack-time error strictly greater than the expected benign error, so the two are separable* in expectation. *This does not certify detection of any individual frame at a fixed threshold.*

The proof is one application of Minkowski's inequality in L_2 (Appendix A); it assumes only a normed space and an attack on modality k alone, and is correspondingly loose. We use it as a design principle rather than a guarantee: reducing normal-time cross-modal error lowers the perturbation size at which attacks become separable. Both quantities live in the learned feature space, so what matters is their ratio; shrinking the representation lowers both without improving separability. The magnitude sweep (Appendix D.4) is consistent with the bound without measuring it.

Observation 1 (Training-deployment correspondence)**.** *Define the* pathway *for modality* k *as the composite route (modality tokenizers* $\rightarrow$ *shared encoder* $\rightarrow$ *head* h_k*) that maps* $\mathcal{M}_k(\mathcal{S}_t) = \mathcal{S}_t \setminus \{s_t^k\}$ *to a prediction of* z_t^k*. The full-modality-dropout training term for modality* k *and the deployment anomaly score* $a_k(t)$ *are computed along this same pathway with the same weights. Training therefore needs no attack labels, and reducing normal-time error along a pathway is what tightens Proposition 1 for it.*

This is a design correspondence, not a theorem. Note where the shift enters: the masked pass sees `[MASK]` in place of modality k plus the uncompromised sensors, exactly its training input, so it is *not* shifted at deployment; the reference pass producing the target does see the corrupted tokens, which is why a score trained only on normal data can flag attacked frames at all.

3.2 Architecture

SensorAD processes multi-modal data in four stages (Figure 1): modality-specific tokenization, cross-modal masking, transformer encoding, and per-modality reconstruction with anomaly scoring.

Modality-Specific Tokenizers. Each modality is converted into a sequence of d-dimensional tokens ($d = 768$): a frozen DINOv2 ViT-B/14 (Oquab et al., 2024) for each of four camera views (with per-camera embeddings; the front view also carries the co-located depth map); a PointPillars-style encoder (Lang et al., 2019) producing a flattened BEV feature map; a CNN stem over polarity-separated DVS event volumes; and a linear projection of 11-dimensional GPS/IMU readings over a 50-step window with positional encoding. Each modality contributes a learnable `[CLS]` token and a modality-type embedding; full specifications are in Appendix F.

Cross-Modal Masking. The masking strategy links training to deployment. Unlike MAE-style approaches that mask patches within a modality (He et al., 2022), we mask whole modalities, forcing cross-modal reasoning to reconstruct the missing one. We stochastically sample five modes: full-modality dropout (one sensor removed), multi-modality dropout (two sensors removed), partial corruption (75% of tokens within one modality), temporal gap (all modalities masked over a window), and spatial masking, in which a contiguous region of the LiDAR BEV grid is masked together with the RGB and DVS patches whose camera rays intersect it under the known extrinsics. The region-correspondence procedure and ablations are in Appendix E.1; sampling probabilities are in Appendix F.1.

Unified Transformer. All tokens are concatenated and processed by an 8-layer pre-LayerNorm transformer ($d = 768$, 12 heads) with unrestricted cross-modal attention.

Anomaly Scoring. Per-modality reconstruction heads (2-layer MLPs) predict features at masked positions. At deployment, for each k we mask all tokens of modality k, encode the rest, and compute

$$a_k(t) = \frac{1}{N_k} \sum_{i=1}^{N_k} \left\| h_k(\mathbf{Z}_{t,i}^{k|\bar{k}}) - \text{sg}[h_k(\mathbf{Z}_{t,i}^{k})] \right\|^2, \tag{2}$$

where N_k is the number of tokens in modality k, h_k is the head for modality k, $\mathbf{Z}^{k|\bar{k}}$ are encoded representations when k is masked, $\mathbf{Z}^k$ when it is visible, and $\text{sg}[\cdot]$ is stop-gradient. The compromised sensor is localized as $k^* = \arg\max_k a_k(t)$, and the overall score is $A(t) = \text{mean}_k\, a_k(t)$. This is the quantity minimized by $\mathcal{L}_{\text{recon}}$ under full-modality dropout, one of the five training masking modes, which gives the correspondence in Observation 1. Reconstruction targets come from a separate all-visible reference pass with a stop-gradient, so prediction and target branches see different

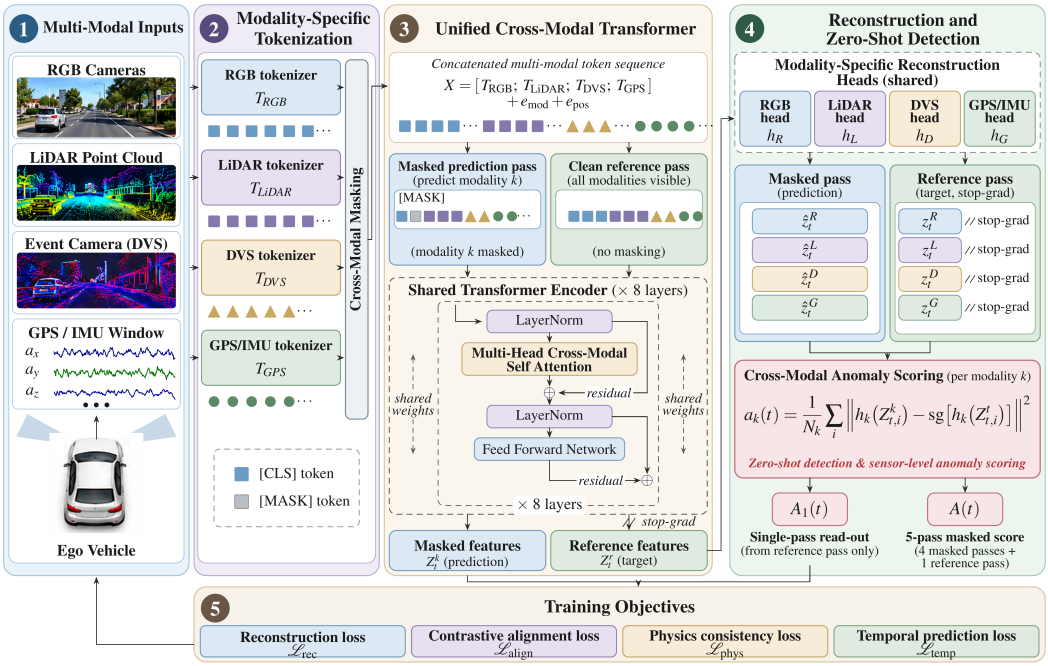


Figure 1: **SensorAD architecture for cross-modal attack detection in autonomous driving.** Multi-modal sensor streams (RGB, LiDAR, DVS, GPS/IMU) are tokenized by modality-specific encoders and fused by a shared 8-layer cross-modal transformer. The same weights serve two passes: the *masked prediction pass* predicts modality k from the rest, while the all-visible *reference pass* supplies the stop-gradiented target. That input asymmetry, with $\mathcal{L}_{\text{align}}$, guards against constant-output collapse in practice (Appendix F.2). Both read-outs are marked: $A_1(t)$ from the reference pass, and the five-pass $A(t)$, which also localizes.

inputs. This input asymmetry, together with the uniformity pressure of $\mathcal{L}_{\text{align}}$, is what avoids the constant-output solution in practice, although that solution remains a global minimizer of $\mathcal{L}_{\text{recon}}$ alone (Appendix F.2).

The same weights admit a cheaper read-out: scoring every head on the reference pass alone, with nothing masked, gives a single-pass global score $A_1(t)$ at negligible extra cost (0.5 ms), since that pass already runs to produce the targets. $A(t)$ is the primary detector and localized scorer; $A_1(t)$ is a cheaper, less accurate read-out (58.1% vs. 67.0% top-1). Section 4.3 measures what the extra passes buy.

3.3 Training Objectives

We use four objectives, activated in stages and combined with fixed weights chosen on a held-out selection pool.

Reconstruction ($\mathcal{L}_{\text{recon}}$). This is the primary objective: MSE between reconstruction-head outputs at masked positions and stop-gradients at unmasked positions. The loss is computed in head-output space to match Eq. (2).

Contrastive alignment ($\mathcal{L}_{\text{align}}$). An InfoNCE loss (van den Oord et al., 2018) aligning `[CLS]` tokens across modality pairs at the same timestep, encouraging a shared semantic space and stronger cross-modal prediction.

Physics consistency ($\mathcal{L}_{\text{phys}}$). Differentiable vehicle-dynamics constraints on the GPS/IMU head: predicted acceleration integrates to predicted velocity, predicted velocity integrates to predicted position, heading rate matches the gyroscope, and speed equals velocity magnitude. These relations are invariant to benign appearance shifts. Details are in Appendix F.2.

Temporal prediction ($\mathcal{L}_{\text{temp}}$). A small MLP maps current multi-modal `[CLS]` representations to next-frame ones, encouraging smooth dynamics.

The combined loss is

$$\mathcal{L}_{\text{total}} = \lambda_{\text{recon}}\mathcal{L}_{\text{recon}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{phys}}\mathcal{L}_{\text{phys}} + \lambda_{\text{temp}}\mathcal{L}_{\text{temp}}, \tag{3}$$

with $(\lambda_{\text{recon}}, \lambda_{\text{align}}, \lambda_{\text{phys}}, \lambda_{\text{temp}}) = (1.0, 0.2, 0.1, 0.1)$, chosen by grid search over each $\lambda \in \{0.05, 0.1, 0.2, 0.5, 1.0\}$ with the others fixed, on a separate pool of attack episodes disjoint from the reporting benchmark; model selection therefore uses attack labels, while training and threshold calibration do not (Appendices G.3 and E.3). The auxiliary terms shape the representation but are never scored. Losses are staged ($\mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{align}}$, then $\mathcal{L}_{\text{phys}}$ with warmup, then $\mathcal{L}_{\text{temp}}$), and fixed weights give lower seed variance than learned uncertainty weighting (Kendall et al., 2018).

3.4 Three-Phase Real-to-Sim Pipeline

No single dataset offers real-world physical grounding, controlled attack injection, and all four modalities, so we use a three-phase pipeline (Figure 6, Appendix F.1). **Phase 1** pre-trains on $\sim$34K real nuScenes keyframes (RGB, LiDAR, GPS/IMU) with DVS treated as permanently masked (its tokenizer exists but receives no gradients), so it can be added later without architectural change. **Phase 2** loads that checkpoint, adds a randomly initialized DVS tokenizer, freezes the shared transformer for two epochs, then fine-tunes on $\sim$200 normal CARLA episodes at a reduced learning rate. **Phase 3** freezes the model and scores attack episodes at a threshold calibrated on the 99th percentile of a held-out normal set.

4 Experiments

We evaluate SensorAD along the six axes of Section 1, together with component ablations and the cost of masking. All numbers are means over three seeds, with standard deviations where shown; per-seed values and paired per-episode significance tests are in Appendix B.1.

4.1 Experimental Setup

Datasets. Phase 1 uses nuScenes (Caesar et al., 2020) v1.0-trainval (850 scenes, 34,149 keyframes, 85/15 scene-level split, four cameras, 32-line LiDAR, 11-dim GPS/IMU derived from ego pose). Phase 2 uses $\sim$200 normal CARLA (Dosovitskiy et al., 2017) 0.9.15 episodes ($\sim$400K frames) over 4 towns and 3 weather conditions. Phase 3 uses 132 held-out attack episodes over the ten attack types of Table 26, each structured as 500 normal $\rightarrow$ 1000 attack $\rightarrow$ 500 normal frames (Appendix G).

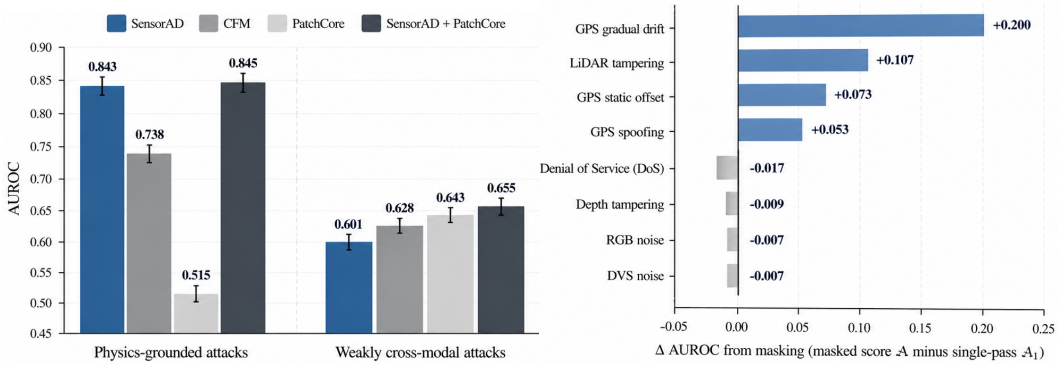
Baselines. Nine methods across five paradigms, all trained normal-only. **Per-sensor:** EfficientAD (Batzner et al., 2024), PatchCore (Roth et al., 2022), PaDiM (Defard et al., 2021), one model per modality. **Score fusion:** z-scored Max-Fusion of EfficientAD scores. **Joint statistical:** Mahalanobis and OC-SVM on pooled multi-modal features. **Temporal:** Anomaly Transformer (Xu et al., 2022) and an LSTM-AE (Malhotra et al., 2016) on the same pooled features, plus two *GPS/IMU-only* controls (static Mahalanobis and LSTM-AE) that isolate temporal modeling from cross-modal conditioning. **Cross-modal:** CFM (Costanzino et al., 2024) over all $\binom{4}{2} = 6$ modality pairs. Representation-based baselines use the same frozen tokenizers as SensorAD; they consume pooled per-modality features whereas SensorAD attends over dense tokens, a granularity asymmetry we control for in Appendix D.2.

Metrics. AUROC (primary) and FPR@95%TPR, per attack then averaged; Top-1/Top-2 localization; median detection delay over detected episodes; FPR under benign shift; and operating-point metrics at calibrated thresholds (Appendix D.8). Ablations report *overall* AUROC pooled over attack frames (0.737), which differs from the eight-attack mean (0.722).

Optimizer settings and schedules are in Appendix F.1.

4.2 Main Results: Attack Detection

Physics-grounded versus weakly cross-modal attacks. Table 1 and Figure 2 report AUROC over all eight attacks; training curves are stable and appear in Appendix E.4. We group attacks by mechanism,



(a) Detection by attack class.

(b) Per-attack gain from masking.

Figure 2: **When does cross-modal consistency help?** (a) SensorAD leads on physics-grounded attacks and trails PatchCore on weakly cross-modal ones; late fusion lifts the eight-attack average to 0.750. (b) Gain of the four masked passes over the single-pass read-out $A_1(t)$, which is why the pooled $+0.013$ understates masking. Mean over 3 seeds.

not outcome: physics-grounded attacks break a cross-sensor geometric or ego-motion relation (GPS spoofing and drift, LiDAR phantoms, DoS); weakly cross-modal ones leave those relations intact (appearance-only RGB, DVS and depth corruption, and a constant GPS offset that preserves every derivative). The result splits along this axis. **On the four physics-grounded attacks SensorAD averages 0.843 against 0.738 for CFM, the strongest cross-modal baseline**, a 10.5-point gap (paired Wilcoxon over 48 episodes, $p = 3.1\times10^{-9}$; 95% episode-bootstrap CI $[+0.083, +0.129]$); on the four weakly cross-modal ones it averages 0.601 and *loses* to PatchCore (0.643), since patch-level artifacts are locally detectable and carry little cross-modal signal. Because DoS is blatant and lifts the average, we also report the gap without it (the per-sensor DoS column additionally reflects Anomalib's normalization rather than detection; Appendix I): excluding DoS the gap is 6.2 points (0.799 vs. 0.737; $p = 3.9\times10^{-7}$, 36 episodes). The advantage is largest on GPS; the LiDAR margin overlaps within one seed standard deviation but is consistent across episodes ($p = 0.0068$, 12 episodes; Appendix B.1). Retrained on the pooled `[CLS]` inputs CFM consumes, SensorAD still leads at matched granularity, 0.802 against 0.738, with dense tokens adding 0.041 (Appendix D.2): roughly 60% of the margin is design (45% excluding DoS).

Complementarity with per-sensor detectors. The families fail on largely disjoint attack classes (both are weak on the static offset), and late fusion with the per-sensor PatchCore camera detector (PatchCore-Cam) recovers much of both: RGB rises to 0.716 and depth to 0.642, still below PatchCore alone (0.73, 0.68), physics-grounded performance is unchanged (0.843→0.845), and the eight-attack average rises to 0.750 (Appendix C.2).

4.3 Ablations, Sensor Localization, and Inference Cost

Component ablation. Table 4 (Appendix B) shows modality-level masking is the most important component, a 0.085 AUROC drop when removed, then the physics loss at 0.038. Contrastive and temporal losses give smaller but consistent gains, and doubling depth (with a wider FFN) adds only 0.002, within noise; since the frozen backbone and token budget are shared, this cannot separate a data limit from a backbone or token limit. Removing DVS costs 0.012, inside the seed spread, so DVS is not load-bearing here; its role is to show that the pipeline absorbs a structurally dissimilar modality without architectural change (pipeline ablation in Appendix B.2).

Sensor localization. SensorAD identifies the attacked sensor as $k^* = \arg\max_k a_k(t)$, at 67.0%/79.5% top-1/top-2 over the seven single-sensor attacks other than DoS (75.6%/85.3% excluding depth). GPS spoofing and drift localize reliably (over 90% top-1) and LiDAR tampering well (82.5%); the static GPS offset only moderately (55.5%). Camera-originated attacks are harder: RGB noise splits between camera and LiDAR, and depth tampering, which enters through the front camera, is attributed to RGB in only 16.0% of frames. That error is systematic rather than random: 68.4% of depth-attacked frames are attributed to LiDAR, the other geometric modality, consistent

Table 1: **Detection over the eight primary (non-adaptive) attack settings (AUROC ↑, FPR@95 ↓).** Left block: physics-grounded, where the attack breaks a cross-sensor geometric or ego-motion relation; right block: weakly cross-modal, where those relations stay intact (Sec. 4.2). DoS zeroes both RGB and LiDAR; depth tampering corrupts the co-located depth map, which enters through the front-camera representation (Appendix G.2). Coordinated and adaptive settings are in Appendices B.3 and D.5. Mean over 3 seeds, std in subscript. †Overlaps CFM within one seed std; the paired per-episode difference is significant ($p = 0.0068$; Table 3).

		Physics-grounded				Weakly cross-modal					
Paradigm	**Method**	**GPS**	**GPS-grad**	**LiDAR**	**DoS**	**RGB**	**DVS**	**Static**	**Depth**	**Avg.**	**FPR@95**
Per-sensor	EfficientAD	$0.52_{\pm 0.02}$	$0.51_{\pm 0.02}$	$0.54_{\pm 0.02}$	$0.51_{\pm 0.01}$	$0.66_{\pm 0.03}$	$0.57_{\pm 0.03}$	$0.53_{\pm 0.02}$	$0.60_{\pm 0.03}$	$0.555_{\pm 0.02}$	$0.96_{\pm 0.02}$
	PatchCore	$0.51_{\pm 0.02}$	$0.52_{\pm 0.02}$	$0.53_{\pm 0.02}$	$0.50_{\pm 0.01}$	**$0.73_{\pm 0.03}$**	$0.64_{\pm 0.03}$	$0.52_{\pm 0.02}$	**$0.68_{\pm 0.02}$**	$0.579_{\pm 0.02}$	$0.94_{\pm 0.02}$
	PaDiM	$0.53_{\pm 0.02}$	$0.50_{\pm 0.02}$	$0.52_{\pm 0.02}$	$0.51_{\pm 0.01}$	$0.69_{\pm 0.03}$	$0.61_{\pm 0.03}$	$0.51_{\pm 0.02}$	$0.65_{\pm 0.03}$	$0.565_{\pm 0.02}$	$0.95_{\pm 0.02}$
Score fusion	Max-Fusion	$0.57_{\pm 0.02}$	$0.56_{\pm 0.02}$	$0.61_{\pm 0.03}$	$0.59_{\pm 0.02}$	$0.71_{\pm 0.03}$	**$0.66_{\pm 0.03}$**	$0.57_{\pm 0.02}$	$0.67_{\pm 0.03}$	$0.618_{\pm 0.02}$	$0.93_{\pm 0.02}$
Joint stat.	Mahalanobis	$0.68_{\pm 0.02}$	$0.71_{\pm 0.02}$	$0.67_{\pm 0.02}$	$0.73_{\pm 0.02}$	$0.62_{\pm 0.03}$	$0.61_{\pm 0.03}$	$0.64_{\pm 0.03}$	$0.61_{\pm 0.03}$	$0.659_{\pm 0.02}$	$0.90_{\pm 0.02}$
	OC-SVM	$0.65_{\pm 0.03}$	$0.67_{\pm 0.03}$	$0.64_{\pm 0.03}$	$0.69_{\pm 0.03}$	$0.60_{\pm 0.03}$	$0.59_{\pm 0.03}$	$0.61_{\pm 0.03}$	$0.59_{\pm 0.02}$	$0.630_{\pm 0.02}$	$0.92_{\pm 0.02}$
Temporal	Anomaly Transf.	$0.43_{\pm 0.04}$	$0.46_{\pm 0.04}$	$0.42_{\pm 0.03}$	$0.45_{\pm 0.03}$	$0.41_{\pm 0.03}$	$0.39_{\pm 0.04}$	$0.41_{\pm 0.03}$	$0.40_{\pm 0.03}$	$0.421_{\pm 0.03}$	$1.00_{\pm 0.00}$
	LSTM-AE	$0.72_{\pm 0.03}$	$0.76_{\pm 0.02}$	$0.65_{\pm 0.03}$	$0.77_{\pm 0.02}$	$0.61_{\pm 0.03}$	$0.60_{\pm 0.03}$	$0.63_{\pm 0.03}$	$0.59_{\pm 0.03}$	$0.666_{\pm 0.02}$	$0.92_{\pm 0.02}$
Cross-modal	CFM	$0.75_{\pm 0.02}$	$0.78_{\pm 0.02}$	$0.68_{\pm 0.02}$	$0.74_{\pm 0.02}$	$0.63_{\pm 0.03}$	$0.62_{\pm 0.03}$	**$0.66_{\pm 0.02}$**	$0.60_{\pm 0.03}$	$0.682_{\pm 0.02}$	$0.91_{\pm 0.02}$
Ours	**SensorAD**	**$0.823_{\pm 0.056}$**	**$0.851_{\pm 0.048}$**	**$0.722_{\pm 0.070}$**†	**$0.974_{\pm 0.018}$**	$0.596_{\pm 0.033}$	$0.644_{\pm 0.056}$	$0.593_{\pm 0.043}$	$0.571_{\pm 0.037}$	**$0.722_{\pm 0.042}$**	**$0.882_{\pm 0.017}$**

with a corrupted depth cue surfacing mainly as disagreement with LiDAR geometry (Table 5). This weakness on camera attacks also motivates the late-fusion ensemble (Appendix C.2).

Temporal modeling versus cross-modal conditioning. The GPS margin could come from the 50-step GPS/IMU window rather than from cross-modal conditioning; matched GPS/IMU-only baselines separate the two (Appendix D.1). Temporal modeling alone explains much of the abrupt-GPS gain (static Mahalanobis 0.651 to GPS-only LSTM-AE 0.742); cross-modal conditioning explains the rest (0.742 to 0.823) and is decisive on gradual drift, where SensorAD fires at 3.4 m against 15.1 m, because a unimodal model cannot flag a smooth ramp until the trajectory becomes self-inconsistent. The GPS-only LSTM-AE is stronger on the static GPS offset (0.646 vs. 0.593): a constant offset translates the trajectory without changing its shape, so it cancels exactly in every physics term and only absolute position is wrong (Appendix F.2).

What masking buys, and what it costs. Relative to the single-pass read-out $A_1(t)$, the four extra masked passes cost 61 ms and buy 0.013 pooled AUROC (0.724 → 0.737), 0.049 averaged over the eight attacks, and 8.9 points of top-1 localization (58.1 → 67.0%; Appendix D.3). The pooled figure conceals where masking works (Figure 2b): it adds $+0.200$ on gradual GPS drift, $+0.107$ on LiDAR tampering, $+0.073$ on the static GPS offset and $+0.053$ on GPS spoofing, and ≈ 0 on DoS, RGB, DVS and depth, because a sensor that still looks locally plausible partly reconstructs *itself* on the all-visible pass and only masking removes that self-information. Five passes cost $2.4\times$ a single pass rather than $5\times$, since the tokenizers run once and the masked passes batch into one call. We make no real-time claim: $A(t)$ at 9.4 FPS is not a per-frame detector here, whereas $A_1(t)$ at 21.8 FPS fits a 20 Hz budget and can run continuously with attribution escalated only on flagged frames, in which case the cascade detects at 0.724, not 0.737.

4.4 Detection Delay and Robustness to Stronger Attackers

Detection delay. On abrupt attacks SensorAD fires within a median of $9.4_{\pm 1.1}$ frames (≈470 ms at 20 Hz), ahead of the temporal and per-sensor baselines (15.4 and 18.0) but indistinguishable from CFM ($9.7_{\pm 1.2}$). The gap appears where the attack is stealthy: on gradual GPS drift SensorAD fires at $3.4_{\pm 0.4}$ m of accumulated offset against $12.6_{\pm 1.4}$ m for CFM, roughly 70% less displacement before the alarm (Appendix B.4).

Coordinated and spatially aligned attacks. Detection degrades gracefully as more sensors are compromised: GPS spoofing plus the static offset is on par with GPS alone ($0.842_{\pm 0.043}$ vs. $0.823_{\pm 0.056}$), while 2-of-4 and 3-of-4 fall to 0.717 and 0.641. In the harder aligned case, related to the frustum attack of Hallyburton et al. (2022), an RGB patch is perturbed jointly with the LiDAR points in its 3D frustum; detection falls to 0.653 but stays above CFM (0.611), consistent with the intact GPS/IMU and DVS pathways still constraining the reconstruction (Appendix B.3). Fully consistent multi-sensor injection (Tu et al., 2022; Cao et al., 2021) remains open.

White-box adaptive attackers. Our threat model (Section 3.1) excludes white-box access to the detector, but the informative question is how the method fails when that assumption is violated. Given the frozen detector, an attacker parameterizes the spoofed GPS trajectory by onset, ramp rate and curvature and minimizes SensorAD's peak anomaly score by projected gradient subject to achieving a target lateral displacement; budgets are nested, so a larger budget is a strictly stronger attacker. AUROC falls from 0.851 (black-box) to 0.792, 0.711 and 0.632 at $0.5\times$, $1\times$ and $2\times$ (Appendix D.5). Low-budget attacks remain detectable, but the detector is materially weakened at larger budgets and we claim no certified robustness. Detection rises monotonically with attack magnitude and falls toward chance at $0.25\times$ (Appendix D.4), telling a deployer the regime it is blind to.

4.5 False Alarms and Generalization

Benign shift and unseen environments. At a calibrated 99th-percentile normal threshold SensorAD has the lowest false-alarm rate of the four methods compared, $1.4_{\pm 0.3}\%$ on normal driving against 3.8% (CFM) and 4.9% (Max-Fusion). Shift still hurts, raising FPR to $8.7_{\pm 1.1}\%$ under weather change and $10.9_{\pm 1.4}\%$ day→night, so environment-conditioned thresholds are needed (Appendix D.9). Re-running Phase 2 from scratch with an entire town or weather condition held out gives 0.719 (Town10HD), 0.719 (Town01) and 0.711 (WetNoon), at most 0.028 below the all-environment model (Appendix D.6).

Operating points and real data. Under split-conformal calibration the realized FPR tracks the target; at $\alpha{=}0.05$ SensorAD reaches precision 0.921, recall 0.612 and F1 0.735 (Appendix D.8). Injecting the benchmark threat models on real nuScenes streams, for the five attacks nuScenes can express, gives a mean AUROC of 0.735 for the CARLA-fine-tuned model (0.755 for the nuScenes-pretrained one) against 0.793 in CARLA, but the CARLA-calibrated threshold does not transfer: on held-out real normal driving the false-alarm rate rises to 5.6% (2.8% for the pretrained model) against a 1% target, so real-data recalibration is required (Appendix C.3). This is a transfer check, not validation against physical attacks.

4.6 Downstream Impact of the Attacks

Task-level cost of each attack. Detection scores matter only if the attacks do damage, so we measure each attack's effect on its target task (Appendix D.7): GPS spoofing shifts ego-pose by 18.3 m, DoS removes every usable RGB and LiDAR frame, LiDAR tampering costs 8.7 3D mAP and 6.9 NDS, depth tampering adds 0.53 m MAE, and RGB noise costs 3.3 mAP. Detectability does not track damage monotonically: an *optimized* adversarial camera perturbation costs 12.4 mAP, more than LiDAR tampering, yet is detected less reliably (0.674 vs. 0.722). It is still detected *better* than random noise (0.674 vs. 0.596), plausibly because a perturbation tuned to corrupt the image induces larger cross-modal inconsistency than unstructured noise does.

5 Conclusion

We formulated AV sensor attack detection as cross-modal physical consistency verification and introduced SensorAD, a self-supervised detector that learns this consistency via leave-one-modality-out masked prediction, with the same pathway serving training and deployment, joint conditioning on *all* remaining sensors, and ego-motion invariants as prediction targets. It is the strongest detector on physics-grounded attacks (0.843 vs. 0.738 AUROC), localizes the compromised sensor at 67.0%/79.5% top-1/top-2, and flags a gradual GPS drift after 3.4 m against 12.6 m for the strongest cross-modal baseline. The headline takeaway is structural: **cross-modal consistency is best for physics-grounded attacks and per-sensor visual detectors for localized appearance artifacts; they fail on largely disjoint attack classes, so a simple late-fusion stack outperforms either alone on average (Appendix C.2).** We therefore recommend the fused stack as the safety monitor.

Scope and limitations. The ego-motion targets constrain derivatives, not absolute pose, so a unimodal baseline wins on the static GPS offset; constant rotations and slow affine drifts share that structure and are untested. Thresholds calibrated in simulation do not transfer to real data without recalibration. The five-pass score is an attribution mode, not a real-time detector on this hardware. AUROC falls to 0.632 under our strongest white-box attack, and a fully joint consistency-preserving attacker (Tu et al., 2022; Cao et al., 2021) remains untested.

REPRODUCIBILITY STATEMENT

All experiments use public datasets and simulators and a single consumer GPU. An anonymized code package (`https://anonymous.4open.science/r/SensorAD-AC07/README.md`) contains the pre-training, fine-tuning, attack-generation, calibration and evaluation scripts, configuration files and random seeds for every experiment reported here, and data-preprocessing instructions. The architecture is specified in Section 3.2 and Appendix F; optimizer settings, loss weights and staging schedules in Appendix F.1; the attack benchmark, its composition and its injection parameters in Appendix G; baseline implementations in Appendix I; the calibration protocol in Appendix C.1; and compute requirements in Appendix J. SensorAD is trained exclusively on normal driving data (nuScenes in Phase 1, normal CARLA episodes in Phase 2), and attack labels are never used for training or threshold calibration. Architecture and loss weights were selected on a pool of attack episodes disjoint from the reporting benchmark (Appendix G.3); that step does use attack labels. We will release the repository and the benchmark-generation scripts publicly after the review process.

ETHICS STATEMENT

This work is defensive in intent: it detects compromised, degraded or faulty sensors before they affect downstream perception, localization and planning. It involves no human subjects and no personally identifying data, and nuScenes, CARLA and DINOv2 are used under their stated licenses (Appendix G). We acknowledge a dual-use consideration: reporting which attack classes are hardest to detect also indicates where such a defense is weakest. We mitigate this by evaluating only in simulation and on injected perturbations of public data, by making no deployment claims, and by stating plainly that SensorAD is not a certified defense against white-box adaptive or physically realized attacks. A real deployment would require closed-course testing, hardware-in-the-loop validation, environment-conditioned threshold calibration, fail-safe integration with planning, and human oversight; we discuss this further in Appendix G.5.

AI USE STATEMENT

We used large language models in two ways. First, as a writing aid for copy-editing, LaTeX formatting and improving the clarity of prose in this manuscript. Second, as a coding assistant for parts of the research code, including implementation and debugging support. We did not use generative AI tools to generate the research idea, to design the method or the experimental protocol, or to produce or analyse results. All AI-assisted text and code were reviewed, tested and edited by the authors, and we take full responsibility for the final content of this work, including its text, claims, code and artifacts.

REFERENCES

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A general framework for self-supervised learning in speech, vision and language. In *International Conference on Machine Learning (ICML)*, pp. 1298–1312. PMLR, 2022.

Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. TransFusion: Robust LiDAR-camera fusion for 3D object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1090–1099, 2022.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *International Conference on Learning Representations (ICLR)*, 2022.

Kilian Batzner, Lars Heckler, and Rebecca König. EfficientAD: Accurate visual anomaly detection at millisecond-level latencies. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 128–138, 2024.

Adith Boloor, Karthik Garimella, Xin He, Christopher Gill, Yevgeniy Vorobeychik, and Xuan Zhang. Attacking vision-based perception in end-to-end autonomous driving models. *Journal of Systems Architecture*, 110: 101766, 2020.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11621–11631, 2020.

Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z. Morley Mao. Adversarial sensor attack on LiDAR-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 2267–2281, 2019.

Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. Invisible for both camera and LiDAR: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 176–194, 2021.

Alex Costanzino, Pierluigi Zama Ramirez, Giuseppe Lisanti, and Luigi Di Stefano. Multimodal industrial anomaly detection by crossmodal feature mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17234–17243, 2024.

Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. PaDiM: A patch distribution modeling framework for anomaly detection and localization. In *Pattern Recognition. ICPR International Workshops and Challenges*, pp. 475–489. Springer, 2021.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Conference on Robot Learning (CoRL)*, pp. 1–16. PMLR, 2017.

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1625–1634, 2018.

Yuan Gong, Andrew Rouditchenko, Alexander H. Liu, David Harwath, Leonid Karlinsky, Hilde Kuehne, and James Glass. Contrastive audio-visual masked autoencoder. In *International Conference on Learning Representations (ICLR)*, 2023.

R. Spencer Hallyburton, Yupei Liu, Yulong Cao, Z. Morley Mao, and Miroslav Pajic. Security analysis of Camera-LiDAR fusion against Black-Box attacks on autonomous vehicles. In *31st USENIX Security Symposium (USENIX Security 22)*, pp. 1903–1920, 2022.

Haoyang He, Yuhu Bai, Jiangning Zhang, Qingdong He, Hongxu Chen, Zhenye Gan, Chengjie Wang, Xiangtai Li, Guanzhong Tian, and Lei Xie. MambaAD: Exploring state space models for multi-class unsupervised anomaly detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, 2022.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17853–17862, 2023.

Todd E. Humphreys. Statement on the vulnerability of civil unmanned aerial vehicles and other systems to civil GPS spoofing. Testimony before the Subcommittee on Oversight, Investigations, and Management, Committee on Homeland Security, U.S. House of Representatives, July 2012.

Radoslav Ivanov, Miroslav Pajic, and Insup Lee. Attack-resilient sensor fusion for safety-critical cyber-physical systems. *ACM Transactions on Embedded Computing Systems*, 15(1):21:1–21:24, 2016.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7482–7491, 2018.

Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. PointPillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12697–12705, 2019.

Wenqiao Li, Bozhong Zheng, Xiaohao Xu, Jinye Gan, Fading Lu, Xiang Li, Na Ni, Zheng Tian, Xiaonan Huang, Shenghua Gao, and Yingna Wu. Multi-sensor object anomaly detection: Unifying appearance, geometry, and internal properties. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9984–9993, 2025.

Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L. Rus, and Song Han. BEV-Fusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2774–2781, 2023.

Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint arXiv:1607.00148*, 2016.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Jonathan Petit, Bas Stottelaar, Michael Feiri, and Frank Kargl. Remote attacks on automated vehicles sensors: Experiments on camera and LiDAR. Black Hat Europe, 2015.

Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14318–14328, 2022.

Mustafa Sakhai, Kaung Sithu, Min Khant Soe Oke, and Maciej Wielgosz. Cyberattack resilience of autonomous vehicle sensor systems: Evaluating RGB vs. dynamic vision sensors in CARLA. *Applied Sciences*, 15(13): 7493, 2025.

Junjie Shen, Jun Yeon Won, Zeyuan Chen, and Qi Alfred Chen. Drift with devil: Security of Multi-Sensor fusion based localization in High-Level autonomous driving under GPS spoofing. In *29th USENIX Security Symposium (USENIX Security 20)*, pp. 931–948, 2020.

Hocheol Shin, Dohyun Kim, Yujin Kwon, and Yongdae Kim. Illusion and dazzle: Adversarial optical channel exploits against lidars for automotive applications. In *Cryptographic Hardware and Embedded Systems (CHES)*, pp. 445–467. Springer, 2017.

Jiachen Sun, Yulong Cao, Qi Alfred Chen, and Z. Morley Mao. Towards robust LiDAR-based perception in autonomous driving: General black-box adversarial sensor attack and countermeasures. In *29th USENIX Security Symposium (USENIX Security 20)*, pp. 877–894, 2020.

Nils Ole Tippenhauer, Christina Pöpper, Kasper Bonne Rasmussen, and Srdjan Capkun. On the requirements for successful GPS spoofing attacks. In *Proceedings of the 18th ACM Conference on Computer and Communications Security (CCS)*, pp. 75–86, 2011.

James Tu, Huichen Li, Xinchen Yan, Mengye Ren, Yun Chen, Ming Liang, Eilyan Bitar, Ersin Yumer, and Raquel Urtasun. Exploring adversarial robustness of multi-sensor perception systems in self driving. In *Proceedings of the 5th Conference on Robot Learning (CoRL)*, volume 164 of *Proceedings of Machine Learning Research*, pp. 1013–1024. PMLR, 2022.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. In *International Conference on Learning Representations (ICLR)*, 2022.

Chen Yan, Wenyuan Xu, and Jianhao Liu. Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle. DEF CON 24, 2016.

Xincheng Yao, Zixin Chen, Chao Gao, Guangtao Zhai, and Chongyang Zhang. ResAD: A simple framework for class generalizable anomaly detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Xincheng Yao, Yan Luo, Zefeng Qian, and Chongyang Zhang. ADPretrain: Advancing industrial anomaly detection via anomaly representation pretraining. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025.

A PROOFS

A.1 PROOF OF PROPOSITION 1

Assumptions. The proof requires only that (i) features lie in a normed vector space, and (ii) the attack acts on modality k alone, so the conditioning sensors $\{z^j\}_{j\neq k}$ retain their clean values (Definition 2). An inner product is *not* required: the reverse triangle inequality holds in any normed space. We do not assume independence across modalities, Gaussianity, calibration of the predictor, or any particular form of $\mathcal{A}_k$. The bound is therefore distribution-free at the cost of being conservative.

Setup. Let z^k be the clean feature representation of modality k and $\hat{z}^k$ the attacked one. Let

$$\tilde{z}^k = f_\theta^{k|\bar{k}}(\{z^j\}_{j\neq k}) \tag{4}$$

be the prediction of modality k from the remaining modalities. Define

$$\varepsilon_k^{\text{normal}} = \mathbb{E}\big[\|\tilde{z}^k - z^k\|^2\big], \qquad \varepsilon_k^{\text{attack}} = \mathbb{E}\big[\|\tilde{z}^k - \hat{z}^k\|^2\big], \qquad \bar{\delta}_k = \big(\mathbb{E}\big[\|\hat{z}^k - z^k\|^2\big]\big)^{1/2}. \tag{5}$$

Note that $\varepsilon_k^{\text{normal}}$ is an expected *squared* error, and that $\bar{\delta}_k$ is the root-mean-square perturbation, not a per-sample constant; treating the perturbation as deterministic would be incorrect, since $\mathcal{A}_k$ acts on samples drawn from the data distribution.

Argument. Write $X = \hat{z}^k - z^k$ and $Y = \tilde{z}^k - z^k$, both random vectors in the same normed space, and let $\|\cdot\|_{L_2}$ denote the L_2 norm of a random vector, $\|X\|_{L_2} = (\mathbb{E}[\|X\|^2])^{1/2}$. Since $\tilde{z}^k - \hat{z}^k = Y - X$, Minkowski's inequality for $\|\cdot\|_{L_2}$ gives the reverse form

$$\sqrt{\varepsilon_k^{\text{attack}}} = \|Y - X\|_{L_2} \;\geq\; \big|\, \|X\|_{L_2} - \|Y\|_{L_2} \,\big| = \big|\, \bar{\delta}_k - \sqrt{\varepsilon_k^{\text{normal}}} \,\big| \;\geq\; \left(\bar{\delta}_k - \sqrt{\varepsilon_k^{\text{normal}}}\right)_+ . \tag{6}$$

Both sides of the final expression are non-negative, so squaring is valid and yields

$$\varepsilon_k^{\text{attack}} \geq \left(\bar{\delta}_k - \sqrt{\varepsilon_k^{\text{normal}}}\right)_+^2 . \tag{7}$$

Working directly in L_2 avoids two pitfalls of a pointwise reverse-triangle argument followed by Jensen's inequality: squaring a bound whose right-hand side may be negative, and treating $\bar{\delta}_k$ as a deterministic constant when it is a property of the attack distribution.

Separation condition. A sufficient condition for the expected attack-time error to exceed the expected benign error is $\left(\bar{\delta}_k - \sqrt{\varepsilon_k^{\text{normal}}}\right)_+^2 > \varepsilon_k^{\text{normal}}$, which holds whenever

$$\bar{\delta}_k > 2\sqrt{\varepsilon_k^{\text{normal}}}. \tag{8}$$

Attacks inducing an RMS feature-space shift larger than twice the normal cross-modal prediction scale therefore satisfy $\varepsilon_k^{\text{attack}} > \varepsilon_k^{\text{normal}}$. We stress the scope: this is a statement about expectations and does not certify that any individual frame crosses a fixed threshold τ. □

Interpretation. The result shows why improving normal cross-modal prediction tightens the detection margin: a smaller $\varepsilon_k^{\text{normal}}$ lowers the minimum perturbation size needed for separation, provided the feature scale that sets $\bar{\delta}_k$ is held fixed. Because both quantities are measured in the learned head-output space, the bound constrains their ratio; a representation that simply shrinks lowers both without improving separability, which is why collapse must be avoided (Appendix F.2).

Multi-sensor case. The bound is stated for one target modality k; SensorAD computes the same score for each. If $K < M$ modalities are compromised, at least $M - K$ remain intact and can still supply evidence through cross-modal prediction. We do not assume independent pathways, since modalities share an encoder and correlated signals, and therefore claim no closed-form scaling law, only that richer suites provide more intact consistency checks under partial compromise. This behavior is evaluated empirically through modality ablation and coordinated-attack experiments.

B Extended Results

B.1 Per-Seed Performance and Paired Significance Tests

Table 2 gives per-seed AUROC. GPS spoofing, gradual GPS drift, DoS, and LiDAR tampering are stable across seeds; RGB, DVS, depth, and the static GPS offset have higher variance, consistent with their weaker detection margins. Seed standard deviations reflect training variability only. To test the comparison with CFM at the episode level, we compute per-episode AUROC for both methods (averaged over seeds) and apply a two-sided Wilcoxon signed-rank test to the paired differences, with a 95% bootstrap confidence interval obtained by resampling episodes (Table 3).

Table 2: Per-seed SensorAD AUROC by attack.

Attack	Seed 1	Seed 2	Seed 3	Mean	Std.
GPS spoofing	0.884	0.775	0.811	0.823	0.056
GPS gradual	0.905	0.812	0.837	0.851	0.048
RGB noise	0.622	0.559	0.607	0.596	0.033
LiDAR tampering	0.797	0.658	0.711	0.722	0.070
DVS noise	0.705	0.594	0.634	0.644	0.056
DoS	0.994	0.958	0.969	0.974	0.018
GPS static offset	0.607	0.545	0.628	0.593	0.043
Depth tampering	0.580	0.530	0.603	0.571	0.037
Overall	0.778	0.695	0.739	0.737	0.042

Table 3: **Paired per-episode comparison of SensorAD and CFM** (two-sided Wilcoxon signed-rank test on per-episode AUROC differences). The 95% episode-bootstrap CI for the four-attack physics-grounded mean difference is $[+0.083, +0.129]$.

Comparison	Episodes	SensorAD	CFM	Difference	p-value
Physics-grounded, all four attacks	48	0.843	0.738	+0.105	3.1×10^{-9}
Physics-grounded, excluding DoS	36	0.799	0.737	+0.062	3.9×10^{-7}
GPS spoofing	12	0.823	0.750	+0.073	4.9×10^{-4}
GPS gradual	12	0.851	0.780	+0.071	4.9×10^{-4}
LiDAR tampering	12	0.722	0.680	+0.042	0.0068

Table 4: **Component Ablation and Sensor Localization.** Mean over 3 seeds. DoS is excluded from localization since it zeroes two sensors at once. Depth tampering counts as correctly localized when attributed to RGB, the modality that carries the depth map.

(a) Component Ablation (Overall AUROC ↑)

Variant	Overall AUROC
Full SensorAD (L=8, d=768)	$0.737_{\pm0.042}$
w/o cross-modal masking	$0.652_{\pm0.026}$
w/o $\mathcal{L}_{\text{phys}}$	$0.699_{\pm0.031}$
w/o $\mathcal{L}_{\text{align}}$	$0.720_{\pm0.035}$
w/o $\mathcal{L}_{\text{temp}}$	$0.726_{\pm0.037}$
RGB + LiDAR (2 mod.)	$0.644_{\pm0.028}$
No DVS (3 mod.)	$0.725_{\pm0.025}$
Larger model (L=16)	$\mathbf{0.739}_{\pm\mathbf{0.040}}$

(b) Sensor Localization Accuracy

Attack	Top-1 (%)	Top-2 (%)
GPS spoofing	$90.3_{\pm7.4}$	$94.9_{\pm4.2}$
GPS gradual	$91.8_{\pm6.5}$	$95.9_{\pm3.2}$
RGB noise	$61.6_{\pm8.6}$	$76.0_{\pm6.5}$
LiDAR tampering	$82.5_{\pm12.3}$	$91.4_{\pm6.2}$
DVS noise	$71.6_{\pm13.8}$	$83.9_{\pm9.4}$
Depth tampering	$16.0_{\pm1.7}$	$44.8_{\pm1.3}$
GPS static offset	$55.5_{\pm4.4}$	$69.8_{\pm2.7}$
Overall	$\mathbf{67.0}_{\pm\mathbf{5.9}}$	$\mathbf{79.5}_{\pm\mathbf{4.0}}$

B.2 Pipeline Ablation

Table 6 compares pre-training and fine-tuning combinations. CARLA-only training gives target-domain adaptation and DVS coverage but lacks real-world variability. nuScenes-only training learns transferable physics (best on GPS) but is not adapted to CARLA and has no DVS. The full pipeline combines both effects.

Table 5: **Sensor localization confusion matrix by attack type** (% of attack frames attributed to each modality by $\arg\max_k a_k(t)$; rows sum to 100; mean over 3 seeds). The diagonal entry of each row (bold) matches Table 4b. *Depth tampering enters through the front-camera representation, so its correct attribution is RGB.

True attack ↓ / attributed to →	RGB	LiDAR	DVS	GPS/IMU
GPS spoofing	2.4	4.1	3.2	**90.3**
GPS gradual	1.8	3.7	2.7	**91.8**
GPS static offset	13.2	18.4	12.9	**55.5**
LiDAR tampering	6.7	**82.5**	4.8	6.0
RGB noise	**61.6**	24.7	8.9	4.8
DVS noise	11.2	9.4	**71.6**	7.8
Depth tampering*	**16.0**	68.4	7.1	8.5

Table 6: Pipeline ablation. Overall AUROC pooled over attack frames; mean over 3 seeds.

Pre-train	Fine-tune	Overall AUROC	GPS	LiDAR
None	CARLA	$0.714_{\pm 0.030}$	$0.754_{\pm 0.026}$	$0.674_{\pm 0.025}$
nuScenes	None	$0.669_{\pm 0.016}$	$0.786_{\pm 0.018}$	$0.692_{\pm 0.020}$
nuScenes	**CARLA**	$\mathbf{0.737_{\pm 0.042}}$	$\mathbf{0.823_{\pm 0.056}}$	$\mathbf{0.722_{\pm 0.070}}$

B.3 Coordinated and Spatially Aligned Attacks

In Table 7, the first four rows perturb each component sensor independently. The final row is the harder, spatially aligned case: an RGB patch is perturbed together with the LiDAR points falling inside its corresponding 3D camera frustum, so the two modalities remain spatially aligned by construction, in the spirit of the frustum attack of Hallyburton et al. (2022). This is partially consistency-preserving rather than fully consistent: the perturbations agree in position but not in appearance or semantics the way a rendered 3D object would. Detection falls from 0.717 to 0.653 but does not collapse, because the intact GPS/IMU and DVS pathways and the ego-motion physics targets still constrain the reconstruction. Fully consistent multi-sensor injection from a shared 3D asset (Tu et al., 2022; Cao et al., 2021) is a strictly harder threat model that we do not claim to handle. The 3-of-4 and spatially aligned rows use additional episodes generated with the same protocol; they are not part of the 132-episode benchmark. These rows report detection; which sensors $\arg\max_k a_k(t)$ selects under simultaneous compromise is a separate question that the single-sensor protocol of Table 4b does not answer, and we leave it to future work.

Table 7: Detection under coordinated attacks (AUROC ↑). The compound row compromises a single sensor (GPS/IMU) with two mechanisms at once. Mean over 3 seeds.

# Compromised	Combination	CFM	SensorAD
1/4	GPS only	$0.748_{\pm 0.021}$	$\mathbf{0.823_{\pm 0.056}}$
1/4 (compound)	GPS + static offset	$0.681_{\pm 0.022}$	$\mathbf{0.842_{\pm 0.043}}$
2/4	RGB + LiDAR	$0.665_{\pm 0.024}$	$\mathbf{0.717_{\pm 0.065}}$
3/4	RGB + LiDAR + DVS	$0.602_{\pm 0.028}$	$\mathbf{0.641_{\pm 0.026}}$
2/4 (aligned)	RGB patch + LiDAR frustum	$0.611_{\pm 0.027}$	$\mathbf{0.653_{\pm 0.031}}$

B.4 Detection Delay

At the gradual-drift rate of ≈0.111 m per frame (Appendix G.2), the 3.4 m offset at first detection corresponds to ≈31 frames after onset. Abrupt DoS and GPS spoofing are detected earliest because they create immediate disagreement between the attacked stream and the rest. Gradual drift, depth tampering, and the static GPS offset take more frames because their effects accumulate or appear indirectly through trajectory mismatch.

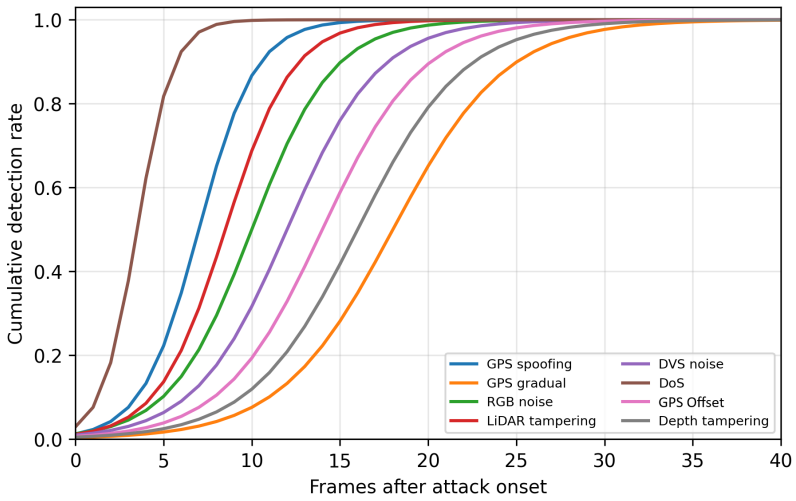


Figure 3: **Detection delay curves.** For each attack, cumulative fraction of eventually detected episodes whose first alarm has fired, as a function of frames after attack onset. Curves are conditioned on detection, so they reach 1 by construction; detection rates themselves are in Tables 1 and 20.

Table 8: Detection delay summary at 20 Hz. Abrupt-delay entries are *medians* over detected episodes; SensorAD and CFM overlap within one standard deviation there and separate on the gradual-drift offset. The offset is computed identically for all methods: per episode, accumulated displacement at the first threshold crossing, aggregated within seed, then mean and std across the three seeds. Lower is better.

Method	**Abrupt Delay ↓**	**GPS Drift Offset ↓**
Per-sensor avg.	$18.0_{\pm 3.1}$ frames	n/a
Anomaly Transformer	$15.4_{\pm 2.6}$ frames	$14.2_{\pm 1.7}$ m
CFM	$9.7_{\pm 1.2}$ frames	$12.6_{\pm 1.4}$ m
SensorAD	$\mathbf{9.4_{\pm 1.1}}$ **frames**	$\mathbf{3.4_{\pm 0.4}}$ **m**

C Post-Hoc Calibration, Ensembling, and Real-Data Checks

This appendix reports three additions that do not require retraining SensorAD: split-conformal calibration of the deployment threshold, late-fusion ensembling with a per-sensor visual detector, and a real-data check covering the five benchmark attacks that nuScenes can express.

C.1 Split-Conformal Calibration

The FPR@95TPR results in Table 1 measure behavior at an extreme operating point and are not the recommended deployment threshold. Using split conformal prediction with normal-only calibration data, we set τ to the $(1-\alpha)$-quantile of held-out normal scores; this controls marginal FPR $\leq \alpha$ on exchangeable normal frames. Table 9 reports operating points at $\alpha \in \{0.01, 0.05, 0.10\}$. The realized FPR matches α within Monte-Carlo error; at α=0.05, SensorAD achieves 61.2% TPR with 5.2% realized FPR, a calibrated operating point useful for deployment analysis rather than evidence of deployment readiness.

C.2 Late-Fusion Ensemble with PatchCore-Cam

SensorAD is weakest on weakly cross-modal attacks (RGB, DVS, depth) where per-sensor visual detectors retain an advantage (Table 1). We test whether the two paradigms compose by ensembling SensorAD with PatchCore-Cam, the per-sensor PatchCore camera model of Table 1, which scores the camera stream including the depth map folded into the front view: $s_{\text{ens}}(t) = s_{\text{SensorAD}}(t) + \alpha \cdot s_{\text{PatchCore-Cam}}(t)$, with α tuned on the normal validation set. Table 10 reports results on all eight

Table 9: **Split-conformal calibration on normal-only validation scores.** Thresholds are the $(1-\alpha)$-quantile of held-out normal scores. Realized normal FPR verifies coverage.

Method	Target FPR α	Threshold τ	TPR ↑	Realized FPR
SensorAD + split conformal	0.01	$4.82_{\pm 0.18}$	$0.421_{\pm 0.031}$	$0.012_{\pm 0.004}$
SensorAD + split conformal	0.05	$3.67_{\pm 0.14}$	$\mathbf{0.612_{\pm 0.027}}$	$0.052_{\pm 0.006}$
SensorAD + split conformal	0.10	$3.08_{\pm 0.12}$	$0.714_{\pm 0.024}$	$0.103_{\pm 0.008}$

attacks. The ensemble lifts RGB AUROC from 0.596 to 0.716 and depth from 0.571 to 0.642, while leaving physics-grounded performance essentially unchanged (0.843 to 0.845); on RGB and depth it remains below PatchCore-Cam alone (0.73, 0.68), so the gain is on average rather than per attack. The full eight-attack average rises from 0.722 to 0.750, and FPR@95 drops from 0.882 to 0.861.

Table 10: **Late-fusion ensemble (SensorAD + PatchCore-Cam).** AUROC ↑ on all eight attacks.

Method	GPS	GPS-grad	LiDAR	DoS	RGB	DVS	Static	Depth	Avg.	FPR@95
CFM	0.75	0.78	0.68	0.74	0.63	0.62	**0.66**	0.60	0.682	0.91
SensorAD	0.823	0.851	0.722	**0.974**	0.596	0.644	0.593	0.571	0.722	0.882
SensorAD + PatchCore-Cam	**0.824**	**0.852**	**0.731**	0.972	**0.716**	**0.659**	0.601	**0.642**	**0.750**	**0.861**

C.3 Real-Data Check on Injected nuScenes Attacks

The CARLA benchmark provides controlled attacks but raises real-to-sim concerns. We therefore inject the benchmark threat models directly on real nuScenes sensor streams, for every attack nuScenes can express: pose/GPS spoofing, gradual pose drift, LiDAR phantom injection, RGB noise, and RGB+LiDAR DoS. DVS is excluded because nuScenes has no event camera, and depth because nuScenes provides no depth-camera stream. No attack labels are used during training or threshold calibration.

Table 11 reports both checkpoints. Detection transfers on all five attacks: the mean falls from 0.793 in CARLA to 0.755 for the nuScenes-pretrained checkpoint and 0.735 for the CARLA-fine-tuned one, a 3.8 to 5.8 point drop with the ordering across attacks preserved. The Phase-1 checkpoint scores better, as expected since it was trained on nuScenes and Phase 2 adapts to CARLA; the Phase-2 column is the sim-to-real transfer figure. The threshold, however, does not transfer. Scoring the 5,122 keyframes of the held-out nuScenes scene split at each checkpoint's CARLA-calibrated 99th-percentile threshold gives a normal-driving FPR of 2.8% (Phase 1) and 5.6% (Phase 2), against 1.4% on CARLA normal driving and a 1% target. Ranking quality survives the domain change but the score scale does not, so a real deployment must recalibrate τ on real normal data. This is evidence that the cross-modal signal is not a CARLA artifact; it is not validation against physically realized attacks, which remains future work.

Table 11: **Injected attacks on real nuScenes streams.** Top: AUROC ↑ (the CARLA column repeats Table 1). Bottom: false-positive rate on normal driving at the CARLA-calibrated 99th-percentile threshold (target $\leq$1%; 72,000 CARLA frames, 5,122 nuScenes keyframes). DVS and depth have no nuScenes counterpart. Mean over 3 seeds.

Attack	CARLA (Phase 2)	nuScenes (Phase 1)	nuScenes (Phase 2)
Pose/GPS spoofing	$0.823_{\pm 0.056}$	$0.764_{\pm 0.032}$	$0.741_{\pm 0.037}$
Gradual pose drift	$0.851_{\pm 0.048}$	$0.786_{\pm 0.039}$	$0.758_{\pm 0.043}$
LiDAR phantom injection	$0.722_{\pm 0.070}$	$0.691_{\pm 0.052}$	$0.668_{\pm 0.057}$
RGB noise	$0.596_{\pm 0.033}$	$0.584_{\pm 0.029}$	$0.571_{\pm 0.034}$
RGB + LiDAR DoS	$0.974_{\pm 0.018}$	$0.951_{\pm 0.022}$	$0.938_{\pm 0.025}$
Mean AUROC	**0.793**	0.755	0.735
Normal-driving FPR (%)	$1.4_{\pm 0.3}$	$2.8_{\pm 0.5}$	$5.6_{\pm 0.8}$

D Additional Robustness, Generalization, and Deployment Analyses

This appendix reports the robustness, generalization, and deployment analyses summarized in Sections 4.3 to 4.6. All reuse the frozen Phase-2 checkpoint except the held-out-environment study (Appendix D.6), which re-runs Phase 2 from scratch.

D.1 Isolating Temporal Modeling from Cross-Modal Conditioning

SensorAD observes a 50-step GPS/IMU window whereas the per-sensor GPS/IMU baseline is static, so the GPS margin could in principle come from temporal modeling rather than from cross-modal conditioning. Table 12 separates the two with matched single-modality temporal detectors trained normal-only under the identical protocol.

Two conclusions follow. Temporal modeling alone accounts for a substantial part of the abrupt-GPS gain (0.651 $\rightarrow$ 0.742), so the confound was real. Cross-modal conditioning accounts for the remainder (0.742 $\rightarrow$ 0.823) and is decisive on gradual drift, where the accumulated offset at first detection is 3.4 m against 15.1 m: a unimodal temporal model cannot flag a smooth linear ramp until the trajectory becomes self-inconsistent, which is late by construction, whereas cross-modal evidence flags it as soon as GPS diverges from LiDAR geometry and visual odometry. Incidentally, the GPS-only LSTM-AE outperforms the pooled multimodal one, because pooling dilutes an 11-dimensional signal against three high-dimensional visual streams; we therefore treat the unimodal variant as the primary temporal reference. It also beats SensorAD on the static GPS offset, for the reason given in Section 4.3.

Table 12: **Single-modality temporal baselines** (AUROC ↑; drift offset in metres, lower is better). Rows 3 and 4 repeat Table 1 and Table 8 for reference. Mean over 3 seeds.

Detector	Modalities	Temporal	GPS	GPS-grad	Static	Drift offset
Mahalanobis, raw GPS/IMU	GPS only	No	0.651	0.688	0.617	18.8 m
LSTM-AE, GPS/IMU only	GPS only	Yes	0.742	0.774	**0.646**	15.1 m
LSTM-AE, pooled multimodal	All	Yes	0.720	0.760	0.630	n/a
SensorAD	All	Yes	**0.823**	**0.851**	0.593	**3.4 m**

D.2 Granularity-Matched Control

CFM and the other representation baselines consume pooled per-modality `[CLS]` vectors, whereas SensorAD attends over dense tokens, so part of the reported margin could reflect input granularity rather than cross-modal design. Table 13 removes that confound from the other side: we retrain SensorAD on the same pooled `[CLS]` inputs CFM receives, holding everything else fixed.

At matched granularity SensorAD leads CFM on physics-grounded attacks by 6.4 points (0.802 vs. 0.738), and dense tokens add a further 4.1 to reach 0.843. Roughly 60% of the 10.5-point headline margin is therefore attributable to the cross-modal design and the remainder to input granularity. DoS contributes disproportionately to the design share: excluding it, pooled SensorAD leads CFM by 2.8 points (0.765 vs. 0.737) out of a 6.2-point total, about 45%, and the pooled LiDAR comparison is close to a tie (0.690 vs. 0.680). The ordering on weakly cross-modal attacks is unchanged: pooled SensorAD scores 0.591 against CFM's 0.628, so granularity is not what makes SensorAD weaker there either. A dense-token CFM would be the symmetric control and remains future work.

D.3 Per-Attack Contribution of Modality Masking

Table 14 shows that the four extra masked passes buy 0.013 pooled AUROC over the single-pass read-out $A_1(t)$, or 0.049 averaged over the eight attacks, and Table 29 gives the latency breakdown. That pooled figure averages two very different regimes, so Table 15 reports the comparison per attack.

On locally plausible attacks masking is decisive. When the compromised sensor still looks locally plausible, the all-visible pass lets it reconstruct itself from its own tokens, hiding the inconsistency;

Table 13: **Granularity-matched control** (AUROC ↑). SensorAD retrained on the pooled `[CLS]` features CFM consumes, against SensorAD on dense tokens. Mean over 3 seeds.

Attack	CFM, pooled	SensorAD, pooled	SensorAD, dense
GPS spoofing	$0.750_{\pm 0.020}$	$0.793_{\pm 0.049}$	**$0.823_{\pm 0.056}$**
GPS gradual	$0.780_{\pm 0.020}$	$0.811_{\pm 0.045}$	**$0.851_{\pm 0.048}$**
LiDAR tampering	$0.680_{\pm 0.020}$	$0.690_{\pm 0.061}$	**$0.722_{\pm 0.070}$**
DoS	$0.740_{\pm 0.020}$	$0.912_{\pm 0.025}$	**$0.974_{\pm 0.018}$**
RGB noise	**$0.630_{\pm 0.030}$**	$0.590_{\pm 0.034}$	$0.596_{\pm 0.033}$
DVS noise	$0.620_{\pm 0.030}$	$0.632_{\pm 0.052}$	**$0.644_{\pm 0.056}$**
GPS static offset	**$0.660_{\pm 0.020}$**	$0.580_{\pm 0.044}$	$0.593_{\pm 0.043}$
Depth tampering	**$0.600_{\pm 0.030}$**	$0.560_{\pm 0.038}$	$0.571_{\pm 0.037}$
Physics-grounded avg.	0.738	$0.802_{\pm 0.045}$	**0.843**
Weakly cross-modal avg.	**0.628**	$0.591_{\pm 0.042}$	0.601
Eight-attack avg.	$0.682_{\pm 0.020}$	$0.696_{\pm 0.044}$	**$0.722_{\pm 0.042}$**

masking removes that self-information and exposes it. On blatant or weakly cross-modal attacks (DoS, RGB, DVS, depth) the gap is within noise, which is exactly what the complementarity story predicts. Masking is additionally what makes attribution reliable: per-head errors on the all-visible pass already localize at 58.1% top-1, and the masked passes lift this to 67.0% (Table 14). Table 4a shows the complementary training-time effect, where removing modality-level masking from the objective costs 0.085 AUROC, the largest single component effect.

Table 14: **Two read-outs of the same weights** (RTX 3090, batch 1, BF16). AUROC is pooled over all attack frames. The five-pass score is the paper's primary detector; the single-pass score is a continuous-monitoring operating point.

Read-out	Passes	Latency	Overall AUROC ↑	Top-1 localization
Single-pass $A_1(t)$	1	45.8 ms (21.8 FPS)	$0.724_{\pm 0.039}$	$58.1_{\pm 6.4}\%$
Five-pass $A(t)$ (primary)	5	106.7 ms (9.4 FPS)	**$0.737_{\pm 0.042}$**	**$67.0_{\pm 5.9}\%$**

Table 15: **What masking contributes, per attack** (AUROC ↑). A_1 is the single-pass all-visible score; A is the five-pass leave-one-modality-out score of Eq. (2).

Attack	Single-pass A_1	Masked A	Δ
GPS gradual drift	0.651	**0.851**	+0.200
LiDAR tampering	0.615	**0.722**	+0.107
GPS static offset	0.520	**0.593**	+0.073
GPS spoofing	0.770	**0.823**	+0.053
DoS	**0.991**	0.974	−0.017
RGB noise	**0.603**	0.596	−0.007
DVS noise	**0.651**	0.644	−0.007
Depth tampering	**0.580**	0.571	−0.009

D.4 Attack Magnitude Sensitivity

The benchmark evaluates one magnitude per attack, which tells a deployer nothing about the regime in which the detector fails. Table 16 re-injects each attack at $0.25\times$, $0.5\times$ and $2\times$ the benchmark magnitude at the sensor-output level on the same episodes. Detection increases monotonically for every attack and the physics-grounded attacks stay strongest at every level. At $0.25\times$ detection falls toward chance (0.526 to 0.638), which is the regime in which the feature-space perturbation falls below the separability scale of Proposition 1; we present this as consistent with the bound's qualitative prediction, not as a measurement of the inequality.

Table 16: **Attack magnitude sweep** (AUROC ↑). The 1× column is the benchmark operating point (Table 1). Mean over 3 seeds.

Attack	0.25×	0.5×	1× (benchmark)	2×
GPS spoofing	0.638	0.734	0.823	**0.907**
GPS gradual	0.614	0.741	0.851	**0.932**
LiDAR tampering	0.574	0.649	0.722	**0.806**
RGB noise	0.526	0.558	0.596	**0.657**

D.5 White-Box Adaptive Attackers

Our threat model (Section 3.1) excludes white-box access to the detector, but the informative question is how the method fails when that assumption is violated. We give the attacker the frozen detector and let it parameterize the spoofed GPS trajectory by (onset, ramp rate, curvature), minimizing the peak anomaly score $\max_t a_{\mathrm{GPS}}(t)$ by projected gradient over those parameters subject to achieving a target lateral ego displacement d by the end of the attack window. The objective is genuinely harmful rather than cosmetic: move the estimated ego position by d while keeping the per-frame score below threshold. The budget multiplier scales the feasible set jointly over d, ramp rate, curvature bound and optimization steps, and the $0.5\times$, $1\times$ and $2\times$ settings are nested, so a larger budget is a strictly stronger attacker and AUROC can only fall as the budget grows; what the experiment measures is how fast. We target GPS because it is the physics-grounded attack an adversary can shape most freely, and therefore the most stringent single-modality test of the consistency signal.

Low-budget adaptive attacks remain detectable (0.792; Table 17), but at $2\times$ the detector is materially weakened (0.632), and SensorAD is not certified against white-box adaptive attackers. A fully joint multimodal-consistency-preserving attacker (Tu et al., 2022; Cao et al., 2021) is strictly harder still, requires simultaneous compromise of multiple physically separated data paths, and is untested here.

Table 17: **Consistency-aware white-box attacker on GPS.** The black-box row is the benchmark gradual-drift attack (Table 1). Mean over 3 seeds.

Attacker	AUROC ↑	TPR@5%FPR ↑
Black-box gradual (benchmark)	**0.851**	**0.648**
White-box, 0.5× budget	0.792	0.557
White-box, 1× budget	0.711	0.432
White-box, 2× budget	0.632	0.318

D.6 Held-Out Towns and Weather

Splitting by route seed alone tests route generalization but not environment generalization, so we re-run Phase 2 in full (20 epochs, 3 seeds) while holding out an entire town or an entire weather condition, evaluating only on the held-out environment.

Degradation to genuinely unseen towns and weather is at most 0.028 AUROC (Table 18), and the physics-grounded GPS channel is the most stable component, consistent with the physics-consistency loss doing the work it was introduced for. Within seen environments the model is flat (0.726 to 0.746 AUROC, 0.9 to 1.8% FPR). This is an additional generalization test; Table 1 and all other results use the route-seed split of Appendix G.3.

Table 18: **Held-out-environment generalization.** Full 20-epoch Phase-2 retrains, 3 seeds. Δ is against the all-environment model evaluated on the same target environment.

Fine-tune data	Evaluate on	Overall AUROC	GPS	Δ
Towns 01/03/05	Town10HD (unseen)	0.719	0.804	−0.027
Towns 03/05/10HD	Town01 (unseen)	0.719	0.795	−0.012
ClearNoon + ClearNight	WetNoon (unseen)	0.711	0.792	−0.028

D.7 Downstream Consequence of Each Attack

Detection quality is only meaningful if the attacks are consequential. Table 19 measures each attack's effect on the task it targets, using a standard 3D detector for the perception rows and ego-pose error for localization. Under DoS both input streams are zeroed, so the detector returns no valid detections and 3D mAP is 0 by construction rather than by measurement. Because the rows use different downstream quantities we do not rank severity across attack classes. Within the 3D mAP rows, DoS is both the most damaging and the best detected, but the ordering is not monotone: the optimized adversarial camera perturbation costs more mAP than LiDAR tampering yet is detected less reliably.

The final row is worth isolating. Random salt-and-pepper RGB noise is a weak threat model, costing 3.3 mAP. An optimized adversarial RGB perturbation costs 12.4 mAP, and SensorAD detects it *better* (0.674 vs. 0.596), because a perturbation tuned to corrupt the image induces larger cross-modal inconsistency than unstructured noise does. The practical reading is that in the regime where an adversary digitally compromises a camera, cross-modal consistency is more informative than a random-noise threat model would suggest.

Table 19: **Downstream impact of each attack.** Mean over 3 seeds; AUROC is from Table 1, with the late-fusion ensemble figure in parentheses (Appendix C.2).

Attack	Downstream metric	Clean	Attacked	SensorAD AUROC
GPS spoofing	ego-pose error (m)	0.42	18.7 (+18.3)	0.823
DoS	3D mAP (see note)	44.6	0.0 (−44.6)	0.974
LiDAR tampering	3D mAP	44.6	35.9 (−8.7)	0.722
LiDAR tampering	3D NDS	57.3	50.4 (−6.9)	0.722
Depth tampering	depth MAE (m)	0.78	1.31 (+0.53)	0.571 (0.642)
RGB noise	3D mAP	44.6	41.3 (−3.3)	0.596 (0.716)
Adversarial RGB	3D mAP	44.6	32.2 (−12.4)	0.674

D.8 Operational Metrics at Calibrated Thresholds

AUROC is a ranking metric and not a deployment metric. Table 20 reports precision, recall, F1, PR-AUC, episode-level detection and early-detection rate at the split-conformal operating points of Appendix C.1, computed on the same scores with no retraining, over the 132 attack episodes (264,000 frames, 49.6% attack-positive). These are the quantities a safety monitor is actually specified against; precision follows from recall, the realized FPR and that class balance.

Episode-level detection counts an attack episode as detected if any frame inside the attack window is flagged. That criterion inflates with sequence length and is meaningless without its counterpart, so we also report the fraction of held-out *normal* episodes of the same length raising at least one alarm: 2.1%, 8.9% and 16.3% at $\alpha = 0.01$, 0.05 and 0.10. The two must be read together.

Table 20: **Operational metrics at calibrated operating points.** Computed over the 132 attack episodes (264,000 frames, 49.6% attack-positive). Recall is the TPR column of Table 9, evaluated at the realized conformal FPR rather than at exactly α. Early detection is the fraction of episodes flagged within 10 frames of onset.

Operating point	Prec.	Recall	F1	PR-AUC	Episode det.	Early det.	Normal-ep. alarm
Conformal $\alpha = 0.01$	**0.972**	0.421	0.588	0.716	0.783	0.468	**2.1%**
Conformal $\alpha = 0.05$	0.921	0.612	0.735	0.716	0.892	0.647	8.9%
Conformal $\alpha = 0.10$	0.872	**0.714**	**0.785**	0.716	**0.941**	**0.731**	16.3%

D.9 False Alarms Under Benign Shift

Table 21 reports the full comparison at the calibrated 99th-percentile normal threshold, summarized in Section 4.5. SensorAD has the lowest false-alarm rate in every condition, but weather and lighting shift still raise it by six to eight times over normal driving, which is why we describe environment-conditioned thresholding as a deployment requirement rather than an optional refinement.

Table 21: **False positive rate under benign distributional shift**, at the calibrated 99th-percentile normal threshold. Lower is better. Mean over 3 seeds.

Condition	Max-Fusion	Mahalanobis	CFM	SensorAD
Normal driving	$4.9_{\pm 0.7}$	$4.2_{\pm 0.6}$	$3.8_{\pm 0.6}$	$\mathbf{1.4}_{\pm \mathbf{0.3}}$
Weather (sun → rain)	$16.4_{\pm 1.9}$	$14.1_{\pm 1.7}$	$12.9_{\pm 1.6}$	$\mathbf{8.7}_{\pm \mathbf{1.1}}$
Lighting (day → night)	$19.8_{\pm 2.1}$	$17.2_{\pm 2.0}$	$15.6_{\pm 1.9}$	$\mathbf{10.9}_{\pm \mathbf{1.4}}$
Traffic density change	$13.7_{\pm 1.8}$	$11.5_{\pm 1.5}$	$9.8_{\pm 1.3}$	$\mathbf{6.1}_{\pm \mathbf{0.9}}$

E Ablation Details

E.1 Masking Strategy

Spatial-masking correspondence. Spatial masking must align three different geometries. We sample an axis-aligned 16×16 m region of the LiDAR BEV grid, project its eight corners into each camera using the known CARLA/nuScenes extrinsics and intrinsics, take the axis-aligned bounding box of the projected corners in image space, and mask every RGB (and, for the front view, DVS) patch whose centre falls inside that box. Cameras whose frustum does not intersect the region are left unmasked. GPS/IMU has no spatial extent and is never spatially masked.

Full-modality dropout is the strongest single strategy (Table 22) because it most directly matches deployment scoring. Spatial masking alone wins on RGB/Depth (spatially localized masking is well matched to within-modality visual perturbations) but underperforms on physics-grounded attacks. The full mixture wins overall.

Table 22: Masking strategy ablation. Overall AUROC over 3 seeds.

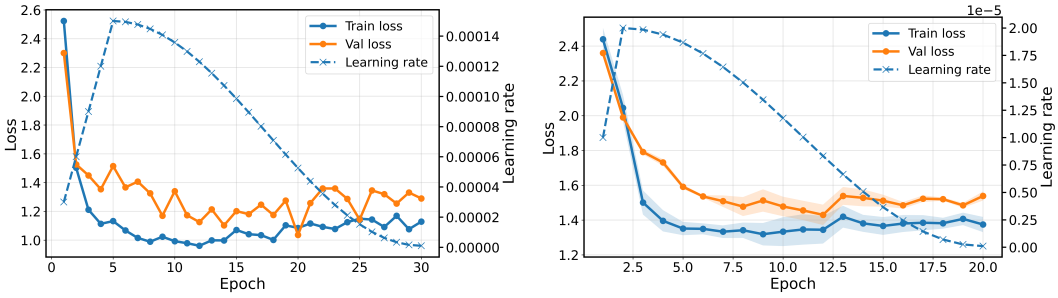
Masking Strategy	Overall AUROC	GPS-grad	RGB/Depth Avg.
Full-modality dropout only	$0.714_{\pm 0.045}$	$0.833_{\pm 0.046}$	$0.570_{\pm 0.028}$
Multi-modality dropout only	$0.695_{\pm 0.047}$	$0.815_{\pm 0.047}$	$0.563_{\pm 0.030}$
Partial corruption only	$0.669_{\pm 0.049}$	$0.756_{\pm 0.051}$	$0.586_{\pm 0.029}$
Temporal gap only	$0.682_{\pm 0.048}$	$0.795_{\pm 0.049}$	$0.548_{\pm 0.031}$
Spatial masking only	$0.660_{\pm 0.050}$	$0.730_{\pm 0.053}$	$\mathbf{0.594}_{\pm \mathbf{0.030}}$
Full mixture	$\mathbf{0.737}_{\pm \mathbf{0.042}}$	$\mathbf{0.851}_{\pm \mathbf{0.048}}$	$0.584_{\pm 0.029}$

E.2 Modality Subsets

Models are trained from scratch with only the indicated modalities (Table 23). The strongest two-modality pair is LiDAR + GPS/IMU, which combines geometry with ego-motion. Adding DVS to that subset comes within 0.004 of the full model on pooled AUROC; pooled AUROC is dominated by the physics-grounded attacks, however, and a model without cameras cannot score camera attacks at all.

Table 23: Modality subset ablation. Overall AUROC over 3 seeds. Numbers match Table 4a for the shared configurations.

Modalities Used	Overall AUROC
RGB + LiDAR	$0.644_{\pm 0.028}$
LiDAR + GPS/IMU	$0.711_{\pm 0.026}$
GPS/IMU + DVS	$0.648_{\pm 0.031}$
RGB + LiDAR + GPS/IMU	$0.725_{\pm 0.025}$
LiDAR + GPS/IMU + DVS	$0.733_{\pm 0.034}$
RGB + LiDAR + GPS/IMU + DVS	$\mathbf{0.737}_{\pm \mathbf{0.042}}$



(a) Pre-training total loss.

(b) Fine-tuning total loss across 3 seeds.

Figure 4: **Training dynamics of SensorAD.** Both stages show stable optimization; visible plateaus correspond to staged activation of auxiliary losses.

E.3 Loss Weight Selection

Selection was carried out on the disjoint pool of Appendix G.3; the values tabulated in Table 24 are the corresponding scores on the reporting benchmark, so that they are directly comparable with Table 1. Selection uses attack labels; training and threshold calibration do not. Reconstruction-only training is already competitive because it directly optimizes the deployment score. Adding alignment, physics, and temporal gives consistent further improvements. Learned uncertainty weighting (Kendall et al., 2018) is comparable on the mean but has substantially higher seed variance.

Table 24: Loss weight selection. Overall AUROC over 3 seeds.

Configuration	**Weights**	**Overall AUROC**	**FPR@95** $\downarrow$
Reconstruction only	$(1.0, 0, 0, 0)$	$0.702_{\pm 0.029}$	$0.914_{\pm 0.022}$
Recon + align + physics	$(1.0, 0.2, 0.1, 0)$	$0.729_{\pm 0.037}$	$0.894_{\pm 0.020}$
Full fixed weights	$(1.0, 0.2, 0.1, 0.1)$	$\mathbf{0.737}_{\pm \mathbf{0.042}}$	$\mathbf{0.882}_{\pm \mathbf{0.017}}$
Learned uncertainty (Kendall et al., 2018)	learned	$0.714_{\pm 0.033}$	$0.921_{\pm 0.027}$

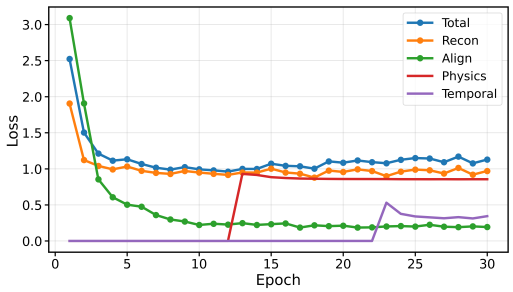
E.4 Training Curves

Figure 4 shows total-loss curves for both stages: optimization is stable, and the visible plateaus correspond to the staged activation of $\mathcal{L}_{\text{phys}}$ and $\mathcal{L}_{\text{temp}}$. Figure 5 reports component-level pre-training and fine-tuning curves. Pre-training component curves are shown for a representative seed, while fine-tuning curves report mean and standard deviation over three seeds. Reconstruction and alignment losses decrease consistently during the early epochs, confirming that the model learns cross-modal predictive structure from normal data. Physics and temporal losses introduce visible but bounded transitions after activation. During fine-tuning, the standard-deviation bands remain narrow, suggesting that the real-to-sim adaptation is stable across seeds.

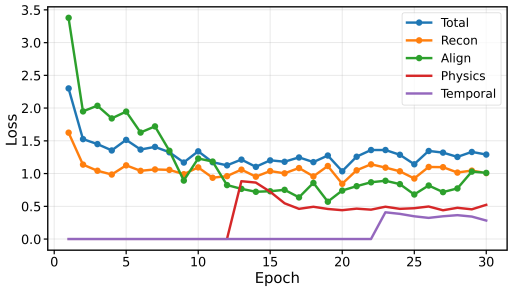
F Implementation Details

Transformer. 8 layers, $d = 768$, 12 heads, $d_{\text{ff}} = 2048$, dropout 0.1, GELU, pre-LayerNorm. Xavier uniform for non-backbone parameters. Mask token from $\mathcal{N}(0, 0.02)$. Modality embeddings: a learned 4×768 table. Counting 4 modality, 4 camera, 1 mask and 50 GPS/IMU positional embeddings at d=768 gives 45,312 parameters (0.045M). Base model has $\sim$149.4M total parameters with 86.6M frozen DINOv2 weights and 62.8M trainable. The larger model used for the scaling ablation has $L = 16$ and $d_{\text{ff}} = 3072$, with all other components unchanged.

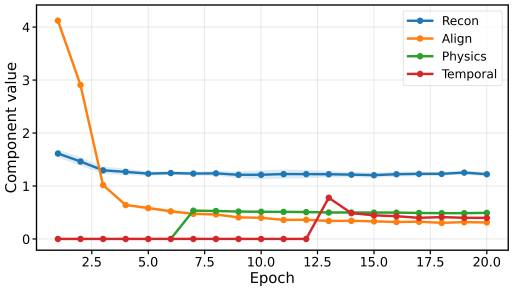
Tokenizers. Table 25 summarizes per-modality tokenizers, and Table 27 the parameter budget. Total sequence length per frame: $1028 + 65 + 50 + 51 = \mathbf{1194}$ tokens with all four modalities, $\mathbf{1144}$ during Phase 1 (no DVS).



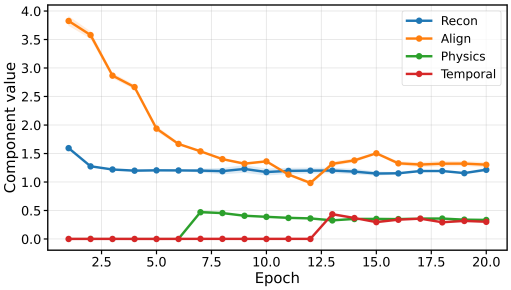
(a) Pre-training train component losses.



(b) Pre-training validation component losses.



(c) Fine-tuning train component losses across three seeds.



(d) Fine-tuning validation component losses across three seeds.

Figure 5: **Component-level training curves.** Pre-training component curves are shown for a representative seed. Fine-tuning component curves report the mean and standard deviation over three seeds. Reconstruction and alignment losses decrease consistently, while physics and temporal losses remain bounded after their staged activation.

Temporal input. SensorAD scores one synchronized clip at a time. RGB, LiDAR, and DVS tokens are taken from the current keyframe, while GPS/IMU uses a 50-step history ending at that timestamp. The transformer processes the resulting tokens jointly within the clip and does not attend across future clips.

Table 25: Modality-specific tokenizers.

Modality	**Tokenizer**	**Tokens / frame**
RGB	Frozen DINOv2 ViT-B/14, 224^2 input, 14×14 patch size, 4 cameras (front view carries the co-located depth map; Appendix G.4), Linear($768 \to 768$)	$4 \times (1 + 256) = 1028$
LiDAR	7-dim point features $\to$ Linear+BN+ReLU $\to$ BEV grid (64^2, voxel 1.5625 m) $\to$ 3-layer CNN	$1 + 8 \times 8 = 65$
DVS	5 temporal bins, 10 channels, 224^2 input, 3-layer CNN stem, Conv2d patch embedding (4^2, stride 4)	$1 + 7 \times 7 = 50$
GPS/IMU	11-dim $\times$ 50-step window, Linear($11 \to 768$) $\to$ LN $\to$ GELU $\to$ Linear, learned positional encoding	$1 + 50 = 51$

Reconstruction heads. 2-layer MLPs per modality: Linear($768 \to 768$)$\to$GELU$\to$Linear($768 \to d_k$), with $d_k = 768$ for RGB (DINOv2 dim), $d_k = 256$ for LiDAR/DVS (encoder feature dim), and $d_k = 11$ for GPS/IMU. Both prediction and target pass through the same head (target stop-gradiented), so the loss is computed in head-output space and matches Eq. (2) exactly.

F.1 Masking Probabilities and Training

Masking modes: full-modality 30%, multi-modality 10% (Phase 1) / 15% (Phase 2), partial corruption (75% of tokens) 35% / 30%, temporal gap 15%, spatial masking 10%. In Phase 1, DVS is always treated as masked.

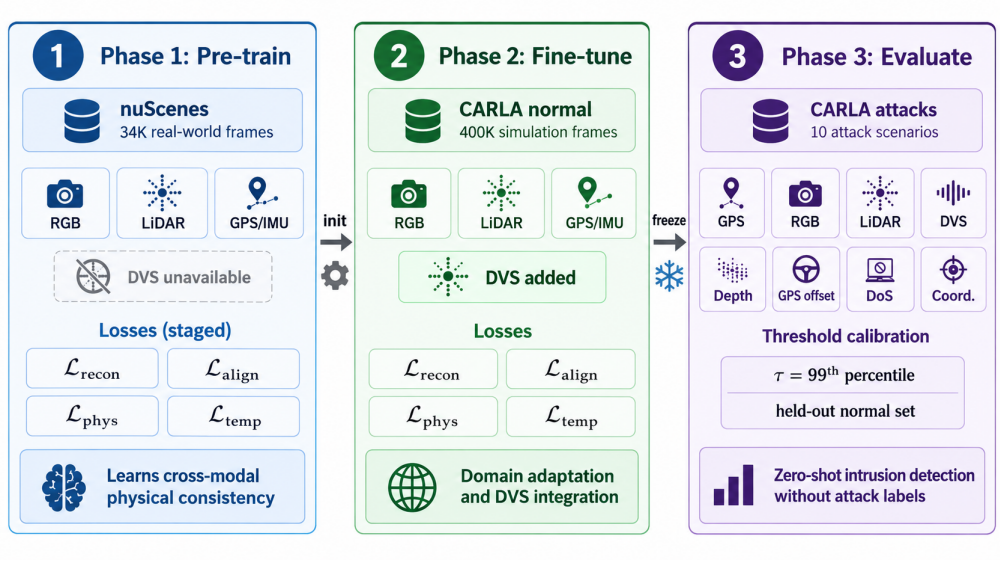


Figure 6: **Three-phase real-to-sim training pipeline.** Phase 1 (*Pre-train*) learns cross-modal physical consistency on nuScenes using RGB, LiDAR, and GPS/IMU, with DVS unavailable. Phase 2 (*Fine-tune*) adapts to the CARLA domain on normal data and integrates the DVS modality. Phase 3 (*Evaluate*) freezes the model and performs attack detection on CARLA attack scenarios, with threshold τ calibrated at the 99th percentile on a held-out normal set.

Phase 1. nuScenes v1.0-trainval, 85/15 scene-level split, 4-keyframe clips with stride 2. AdamW (β_1=0.9, β_2=0.95, wd 0.05), base lr 1.5×10^{-4}, cosine to 10^{-6}, 5-epoch warmup, 30 epochs, batch 64/GPU, grad clip 1.0, BF16 mixed precision, gradient checkpointing. Loss staging: epochs 1 to 12 recon + align; 13 to 22 add $\mathcal{L}_{\text{phys}}$ with 6-epoch warmup; 23 to 30 add $\mathcal{L}_{\text{temp}}$.

Phase 2. CARLA normal, ∼200 episodes, 4 towns, 3 weather conditions. Initialize from Phase 1 best checkpoint (`strict=False`); DVS tokenizer randomly initialized. AdamW with two parameter groups: non-DVS lr 2.0×10^{-5}, DVS lr 2.0×10^{-4} ($10\times$). 20 epochs, batch 64/GPU. Backbone (transformer + final norm) frozen for epochs 1 and 2. Loss staging: 1 to 6 recon + align; 7 to 12 add $\mathcal{L}_{\text{phys}}$ with 4-epoch warmup; 13 to 20 add $\mathcal{L}_{\text{temp}}$.

Phase 3. Frozen Phase 2 checkpoint. Threshold calibrated at the 99th percentile of $A(t)$ on a held-out normal set, targeting $\leq$1% FPR.

F.2 Loss Details

Reconstruction. MSE between head outputs at masked and stop-gradiented unmasked positions.

Collapse prevention. The reconstruction target is computed from the visible-token pass with stop-gradient, while the prediction branch receives the target modality masked and must infer it from the remaining modalities. A constant encoder output would still minimize $\mathcal{L}_{\text{recon}}$ alone; in practice this input asymmetry, together with modality-specific tokenizers, modality and position embeddings, the uniformity term of the InfoNCE alignment loss, and the physics loss, which anchors the GPS/IMU head to raw measurements, prevents the constant-output solution. We also observed non-zero per-modality representation variance throughout training.

During training, reconstruction targets are obtained from a clean reference pass before applying modality dropout; the masked pass is used only for prediction.

Contrastive alignment. Symmetric InfoNCE on ℓ_2-normalized `[CLS]` tokens, $\tau = 0.07$. Positives: tokens from different modalities at the same timestep; negatives: other samples in the batch.

Physics consistency. Smooth-ℓ_1 loss on the GPS/IMU head's predictions (with gradients) versus raw batch data (detached). Four terms, each clamped at 2.0: position grounding, which compares the predicted position against the reported GPS/IMU position after subtracting the first sample of the window from both, so it constrains the trajectory relative to its own start rather than absolute world position (a constant translation cancels exactly under that subtraction, which is why the static GPS offset of Table 26 is invisible to this term and to the derivative terms alike); velocity-acceleration consistency, comparing predicted acceleration $\times \Delta t$ against predicted velocity differences; position-velocity consistency, comparing predicted velocity $\times \Delta t$ against predicted position differences; and heading-gyro consistency. Min sequence length 3. Linear warmup factor $\alpha = \min(1, (\text{epoch} - \text{stage2_start})/W)$ with $W = 6$ in Phase 1 and $W = 4$ in Phase 2, and `stage2_start` $= 13$ in Phase 1 and 7 in Phase 2, matching the staging schedule above.

Temporal prediction. Concatenated 4-modality `[CLS]` (3072 dims) $\rightarrow$ Linear($3072 \rightarrow 1536$) $\rightarrow$ GELU $\rightarrow$ Linear($1536 \rightarrow 3072$), MSE against next-frame `[CLS]` (stop-gradiented).

F.3 Inference

For each frame: (1) tokenize all modalities; (2) for each k, replace tokens of k with `[MASK]`, encode, and pass through head h_k; (3) one unmasked reference pass; (4) compute $a_k(t)$; (5) overall score $A(t) = \text{mean}_k\, a_k(t)$, localization $k^* = \arg\max_k a_k(t)$, detection $A(t) > \tau$. This requires 5 forward passes per frame, batchable.

All measurements use a single NVIDIA RTX 3090 (24 GB) with mixed precision. Phase 1 nuScenes pre-training takes 28.4 h with 24 GB peak memory, and Phase 2 CARLA fine-tuning takes 74.1 h with 21 GB peak memory. Single-pass inference takes 45.3 ms with 0.61 GB peak memory, while full five-pass localized scoring takes 106.7 ms with 0.63 GB peak memory. SensorAD is heavier than most baselines because localized scoring performs one masked prediction pass per modality to obtain both global anomaly scoring and per-sensor attribution. Real-time deployment would benefit from caching, early-exit scoring, or distillation. Cost also grows quadratically in sequence length, so additional camera views or larger detection ROIs raise the cross-modal share; token reduction is the natural mitigation.

G Dataset Details

The CARLA attack benchmark uses simulator version 0.9.15 in synchronous mode at 20 Hz. The ego vehicle is a Tesla Model 3 controlled by the CARLA autopilot, with 30 background vehicles and 20 pedestrians (fixed seed). Each frame contains four RGB cameras (front, front-left, front-right, rear), a depth camera co-located with the front RGB camera, a roof-mounted LiDAR, a co-located DVS sharing the front camera's field of view, and an 11-dim GPS/IMU vector. Frame-level ground truth records both attack status and attack type.

G.1 Composition

The benchmark contains 132 attack episodes across the ten attack types of Table 26, together with 36 attack-free episodes used as the normal evaluation pool. Every episode is 2000 frames; attack episodes are structured as 500 clean $\rightarrow$ 1000 attack $\rightarrow$ 500 clean, and the clean bookends support both detection-quality and detection-delay analysis. The 168 episodes total 336,000 frames, of which 131,064 are attack-positive and 204,936 normal. The attack-positive count falls below the nominal 132×1000 because 936 clips straddling attack-window boundaries are labelled normal; the normal count is therefore the 132,000 frames of clean bookends, the 72,000 frames of the attack-free episodes, and those 936 boundary clips. Coverage spans 4 towns (Town01, Town03, Town05, Town10HD) and 3 weather conditions (ClearNight, ClearNoon, WetNoon), with 3 routes per (town, weather) cell.

G.2 ATTACK INJECTION

Justification of the threat models. The single-sensor attacks build on the CARLA attack vectors of Sakhai et al. (2025), to which we add a gradual drift variant and compound settings. GPS spoofing and gradual drift follow the SDR-based civilian-GPS override and stealthy-ramp threats of Tippenhauer et al. (2011) and Shen et al. (2020), with per-frame offsets that cover and exceed the documented 5 to 50 m range. LiDAR phantom injection follows the synchronized-laser relay attacks of Cao et al. (2019) and Sun et al. (2020). Depth and RGB perturbations model ISP/firmware compromise and physical-world camera corruption (Boloor et al., 2020; Petit et al., 2015). DoS models cable failure or bus jamming. RGB salt-and-pepper and DVS event flooding are deliberately generic corruptions rather than replicas of a physically realized attack, included so that the benchmark is not tuned to a single threat model. The value of the suite is the six-axis protocol, which is independent of any single attack's fidelity, and Appendix D.7 reports what each attack actually costs downstream.

Attacks are injected at the sensor-output level, after ground-truth measurements have been generated and before streams are written to disk. This corresponds to a threat in which an attacker has compromised a single sensor's data path (driver, ISP, or upstream bus) but has not compromised the rest of the perception stack. Coordinated attacks reuse single-attack injection for each component sensor and apply both within the 1000-frame attack window. Table 26 lists per-modality parameters.

Table 26: Attack scenarios, primary affected modality, and threat-model parameters.

Attack	**Modality**	**Threat model and parameters**
GPS spoofing	GPS/IMU	Civilian GPS lacks message authentication; commodity SDR transmitters override real signals (Tippenhauer et al., 2011; Humphreys, 2012). Per-frame uniform offset $\in [-10^{-3}, 10^{-3}]^\circ$ on latitude and longitude, i.e. up to $\approx$111 m and $\approx$55 m in expectation.
GPS gradual	GPS/IMU	Stealthy variant: linear ramp $0 \to \pm 10^{-3\circ}$ over the 1000-frame attack window, i.e. $\approx$0.111 m per frame and $\approx$111 m at the end, designed to evade threshold-based detectors (Shen et al., 2020).
RGB noise	RGB	Salt-and-pepper at $p = 0.05$ per pixel per camera, modeling sensor compromise via dust, EM interference, or ISP/firmware tampering.
Depth tampering	RGB (front)	Per-frame multiplicative scale $\in [0.5, 2.0]$ on the depth map of the depth camera co-located with the front RGB camera (Boloor et al., 2020). The depth map is folded into the front-camera representation (Appendix G.4), so the attack reaches SensorAD through the RGB modality, and the per-sensor camera detectors score the same stream.
LiDAR tampering	LiDAR	500 phantom points injected in a $20\times20\times3$ m box at $(+20, 0, 0.5)$ m, modeling laser-relay attacks (Cao et al., 2019; Sun et al., 2020; Shin et al., 2017).
DVS noise	DVS	2000 random events per frame, modeling event flooding from stray IR or laser interference.
DoS	RGB+LiDAR	RGB and LiDAR streams replaced by zeros (cable failure, jamming, bus denial).
GPS static offset	GPS/IMU	Constant $+5\times10^{-4\circ}$ offset added to the reported GPS longitude for the whole attack window ($\approx$56 m due east at this map georeference), modeling a compromised pose message on a shared vehicle bus (Ivanov et al., 2016). Because the offset is constant, it translates the trajectory without altering its shape: velocity, acceleration and heading remain mutually consistent and only absolute position is wrong.
GPS + static	GPS/IMU compound	Union of the GPS spoofing and GPS static-offset attacks above (the latter is named `steering_bias` in the released code, retained for compatibility): a per-frame spoof superimposed on a constant offset.
RGB + LiDAR	RGB and LiDAR	Union of `rgb_noise` and `lidar_tampering`: vision blinding combined with phantom injection.

G.3 Train/Evaluation Split

Evaluation uses the 132 attack episodes together with the 36 attack-free episodes, all held out. Attack labels are never used for training or for threshold calibration. Architecture, loss weights, masking probabilities and modality subsets were selected on a *separate* pool of 30 CARLA attack episodes, three for each of the same ten attack types, generated with the same injection pipeline but with episode identifiers and route seeds disjoint from the 132 reporting episodes. Model selection therefore does use attack labels, and we do not claim label-freedom for it; what we do claim is that no benchmark episode influenced any hyperparameter or architecture choice, and that every value reported in this paper is scored on the benchmark rather than on the selection pool. Pre-training uses nuScenes (no CARLA overlap). Fine-tuning uses a separate pool of $\sim$200 CARLA *normal* episodes spanning the same towns and weather but with disjoint route seeds, and detection thresholds are calibrated on a held-out split of that pool. No episode used for fine-tuning or calibration appears in the evaluation set, so the benchmark is held out throughout the pipeline.

G.4 Data Preprocessing

nuScenes. Cameras resized to 224×224 and normalized to $[0, 1]$. LiDAR filtered to $[-50, 50]^2 \times [-5, 3]$ m, subsampled to 50,000 points, zero-padded to a 100-point minimum. GPS/IMU is constructed per clip from ego-pose: position from translation, velocity and acceleration from finite differences, angular velocity from quaternion-derived yaw differences, heading from yaw, speed from velocity magnitude. Interpolated to 50 timesteps per window. nuScenes has no depth camera, so the front view carries no depth map in Phase 1.

CARLA. Cameras and LiDAR processed as above. The depth map of the depth camera co-located with the front RGB camera is folded into the front-camera representation, so the RGB modality carries depth for the front view and any tampering with the depth stream changes SensorAD's RGB tokens. DVS events stored as compressed `.npz` with (x, y, t, pol), loaded as event volumes of shape $(10, 224, 224)$. GPS/IMU read directly from CARLA's APIs.

Licenses. The external assets used in this work are: nuScenes v1.0-trainval for real-world pre-training, released under CC BY-NC-SA 4.0 for non-commercial research use; CARLA 0.9.15 for simulation and controlled attack generation, released under the MIT License; and DINOv2 ViT-B/14 as the frozen RGB backbone, released under the Apache License 2.0. All assets were used for non-commercial research purposes and according to their respective terms of use.

G.5 Broader Impact and Safeguards

SensorAD is intended as a defensive method for improving autonomous-vehicle sensor robustness by detecting compromised, degraded, or faulty sensors before they affect downstream perception, localization, or planning. Since training and threshold calibration use only normal driving data, with attack labels entering solely at model selection and evaluation, it may reduce the cost of developing safety monitors for new sensor suites and domains. The work also has dual-use risk because simulated attack benchmarks and detection results may reveal which attack types are harder to detect. We mitigate this by framing SensorAD as a simulation-based defensive study, avoiding deployment claims, and clearly noting that it is not a certified defense against white-box adaptive attackers or physical real-world attacks. Real deployment would require closed-course testing, hardware-in-the-loop validation, environment-conditioned threshold calibration, fail-safe planning integration, and human oversight.

H Qualitative Score Traces

H.1 Per-Modality Score Traces

Per-modality scores $a_k(t)$ are z-scored using held-out normal statistics. Each clip spans two frames (stride 2), so the shaded regions in Figures 7 to 9 (clips 250 to 750) are the 1000-frame attack window. The episodes are chosen to illustrate attribution rather than typical delay.

Under GPS spoofing (Fig. 7), the GPS score rises to a plateau at $z \approx 5$ for much of the attack window. This episode responds later than the median episode of Table 8, and a second GPS excursion near clip 970, after the window, is a false alarm of the kind counted in Table 20. While the GPS score is elevated, LiDAR stays near $z \approx -1$ and DVS near $z \approx 0$, i.e. at or slightly below the benign level: the elevation is specific to the attacked GPS pathway rather than diffused across modalities, which is what makes $\arg\max_k a_k(t)$ a usable attribution rule. Under `lidar_tampering` (Fig. 8), the LiDAR score rises at attack onset to $z \in [+2, +3]$ and leads the other modalities for most of the window, apart from a transient GPS excursion near clip 680. The coordinated `rgb_noise+lidar_tampering` episode (Fig. 9) is a hard case: with two sensors compromised, the overall score stays only slightly above its benign level inside the window, consistent with the lower AUROC of this setting (0.717, Table 7), and a GPS excursion after the window is again a false alarm.

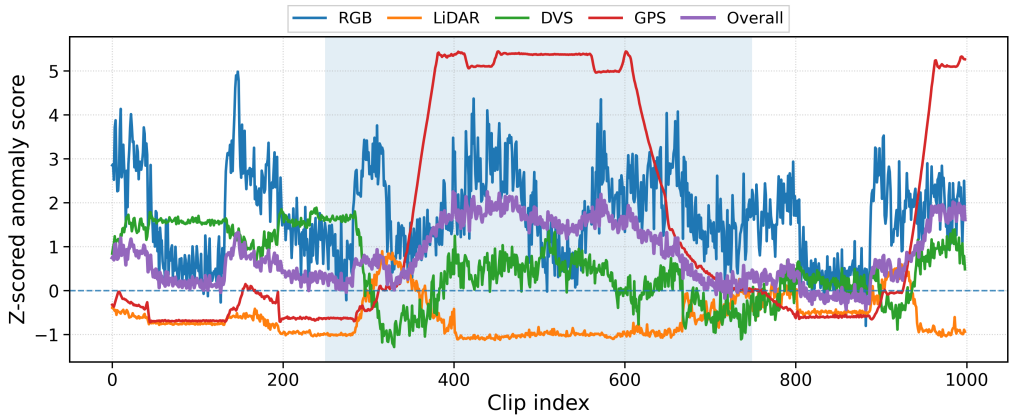


Figure 7: **GPS spoofing.** Per-modality and overall anomaly-score traces (z-scored).

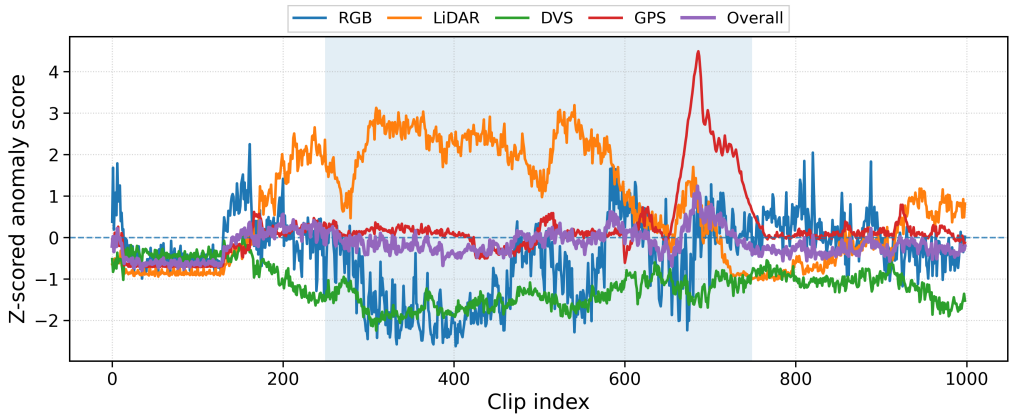


Figure 8: **LiDAR tampering.** Per-modality and overall anomaly-score traces.

H.2 Representation-Space Structure

Figures 10 and 11 visualize the structure of the unified transformer's `[CLS]` representations on held-out clips using t-SNE. Figure 10 colors per-modality `[CLS]` tokens by their source modality. The clusters are mostly mixed across modalities rather than separated by color, indicating that co-occurring sensor readings are mapped to nearby points in the shared space regardless of which modality they came from. This is the intended effect of the contrastive alignment loss $\mathcal{L}_{\text{align}}$ (Section 3.3): if modalities were sharply separable in this space, cross-modal prediction would degrade to a within-modality task. A small region in the lower-center of the plot does show some modality stratification, suggesting that alignment is strong on average but imperfect for a subset of clips.

Figure 11 colors the same multimodal `[CLS]` tokens by attack/normal label. Attack clips concentrate in several compact regions of the embedding space that are largely disjoint from the normal-clip

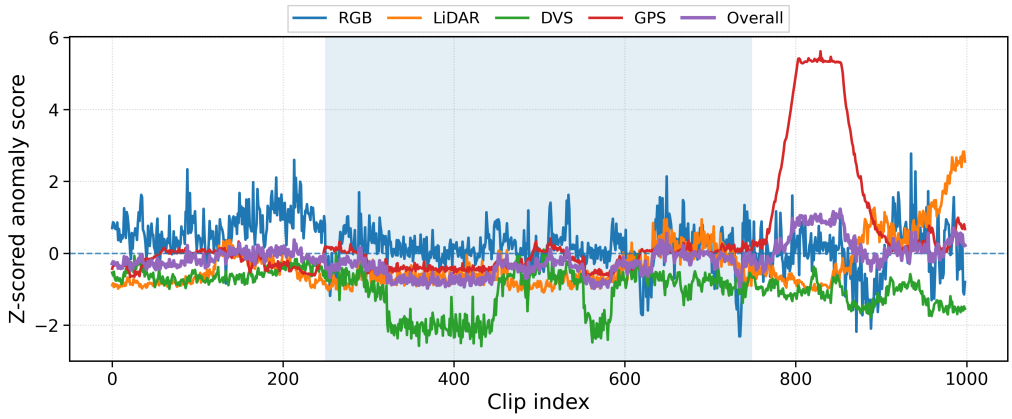


Figure 9: **Coordinated `rgb_noise+lidar_tampering`.** A hard case: the overall score stays only slightly above its benign level inside the window.

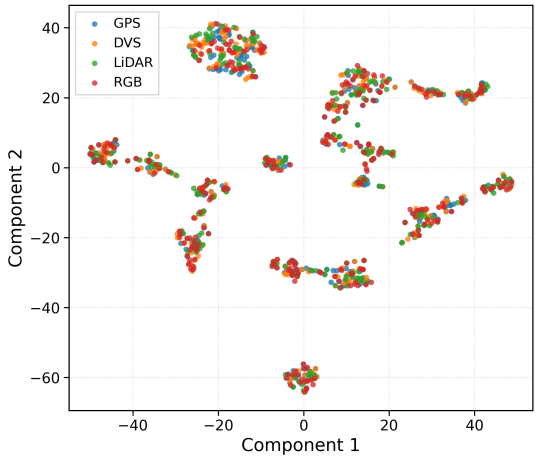


Figure 10: **t-SNE of Per-Modality `[CLS]` Embeddings, Colored by Modality.** Tokens from different modalities are largely co-located in the shared embedding space. A modality-stratified region appears in the lower-center, indicating that alignment is strong on average but not uniform across clips.

clusters. A mixed transition region is visible where attack and normal points overlap, which is consistent with the non-saturated ranking quality reported in Table 1: SensorAD separates most attack and normal clips in the representation space, but a fraction of attack clips remain embedded inside or near normal clusters and are responsible for the residual error in the AUROC and FPR@95 metrics.

I Baseline Implementation Details

This appendix describes the implementation of all baselines used in the main experiments. All baselines are trained only on normal data and evaluated on the same attack episodes as SensorAD without attack labels. For a fair comparison, methods that operate on learned representations use the same frozen modality tokenizers as SensorAD, while per-sensor visual anomaly detectors operate directly on modality-specific inputs.

Notation. Let x_t^m denote the raw input from modality $m \in \{\text{RGB}, \text{LiDAR}, \text{DVS}, \text{GPS/IMU}\}$ at time t. Let

$$z_t^m = g_m(x_t^m) \in \mathbb{R}^d \tag{9}$$

denote the corresponding modality-level feature extracted by the frozen tokenizer g_m. For methods that use all modalities jointly, we concatenate the per-modality features as

$$z_t = [z_t^{\text{RGB}}; z_t^{\text{LiDAR}}; z_t^{\text{DVS}}; z_t^{\text{GPS/IMU}}] \in \mathbb{R}^{4d}. \tag{10}$$

Each baseline produces an anomaly score s_t, where larger values indicate more anomalous frames.

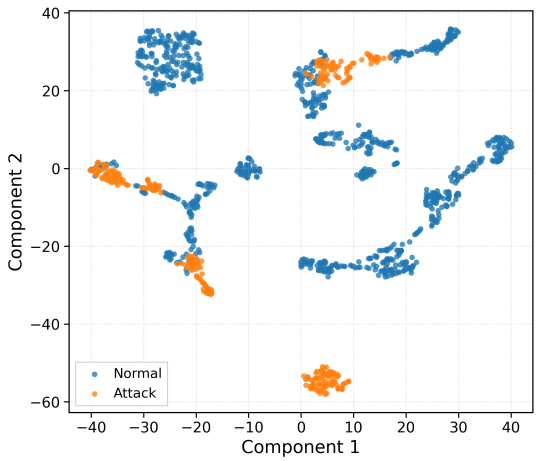


Figure 11: **t-SNE of Multimodal [CLS] Embeddings, Colored by Attack vs. Normal.** Attack clips concentrate in several compact regions, largely disjoint from normal clusters. A mixed transition region remains, accounting for the residual ranking error reported in Table 1.

Per-sensor anomaly detection baselines. EfficientAD, PatchCore, and PaDiM are trained independently for each modality using normal training data only. We use the standard Anomalib implementations and default training recipes unless stated otherwise. Camera inputs are processed as image frames after the same preprocessing as SensorAD, so the front view includes the folded depth map; DVS inputs are converted into rasterized event-volume images, LiDAR point clouds are converted into BEV pseudo-images, and GPS/IMU is evaluated as a raw time-series anomaly signal. The per-sensor camera model is therefore the one that scores both RGB noise and depth tampering in Table 1, and its PatchCore instance is the PatchCore-Cam detector used in the ensemble (Appendix C.2). A note on the DoS column: on DoS, where RGB and LiDAR are replaced by zeros, all three per-sensor detectors score near chance (0.50 to 0.51, Table 1). This is a normalization effect rather than a detection failure. Anomalib min-max normalizes anomaly maps against calibration statistics, and an all-zero input produces a spatially uniform map that ties at the normalized ceiling for both attacked and normal frames, collapsing the ranking. Their DoS column should therefore be read as uninformative rather than as evidence that per-sensor detectors cannot see a zeroed stream; Section 4.2 reports the physics-grounded average with DoS excluded for this reason.

For each modality m, the detector produces a per-modality anomaly score

$$s_t^m = \phi_m(x_t^m), \tag{11}$$

where ϕ_m is the trained single-modality detector. For single-sensor evaluation, the score corresponding to the attacked modality is used. For multi-sensor aggregation, we report the maximum score across modalities:

$$s_{\max}(x) = \max_m \frac{s_m(x) - \mu_m^{\text{val}}}{\sigma_m^{\text{val}} + \epsilon}, \tag{12}$$

where μ_m^{val} and σ_m^{val} are computed from normal validation scores, which prevents modalities with larger raw score scales from dominating the fusion. This max-fusion variant is included as a separate baseline because it represents the strongest simple combination of independent per-sensor detectors.

EfficientAD. EfficientAD is implemented as a teacher-student feature-distillation detector. Let $T_m(\cdot)$ be the frozen teacher network and $S_m(\cdot)$ the student network for modality m. The student is trained on normal samples to match the teacher representation:

$$\mathcal{L}_{\text{EffAD}}^m = \frac{1}{N} \sum_{t=1}^{N} \|S_m(x_t^m) - T_m(x_t^m)\|_2^2 . \tag{13}$$

At test time, the anomaly score is the feature discrepancy:

$$s_t^m = \|S_m(x_t^m) - T_m(x_t^m)\|_2^2 . \tag{14}$$

For spatial modalities, the pixel-level anomaly map is averaged to obtain a frame-level score.

PatchCore. PatchCore constructs a memory bank of normal patch embeddings. For each normal training input x_t^m, patch features are extracted as

$$P_t^m = \{p_{t,1}^m, \ldots, p_{t,n}^m\}. \tag{15}$$

The memory bank is

$$\mathcal{B}_m = \{p_{t,i}^m : x_t^m \in \mathcal{D}_{\text{normal}}, i = 1, \ldots, n\}, \tag{16}$$

with coreset subsampling as provided by Anomalib. At inference, each patch is scored by nearest-neighbor distance to the memory bank:

$$d_{t,i}^m = \min_{b \in \mathcal{B}_m} \left\| p_{t,i}^m - b \right\|_2. \tag{17}$$

The frame-level anomaly score is the maximum patch distance:

$$s_t^m = \max_i d_{t,i}^m. \tag{18}$$

This makes PatchCore especially strong for localized visual corruption, such as RGB noise and depth-map artifacts.

PaDiM. PaDiM models the distribution of normal patch features with a multivariate Gaussian. For each spatial location i and modality m, normal features are modeled as

$$p_{t,i}^m \sim \mathcal{N}(\mu_i^m, \Sigma_i^m). \tag{19}$$

At inference, the patch anomaly score is the Mahalanobis distance:

$$d_{t,i}^m = \left(p_{t,i}^m - \mu_i^m\right)^\top (\Sigma_i^m)^{-1} \left(p_{t,i}^m - \mu_i^m\right). \tag{20}$$

The frame-level score is

$$s_t^m = \max_i d_{t,i}^m. \tag{21}$$

As with PatchCore, PaDiM is applied independently to each sensor-derived image representation.

GPS/IMU raw time-series baseline. For GPS/IMU, we include a simple multivariate statistical anomaly baseline on the raw temporal signal. Let $u_t \in \mathbb{R}^q$ denote the GPS/IMU vector at time t, including position, velocity, acceleration, heading, angular velocity, and speed. We estimate the normal mean and covariance from training data:

$$\mu_u = \frac{1}{N} \sum_{t=1}^{N} u_t, \qquad \Sigma_u = \frac{1}{N-1} \sum_{t=1}^{N} (u_t - \mu_u)(u_t - \mu_u)^\top. \tag{22}$$

The anomaly score is

$$s_t^{\text{GPS/IMU}} = (u_t - \mu_u)^\top \Sigma_u^{-1} (u_t - \mu_u). \tag{23}$$

This baseline captures large deviations in GPS/IMU values but cannot detect attacks that remain statistically plausible while becoming inconsistent with other modalities.

Anomaly Transformer. The Anomaly Transformer baseline is applied to the temporal sequence of concatenated modality-level features. Given a window of length $W = 50$,

$$Z_{t-W+1:t} = [z_{t-W+1}, \ldots, z_t], \qquad z_t \in \mathbb{R}^{4d}, \tag{24}$$

the model reconstructs or predicts normal temporal patterns using self-attention. We use a 3-layer transformer with the default association-discrepancy objective. The reconstruction loss is

$$\mathcal{L}_{\text{rec}} = \frac{1}{W} \sum_{\tau=t-W+1}^{t} \|\hat{z}_\tau - z_\tau\|_2^2. \tag{25}$$

The anomaly score combines reconstruction error and association discrepancy:

$$s_t = \mathcal{L}_{\text{rec}}(t) + \alpha \mathcal{D}_{\text{assoc}}(t), \tag{26}$$

where $\mathcal{D}_{\text{assoc}}$ is the discrepancy between prior and learned attention associations, and α is the default weighting coefficient. This baseline tests whether temporal irregularity alone is sufficient for detecting sensor attacks.

LSTM autoencoder. The LSTM-AE baseline is trained on normal sequences of concatenated `[CLS]` features. The encoder is a 2-layer bidirectional LSTM with hidden size 256, and the decoder is a 2-layer LSTM trained to reconstruct the input sequence. For a window $Z_{t-W+1:t}$, the encoder produces a latent state

$$h_t = \mathrm{LSTM}_{\mathrm{enc}}(Z_{t-W+1:t}), \tag{27}$$

and the decoder reconstructs the feature sequence:

$$\hat{Z}_{t-W+1:t} = \mathrm{LSTM}_{\mathrm{dec}}(h_t). \tag{28}$$

The training objective is

$$\mathcal{L}_{\mathrm{LSTM}} = \frac{1}{W} \sum_{\tau=t-W+1}^{t} \|\hat{z}_\tau - z_\tau\|_2^2 . \tag{29}$$

At inference, the anomaly score is the final-frame reconstruction error:

$$s_t = \|\hat{z}_t - z_t\|_2^2 . \tag{30}$$

This baseline captures temporal deviations in the joint feature space but does not explicitly learn cross-modal prediction pathways.

Crossmodal Feature Mapping. The CFM baseline is the closest cross-modal baseline. We train pairwise mapping networks between all ordered modality pairs. For modalities i and j, a 3-layer MLP $F_{i\to j}$ predicts the feature of modality j from modality i:

$$\hat{z}_t^j = F_{i\to j}(z_t^i). \tag{31}$$

Each mapping network is trained on normal data with the objective

$$\mathcal{L}_{i\to j} = \frac{1}{N} \sum_{t=1}^{N} \left\| F_{i\to j}(z_t^i) - z_t^j \right\|_2^2 . \tag{32}$$

With four modalities, there are $\binom{4}{2} = 6$ unordered pairs or $4(4-1) = 12$ directed mappings. We use directed mappings and aggregate both directions for each pair. At inference, the modality-level score for target modality j is

$$s_t^j = \frac{1}{M-1} \sum_{i\neq j} \left\| F_{i\to j}(z_t^i) - z_t^j \right\|_2^2 . \tag{33}$$

The final frame-level score is

$$s_t = \frac{1}{M} \sum_{j=1}^{M} s_t^j . \tag{34}$$

Unlike SensorAD, CFM relies on independent pairwise mappings and does not jointly condition on the full set of remaining sensors. It also does not include modality-level masking, shared transformer attention, temporal prediction, or physics consistency losses.

Mahalanobis joint-statistical baseline. The Mahalanobis baseline models the distribution of normal concatenated multi-modal features with a single Gaussian:

$$z_t \sim \mathcal{N}(\mu_z, \Sigma_z). \tag{35}$$

The parameters are estimated from normal training data:

$$\mu_z = \frac{1}{N} \sum_{t=1}^{N} z_t, \qquad \Sigma_z = \frac{1}{N-1} \sum_{t=1}^{N} (z_t - \mu_z)(z_t - \mu_z)^\top . \tag{36}$$

The anomaly score is

$$s_t = (z_t - \mu_z)^\top (\Sigma_z + \lambda I)^{-1} (z_t - \mu_z), \tag{37}$$

where a small ridge term λI is added for numerical stability. This baseline tests whether attacks can be detected by global deviation from the normal joint feature distribution.

OC-SVM. The one-class SVM baseline is trained on concatenated `[CLS]` features z_t. We use an RBF kernel

$$K(z_i, z_j) = \exp(-\gamma \|z_i - z_j\|_2^2), \tag{38}$$

with $\nu = 0.01$, corresponding to the target false-positive rate used for threshold calibration. The one-class SVM learns a decision function

$$f(z_t) = \sum_{i=1}^{N} \alpha_i K(z_i, z_t) - \rho. \tag{39}$$

The anomaly score is the negative decision value:

$$s_t = -f(z_t). \tag{40}$$

This baseline captures nonlinear deviations in the joint feature space but does not explicitly model which modality is inconsistent with the others.

Thresholding and evaluation. For all baselines, thresholds are selected using the held-out normal validation set. Unless otherwise specified, the threshold is set to the 99th percentile of normal validation scores:

$$\tau = Q_{0.99}\left(\{s_t : x_t \in \mathcal{D}_{\text{val}}^{\text{normal}}\}\right). \tag{41}$$

A frame is classified as anomalous when

$$\mathbb{I}_{\text{attack}}(t) = \mathbf{1}[s_t > \tau]. \tag{42}$$

For AUROC and AUPRC, the raw anomaly scores are used directly without thresholding. For FPR@95, the threshold is chosen to achieve 95% TPR on the attack split, and the corresponding false-positive rate is computed on normal frames.

Single-modality temporal baselines. To separate temporal modeling from cross-modal conditioning we add two GPS/IMU-only detectors, trained normal-only on the same 50-step window with the same threshold protocol: (i) the static multivariate Mahalanobis detector of Eqs. (22) and (23), and (ii) an LSTM autoencoder with the same architecture as the multimodal LSTM-AE but consuming only the raw 11-dim GPS/IMU sequence. Results are in Appendix D.1.

Implementation fairness, and its limits. All representation-based baselines use the same frozen SensorAD tokenizers to avoid confounding the comparison with feature-extractor quality. Two asymmetries remain, and they cut in different directions. First, CFM, Mahalanobis, OC-SVM and the LSTM-AE consume pooled per-modality `[CLS]` vectors, whereas SensorAD's transformer attends over up to 1,194 dense tokens per clip; pooling deprives those baselines of fine-grained spatial structure, so the reported margins are evidence for the full design rather than for joint conditioning in isolation (Appendix D.2 controls for this from SensorAD's side). Second, the per-sensor image detectors run on raw modality-specific inputs rather than on our frozen tokenizer outputs, since they are designed for spatial anomaly detection; that choice favours them on localized visual corruption. All methods are trained on the same normal training split and evaluated on the same CARLA attack episodes. No baseline is trained on attack labels.

J Computational Analysis

This appendix summarizes the computational profile of SensorAD. All reported measurements are from a single NVIDIA RTX 3090 GPU with 24 GB VRAM and an AMD Ryzen 9 5950X CPU. Training and inference use mixed precision. Gradient checkpointing is enabled during training to keep memory usage within the single-GPU budget.

J.1 Parameter Budget

SensorAD has 149.4M total parameters (Table 27), of which 62.8M are trainable during CARLA fine-tuning. The remaining 86.6M parameters come from the frozen DINOv2 ViT-B/14 RGB backbone (Oquab et al., 2024). Freezing the RGB backbone reduces memory usage and avoids overfitting while preserving strong visual features for cross-modal prediction.

Table 27: **SensorAD parameter budget.** The RGB backbone is frozen and is not counted toward the trainable total.

Component	Parameters
Frozen RGB backbone: DINOv2 ViT-B/14	86.6M
Cross-modal transformer encoder	44.1M
Reconstruction and auxiliary heads	12.8M
LiDAR, DVS, and GPS/IMU tokenizers	5.9M
Modality, camera, position, and mask embeddings	0.045M
Trainable total	**62.8M**
Frozen total	**86.6M**
Overall total	**149.4M**

Table 28: **Inference latency on RTX 3090.** Latency is measured for batch size 1 with mixed precision. Values are medians over 20 repeated measurements after 3 warmup iterations.

Computation	Latency per clip
All tokenizers	31.6 ms
Single-pass inference	45.3 ms
Five-pass localized scoring	106.7 ms
Single-pass throughput	22.1 FPS
Single-pass score $A_1(t)$ throughput	21.8 FPS
Localized-scoring throughput	9.4 FPS

Most trainable parameters are concentrated in the cross-modal transformer, which is responsible for modeling consistency across RGB, LiDAR, DVS, and GPS/IMU. The reconstruction and auxiliary heads are the second largest trainable component because they include modality-specific prediction heads used for cross-modal reconstruction and scoring. The modality, camera, position, and mask embeddings are negligible in parameter count relative to the transformer and tokenizers.

J.2 Inference Runtime

SensorAD supports two inference modes. A single forward pass produces fused multi-modal representations for global scoring. Sensor-localized attack detection uses one unmasked reference pass and one masked pass per modality. With four modalities, this gives five transformer passes per clip. The tokenizers are executed once and their outputs are reused across masked passes.

Table 29 breaks the single pass down further. Tokenization dominates: 31.6 of the 45.3 ms, of which 24.7 ms is the frozen DINOv2 encoder that a deployed AV would run regardless. Only 12.8 ms (transformer) plus 0.9 ms (heads) is specific to cross-modal scoring. Sharing the image backbone with the perception stack would leave the detector at roughly 20 ms; this is an estimate obtained by subtraction, not a measurement in a shared-backbone configuration. Because the tokenizers execute once and the four masked passes batch into a single call, five-pass scoring costs $2.4\times$ rather than $5\times$ a single pass.

As shown in Table 28, the frozen RGB backbone and repeated transformer passes dominate runtime. Single-pass inference takes 45.3 ms (22.1 FPS), while five-pass localized scoring takes 106.7 ms (9.4 FPS). The latter is best viewed as an attribution-oriented diagnostic mode rather than a fully optimized real-time implementation. In deployment, the localized scorer can be invoked only after a single-pass anomaly trigger or at a lower temporal frequency. The masked passes can also be batched together, reducing launch overhead compared with five independent inference calls.

Detection-delay results in Appendix B.4 measure algorithmic delay only and exclude wall-clock latency. SensorAD is computationally heavier than per-sensor baselines because it explicitly evaluates cross-modal consistency for each modality. Pairwise methods such as CFM are closer computationally but use independent modality mappings rather than a shared all-to-one transformer. SensorAD therefore trades additional computation for unified detection and sensor-level localization.

Table 29: **Per-component inference latency** (RTX 3090, batch 1, BF16, four camera views batched as one input; medians over 20 runs after 3 warmup iterations).

Component	ms	Note
DINOv2 ViT-B/14, four views	24.7	frozen; shared with any image perception head
PointPillars BEV	5.1	shared with any BEV head
DVS stem + GPS/IMU projection	1.8	
Tokenizer subtotal	31.6	executed once per clip, reused across passes
Transformer, one pass	12.8	the only cross-modal-specific cost
Reconstruction heads	0.9	
Single fused pass	45.3	
Single-pass score $A_1(t)$	45.8	adds per-head scoring on the reference pass
Five-pass localized score $A(t)$	106.7	$2.4\times$, not $5\times$

J.3 Training Cost

Table 30 reports the wall-clock cost of the two training stages. Phase 1 trains the model from scratch on nuScenes with RGB, LiDAR, and GPS/IMU. Phase 2 starts from the Phase 1 checkpoint, introduces the DVS tokenizer, and adapts the model to CARLA.

Table 30: **Training and inference cost on RTX 3090.** Training time includes per-epoch validation; inference latency uses batch size 1 with mixed precision.

Stage	Epochs / setting	Wall-clock / latency	Peak memory
Phase 1: nuScenes pre-training	30 epochs	28.4 h	24 GB
Phase 2: CARLA fine-tuning	20 epochs	74.1 h	21 GB
Single-pass inference	batch size 1	45.3 ms	0.61 GB
Localized scoring	batch size 1	106.7 ms	0.63 GB

Fine-tuning is slower than pre-training because CARLA training uses all four modalities, including DVS, over roughly ten times as many frames, with per-epoch validation. Transformer activations dominate memory usage, requiring gradient checkpointing for single-GPU training. Larger runs can use standard DistributedDataParallel.