Enhancing Robustness and Security of Edge AI Systems for Safety-Critical Applications

WP2 – Comprehensive Toolkit for Robust Edge AI

D2.2: Report on adversarial, context-aware and anomaly detection techniques – Initial version

This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No. 101168067. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Document Information

GRANT AGREEMENT NUMBER	101168067	**ACRONYM**	GuardAI
FULL TITLE	Enhancing Robustness and Security of Edge AI Systems for Safety-Critical Applications		
START DATE	1st October 2024	**DURATION**	36 months
PROJECT URL	https://www.kios.ucy.ac.cy/guardai/		
DELIVERABLE	D2.2: Report on adversarial, context-aware and anomaly detection techniques – Initial version		
WORK PACKAGE	WP2 – Comprehensive Toolkit for Robust Edge AI		
DATE OF DELIVERY	**CONTRACTUAL** 1/2026	**ACTUAL**	1/2026
TYPE	Report	**DISSEMINATION LEVEL**	PU
LEAD BENEFICIARY	ATHENA		
RESPONSIBLE AUTHORS	Erion-Vasilis Pikoulis, Christos Mavrokefalidis, Aris Lalos		
CONTRIBUTIONS (FROM)	Maria Damanaki (ATHENA), Ioulia Kapsali (ATHENA), Nikos Potamianos (ATHENA), Erion-Vasilis Pikoulis (ATHENA), Christos Mavrokefalidis (ATHENA), Alexandros Gkillas (ATHENA), Aris Lalos (ATHENA), Daniel Bethell (YORK), Charmaine Barker (YORK), Simos Gerasimou (YORK), Amalia Damianou (CERTH), George Lazaridis (CERTH), Grigoris Kalogiannis (CERTH), Christos Kyrkou (UCY-KIOS CoE), Rafaella Elia (UCY-KIOS CoE), Mehmet Demirel (UCY-KIOS CoE), Yeshwanth Adimoolam (UCY-KIOS CoE), Antonis Savva (UCY-KIOS CoE), Artemis Androni (SPH)		
ABSTRACT	This deliverable reports the initial set of techniques investigated in WP2 of GuardAI pertaining to adversarial resilience, context-aware multi-X protection mechanisms, anomaly detection, and holixtic protection approaches in AI-enabled perception and decision pipelines. It surveys relevant threat models and attack vectors and presents defensive mechanisms spanning RGB and LiDAR perception (e.g., purification-based defenses for semantic segmentation, training-free recovery from adversarial patches in object detection, and robustness considerations for point-cloud segmentation), complemented by data-centric measures addressing augmentation, poisoning, and similarity-based attack detection. The document also outlines evaluation considerations and practical constraints for deployment, providing a foundation for the next integration and validation steps within the project.		

Document History

VERSION	ISSUE DATE	STAGE	DESCRIPTION	CONTRIBUTOR
V 0.1	28/11/2025	Draft	ToC	Erion-Vasilis Pikoulis (ATHENA), Christos Mavrokefalidis (ATHENA), Christos Kyrkou (UCY-KIOS CoE), Rafaella Elia (UCY-KIOS CoE), Mehmet Demirel (UCY-KIOS CoE), Antonis Savva (UCY-KIOS CoE)
V 0.2	17/12/2025	Draft	First Merged version	Christos Kyrkou (UCY-KIOS CoE), Rafaella Elia (UCY-KIOS CoE), Mehmet Demirel (UCY-KIOS CoE), Yeshwanth Adimoolam (UCY-KIOS CoE), Antonis Savva (UCY-KIOS CoE), Maria Damanaki (ATHENA), Ioulia Kapsali (ATHENA), Nikos Potamianos (ATHENA), Erion-Vasilis Pikoulis (ATHENA), Christos Mavrokefalidis (ATHENA), Alexandros Gkillas (ATHENA), Aris Lalos (ATHENA), Daniel Bethell (YORK), Charmaine Barker (YORK), Simos Gerasimou (YORK), Amalia Damianou (CERTH), George Lazaridis (CERTH), Grigoris Kalogiannis (CERTH)
V 0.3	23/12/2025	Draft	Deliverable version for internal review	Erion-Vasilis Pikoulis (ATHENA), Christos Mavrokefalidis (ATHENA)
V 0.4	19/1/2025	Draft	Review completed	Daniel Bethell (YORK), Charmaine Barker (YORK), Simos Gerasimou (YORK), Amalia Damianou (CERTH), George Lazaridis (CERTH), Grigoris Kalogiannis (CERTH)
V 1.0	31/1/2026	Final	Internal review comments addressed, formatting, final version ready	Christos Kyrkou (UCY-KIOS CoE), Rafaella Elia (UCY-KIOS CoE), Mehmet Demirel (UCY-KIOS CoE), Yeshwanth Adimoolam (UCY-KIOS CoE), Antonis Savva (UCY-KIOS CoE), Maria Damanaki (ATHENA), Ioulia Kapsali (ATHENA), Nikos Potamianos (ATHENA), Erion-Vasilis Pikoulis (ATHENA), Christos Mavrokefalidis (ATHENA), Alexandros Gkillas (ATHENA), Aris Lalos (ATHENA), Daniel Bethell (YORK),

				Charmaine Barker (YORK), Simos Gerasimou (YORK), Amalia Damianou (CERTH), George Lazaridis (CERTH), Grigoris Kalogiannis (CERTH), Artemis Androni (SPH)

Disclaimer

Copyright message

Acronym	Meaning
AI	Artificial Intelligence
AID	Aerial Image Dataset
AGE	Attack Generation Engine
AP	Average Precision
AT	Adversarial Training
BPDA	Backward Pass Differentiable Approximation
CAM	Cooperative Awareness Message
CAV	Connected and Vehicle
CC	Context Consistency
C-EDL	Conflict-aware Evidential Deep Learning
CMPA	Cross-Modal Patch Attack
CNN	Convolutional Neural Network
CR	Context Relevance
CSV	Comma Separated Values
CVGL	Cross-View Geo-Localization
CW	Carlini & Wagner
DL	Deep Learning
DDIM	Denoising Diffusion Implicit Model
DDPM	Denoising Diffusion Probabilistic Model
DR	Detection Rate
DU	Deep Unrolling
DU-AP	Deep Unrolling Adversarial Purification

EOT	Expectation over Transformation
EDL	Evidential Deep Learning
FGSM	Fast Gradient Sign Method
FPR	False Positive Rate
FPS	Frames per Second
GANs	Generative Adversarial Networks
GNSS	Global Navigation Satellite System
GUIDE	Gradual Uncertainty Refinement via Noise-Driven Curriculum
IDS	Intrusion Detection System
IoU	Intersection over Union
IR	Infrared
IMU	Inertial Measurement Unit
LiDAR	Light Detection & Ranging
Map	Mean Average Precision
MSE	Mean Squared Error
ML	Machine Learning
pcap	packet capture
PE	Perception Effectiveness
PGD	Projected Gradient Descent
RGB	Red, Green, Blue
RB	Reliability
RT	Response Time
RTK	Real Time Kinematic
SLC	Scenario Logic Core
SOC	Security Operations Center
SUMO	Simulation of Urban MObility
TC	Temporal Coherence
UAV	Unmanned Aerial Vehicle
UC	Use Case
UGV	Unmanned Ground Vehicle
VIO	Visual Inertial Odometry

Table of Contents

1. Executive summary

This deliverable (D2.2) provides the initial survey, design rationale, and early experimental baselines for the GuardAI toolkit components targeting adversarial robustness, context-aware perception, and anomaly detection in safety-critical edge AI systems. It consolidates techniques developed across the consortium for camera, LiDAR, and multi-modal perception, focusing on methods that are effective under realistic threat models and feasible for deployment on resource-constrained platforms.

The report is organized around the activities of the four relevant tasks in WP2, pertaining to: (i) enhanced adversarial resiliency, where inputs or internal representations are hardened against intentional perturbations (T2.2); (ii) robustness through Multi-X context awareness, where cross-modal, cross-view, and multi-agent consistency is used to detect or mitigate attacks and operational corner cases (T2.3); (iii) anomaly detection, where system-level monitoring and uncertainty estimation help identify malicious behaviour, degradation, and out-of-distribution conditions that may compromise autonomy (T2.4); and (iv) holistic protection combining adversarial resilience with context-aware mitigation mechanisms (T2.5).

Key outcomes documented in this deliverable include:

- *Enhanced Adversarial Resiliency:* Achieves significantly improved robustness of RGB and LiDAR perception pipelines against adversarial attacks through lightweight and effective defense mechanisms.
- *Robustness Through Multi-X Context Awareness:* Delivers increased attack detection and perception reliability by exploiting multi-modal, multi-task, multi-agent and cross-view contextual consistency.
- *Anomaly Detection Framework:* Delivers an end-to-end anomaly detection capability combining attack simulation, AI-driven intrusion detection, uncertainty quantification, and 5G analytics.
- *Holistic Protection with Context and Robustness:* Unifies perception defenses and contextual reasoning into a scenario-driven, adaptive protection strategy for edge systems (CAVs, UAVs).

Overall, D2.2 establishes baseline designs and evaluation methodology for GuardAI's WP2 robustness toolkit and identifies the next integration steps towards end-to-end protected perception and decision-making pipelines in the project's target use cases.

2. Introduction

Edge AI systems in safety-critical domains (e.g., connected and autonomous vehicles, aerial robotics, and intelligent infrastructure) increasingly rely on deep learning models for perception and decision support. While these models offer strong nominal accuracy, they remain vulnerable to deliberate adversarial manipulation and to benign but unexpected operating conditions. In practice, small input perturbations, localized physical patches, sensor faults, communication tampering, and distribution shifts can all trigger incorrect predictions that propagate to planning and control, amplifying safety risk.

GuardAI addresses this gap by developing a comprehensive, deployable toolkit that strengthens robustness and security of edge AI components without sacrificing real-time constraints. Within this context, D2.2 documents the project's initial progress on three technical fronts: adversarial resiliency techniques that harden perception models against crafted attacks; context-aware mechanisms that leverage redundancy across modalities, views, and agents to expose inconsistencies; and anomaly detection modules that monitor behavior and uncertainty to flag abnormal conditions and potential intrusions.

In addition, D2.2 emphasizes solutions aligned with GuardAI's deployment constraints: limited compute and memory budgets, low-latency inference, and operation across heterogeneous sensors and platforms. The techniques reported here include lightweight model-based purification methods, efficient diffusion pipelines, representation-aware threat models (e.g., LiDAR range-view attacks), and training-free recovery strategies designed to generalize across datasets and attack variants.

The remainder of the report is structured as follows:

- Section 3 develops and validates lightweight, scalable defenses to improve robustness of RGB and LiDAR perception against adversarial attacks, including purification, saliency-guided recovery, imbalance-aware learning, adversarial patch analysis, and attack-aware dataset augmentation.
- Section 4 focuses on how multi-modal, multi-task, multi-agent, and cross-view context awareness significantly enhances perception robustness and attack detection through consistency checks, sensor fallback strategies, and collaborative situational awareness.
- Section 5 introduces the anomaly detection framework, covering adversarial attack simulation, AI-based IDS, uncertainty quantification, and 5G network analytics to identify and respond to cyber-physical threats in edge systems.
- Section 6 discusses how the above components can be composed into holistic protection strategies, highlighting integration directions within WP2 and with downstream work packages.
- Section 7 summarizes key conclusions and outlines next steps for maturing the techniques from initial baselines to integrated, validated toolkit components.

As an initial version, D2.2 prioritizes clarity of threat assumptions, reproducibility of evaluation setups, and identification of integration points. Subsequent iterations will expand experimental coverage, refine interfaces for the WP2 toolkit, and validate performance in progressively more realistic conditions and project pilots.

3. Enhanced Adversarial Resiliency

This section focuses on strengthening the robustness of GuardAI's perception components against adversarial perturbations, with emphasis on the two sensing modalities that dominate automated situational awareness: RGB cameras and LiDAR. The section motivates adversarial resiliency through concrete perception workloads (semantic segmentation and object detection), where small, carefully crafted perturbations (e.g., bounded pixel-level noise) or localized manipulations (e.g., adversarial patches) can cause disproportionate downstream impact. In this framing, resiliency is treated as an operational requirement: the perception stack should maintain usable performance under realistic attack conditions and fail in predictable, controllable ways when perfect recovery is not possible.

3.1. Adversarial Resiliency for RGB data: Benefits on Image Segmentation

Camera-based perception constitutes a core sensing modality in autonomous and assisted driving systems, providing dense visual information that supports scene understanding, localization, and decision-making[1]. In particular, semantic segmentation of RGB images plays a critical role in identifying drivable areas, lane markings, and scene structure, which are essential for downstream tasks such as semantic SLAM[2], and map-based localization. As these modules operate directly on pixel-level predictions, segmentation errors can propagate to planning and control, amplifying their impact on overall system safety.

3.1.1. Adversarial Threats and Defense Strategies for RGB segmentation

Adversarial attacks in the RGB domain are commonly formulated under bounded perturbation models, such as l_∞ or l_2 constraints and can be executed in both white-box and black-box settings. Gradient-based attacks, most notably Projected Gradient Descent (PGD)[3], are particularly effective against segmentation networks due to their ability to directly optimize dense loss functions across all pixels.

To mitigate these threats, several defense strategies have been proposed. Adversarial training[4,5] improves robustness by retraining segmentation networks on adversarially perturbed images. However, its practical applicability is limited by high computational cost[6], reliance on labeled adversarial data, and reduced generalization to unseen

[1] Seo, J.-H. (2025) 'Semantic segmentation and real-time tracking of vehicle drivable areas using a supervised deep learning approach', *Journal of Information & Communication Convergence Engineering*, 23(1).
[2] Zheng, C., Zhang, P. and Li, Y. (2025) 'Semantic SLAM system for mobile robots based on large visual model in complex environments', *Scientific Reports*, 15(1), p. 8450.
[3] Madry, A., Makelov, A., Schmidt, L., Tsipras, D. and Vladu, A. (2017) 'Towards deep learning models resistant to adversarial attacks', *arXiv preprint*, arXiv:1706.06083.
[4] Lau, C.P., Liu, J., Souri, H., Lin, W.-A., Feizi, S. and Chellappa, R. (2023) 'Interpolated joint space adversarial training for robust and generalizable defenses', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11), pp. 13054–13067.
[5] Zhao, M., Zhang, L., Ye, J., Lu, H., Yin, B. and Wang, X. (2024) 'Adversarial training: A survey', *arXiv preprint*, arXiv:2410.15042.
[6] Tsipras, D., Santurkar, S., Engstrom, L., Turner, A. and Madry, A. (2018) 'Robustness may be at odds with accuracy', *arXiv preprint*, arXiv:1805.12152.

attack types or perturbation budgets[7]. More recently, input purification methods have emerged as an attractive alternative, aiming to remove adversarial perturbations prior to inference, decoupling robustness from the segmentation model itself. State-of-the-art purification approaches largely rely on generative models, such as GANs[8,9,10] or diffusion models[11,12,13], which attempt to project adversarial inputs back onto the natural image manifold.

While generative purification methods demonstrate strong denoising capabilities, they are typically associated with extremely large model sizes[14] and iterative inference procedures and substantial computational overhead, often exceeding the computational footprint of the segmentation network itself, making real-time deployment on embedded automotive platforms impractical where latency, memory, and power consumption are tightly constrained.

3.1.2. LightPure: Lightweight Adversarial Resiliency for RGB Image Segmentation

Motivated by the limitations of adversarial training and heavyweight generative purification methods, we adopt a lightweight, model-based purification framework, termed LightPure[15], which formulates adversarial defense as a structured optimization problem rather than a black-box inverse mapping. This design is grounded in a key observation: under common bounded threat models, adversarial perturbations typically preserve the global semantic and visual structure of an image while selectively corrupting local, high-frequency components, such as edges and fine textures. These components are particularly critical for dense prediction tasks.

From a signal processing perspective, natural images admit sparse representations in multiscale transform domains, where dominant semantic structures concentrate in a small number of coefficients at coarser scales, whereas high-frequency coefficients

[7] Lin, W.-A., Lau, C.P., Levine, A., Chellappa, R. and Feizi, S. (2020) 'Dual manifold adversarial robustness: Defense against ℓp and non-ℓp adversarial attacks', *Advances in Neural Information Processing Systems*, 33, pp. 3487–3498.
[8] Jin, G., Shen, S., Zhang, D., Dai, F. and Zhang, Y. (2019) 'APE-GAN: Adversarial perturbation elimination with GAN', *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3842–3846.
[9] Samangouei, P., Kabkab, M. and Chellappa, R. (2018) 'Defense-GAN: Protecting classifiers against adversarial attacks using generative models', *arXiv preprint*, arXiv:1805.06605.
[10] Wang, Y., Liao, X., Cui, W. and Yang, Y. (2025) 'Defending against and generating adversarial examples together with generative adversarial networks', *Scientific Reports*, 15(1), p. 12994.
[11] Nie, W., Guo, B., Huang, Y., Xiao, C., Vahdat, A. and Anandkumar, A. (2022) 'Diffusion models for adversarial purification', *arXiv preprint*, arXiv:2205.07460. Available at: https://arxiv.org/abs/2205.07460
[12] Lei, C.T., Yam, H.M., Guo, Z., Qian, Y. and Lau, C.P. (2025) 'Instant adversarial purification with adversarial consistency distillation', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24331–24340.
[13] Li, Y., Li, Z., Huang, L., Hu, L., Zeng, L. and Shen, D. (2025) 'Adversarial purification with one-step guided diffusion model', *Neural Networks*, p. 107877.
[14] Kang, M., Song, D. and Li, B. (2023) 'DiffAttack: Evasion attacks against diffusion-based adversarial purification', *Advances in Neural Information Processing Systems*, 36, pp. 73919–73942.
[15] Kapsali I., Gkillas A, and Lalos A. S. (2026) 'A Lightweight Model-Based Method for Adversarial Purification in Autonomous Driving Segmentation', Submitted to the *28th International Conference on Pattern Recognition (ICPR).*

capture texture, noise-like patterns, and fine details[16,17]. Adversarial perturbations, in contrast, tend to manifest as low-magnitude coefficients distributed across high-frequency subbands. This spectral separation motivates the use of a wavelet-domain sparsity prior to suppress perturbation-dominated components in a principled and interpretable manner.

However, wavelet-domain shrinkage alone is insufficient, as fixed or statistically driven thresholds cannot reliably distinguish adversarial noise from legitimate fine-scale image details. To overcome this limitation, LightPure augments the wavelet sparsity prior with a compact learned CNN-based prior, which performs content-aware refinement and restores structural details weakened during shrinkage. By combining these complementary priors within a unified framework, LightPure achieves effective adversarial suppression while preserving task-relevant semantic information.

Threat Model and Attack Formulation

Let $X_c \in [0,1]^{3\times H\times W}$ denote a clean RGB image and let $f(\cdot)$ denote a semantic segmentation network producing dense per-pixel predictions. An adversarial image is generated as: $\boldsymbol{X}_n = X_c + \Delta$ where the perturbation Δ is constrained by:

$$\max_{\Delta} -L_{seg}(X_n, Y) \text{ s.t } \|\Delta\|_p \leq \varepsilon, \tag{1}$$

with $p \in \{2, \infty\}$, where L_{seg} denotes the segmentation loss funciton. The optimization problem is solved using PGD, yielding adversarial inputs that remain visually indistinguishable from the clean image while inducing incorrect segmentation predictions. The resulting adversarial image $\mathbf{X}_n$ serves as the input to the purification framework described next.

Model-Based Purification Framework

Given an adversarially perturbed image $\boldsymbol{X}_n$, the objective of LightPure is to recover a purified image $\boldsymbol{X}_r$ such that the segmentation output remains consistent with that of the clean input. This objective is formulated as:

$$\min_{X_r} \frac{1}{2}\|X_n - X_r\|_F^2 + \lambda R(X_r) + \gamma Q(X_r) \tag{2}$$

where each term serves a distinct and complementary role. The data fidelity term $\frac{1}{2}\|X_n - X_r\|_F^2$ (where $\|\cdot\|_F^2$ denotes the Frobenius norm) enforces consistency with the observed input, ensuring that the purified image remains close to the adversarial observation and preventing the introduction of hallucinated content. The regularizer $\mathcal{R}(\cdot)$ denotes a learned image prior, implemented via a lightweight convolutional neural network, which restores fine structural details and semantic textures that may be weakened during denoising. The regularizer $Q(\cdot)$represents a wavelet-domain sparsity prior, which suppresses perturbation-dominated high-frequency components by exploiting the sparse multiscale structure of natural images.

[16] Mustafa, A., Khan, S.H., Hayat, M., Shen, J. and Shao, L. (2019) 'Image super-resolution as a defense against adversarial attacks', *IEEE Transactions on Image Processing*, 29, pp. 1711–1724.
[17] Yin, D., Gontijo Lopes, R., Shlens, J., Cubuk, E.D. and Gilmer, J. (2019) 'A Fourier perspective on model robustness in computer vision', *Advances in Neural Information Processing Systems*, 32.

Rather than relying on a black-box generative mapping, this formulation explicitly integrates both handcrafted and learned priors within a principled optimization framework. The resulting problem is solved using a model-based iterative scheme, in which data consistency, wavelet-domain shrinkage, and learned refinement are alternated to progressively remove adversarial artifacts while preserving semantic content.

At each iteration k, LightPure first performs a data-consistency update, which balances the contribution of the adversarial input with the outputs of the two priors:

$$X_r^{(k+1)} = \frac{1}{1+2\beta}\left(X_n + \beta Z^{(k)} + \beta U^{(k)}\right) \quad (3)$$

This step ensures that the purified estimate remains anchored to the observed image while allowing gradual refinement through the regularization terms.

Next, a wavelet-domain sparsity update is applied to suppress perturbation-dominated high-frequency components. The current estimate is transformed into the wavelet domain, where sub-band–adaptive soft-thresholding is performed, followed by inverse transformation:

$$U^{(k+1)} = W^{-1}\left[\bigcup_{s=1}^{S} Soft(C_s, \tau_s)\right] \quad (4)$$

This operation exploits the sparsity of natural images in multiscale representations, effectively attenuating adversarial noise while preserving dominant structural information.

Finally, a learned refinement step restores fine-scale semantic details that may be weakened by wavelet shrinkage:

$$Z^{(k+1)} = \mathcal{G}_\theta\left(X_r^{(k+1)}\right) \quad (5)$$

Where $\mathcal{G}_\theta$ denotes a compact CNN-based denoiser trained to recover texture and boundary information in a content-aware manner.

These three steps are iterated for a fixed number of iterations and unrolled into a compact feed-forward architecture using a Deep Unrolling strategy, where each unrolled stage corresponds to one iteration of the underlying optimization process. This design yields an interpretable, lightweight, and computationally efficient purification module. The resulting purified image enables robust drivable-area and lane-line segmentation under adversarial conditions, while remaining suitable for real-time operation on embedded automotive platforms.

An overview of the proposed LightPure purification pipeline and its integration with the YOLOP segmentation model is illustrated in Figure 3.1.

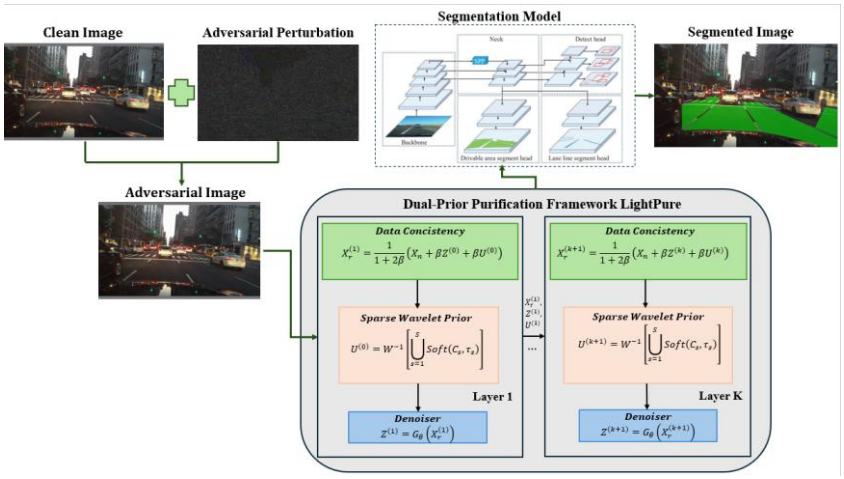


Figure 3.1: Architecture of the LightPure purification framework coupled with the YOLOP[18] segmentation model. LightPure addresses adversarial corruption by unrolling a structured optimization process that integrates data fidelity, wavelet-domain sparsity, and a lightweight learned prior. The purified output is then fed into YOLOP, enabling robust drivable-area and lane-line segmentation under adversarial conditions.

3.1.3. Experimental Evaluation

This section evaluates the effectiveness and practical relevance of LightPure in improving adversarial robustness for RGB-based semantic segmentation. The evaluation focuses on three key aspects: (i) the impact of adversarial perturbations on camera-based segmentation, (ii) the robustness gains provided by LightPure on a large-scale benchmark, and (iii) the feasibility of deployment under real-time and embedded constraints. To evaluate the effectiveness of the proposed LightPure framework, we compare against representative diffusion-based adversarial purification methods, namely DiffPure[19] and OSCP[20], which constitute state-of-the-art defenses for RGB-based perception. DiffPure leverages iterative denoising diffusion models to purify adversarial inputs prior to inference, while OSCP adopts a one-step guided diffusion formulation to reduce inference latency compared to traditional diffusion pipelines.

Datasets and Experimental Setup

Experiments are conducted on the BDD100K[21] dataset, a large-scale benchmark for autonomous driving perception containing diverse real-world driving scenarios. We focus on drivable-area and lane-line semantic segmentation, which are critical for downstream planning and localization tasks. The YOLOP[22] model is employed as the

[18] Wu, D., Liao, M., Zhang, W., Wang, X., Bai, X., Cheng, W. and Liu, W. (2021) 'YOLOP: You only look once for panoptic driving perception', *arXiv preprint*, arXiv:2108.11250.
[19] Nie, W., Guo, B., Huang, Y., Xiao, C., Vahdat, A. and Anandkumar, A. (2022) 'Diffusion models for adversarial purification', *arXiv preprint*, arXiv:2205.07460.
[20] Lei, C.T., Yam, H.M., Guo, Z., Qian, Y. and Lau, C.P. (2025) 'Instant adversarial purification with adversarial consistency distillation', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24331–24340.
[21] Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V. and Darrell, T. (2020) 'BDD100K: A diverse driving dataset for heterogeneous multitask learning', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2636–2645.
[22] Wu, D., Liao, M., Zhang, W., Wang, X., Bai, X., Cheng, W. and Liu, W. (2021) 'YOLOP: You only look once for panoptic driving perception', *arXiv preprint*, arXiv:2108.11250.

segmentation backbone. Its segmentation heads are kept frozen throughout all experiments and serve as a fixed target for adversarial attacks and defense evaluation.

Effectiveness of LightPure on Open Datasets

Table 1 and Table 2 summarize segmentation performance under ℓ∞ and ℓ2 PGD attacks with increasing perturbation strengths. Across all attack settings, LightPure consistently recovers a substantial portion of the performance lost due to adversarial perturbations. Under strong l_∞ attacks, LightPure restores up to +167% mIoU improvement for drivable-area segmentation and up to +27% mIoU improvement for lane-line segmentation relative to the attacked baseline. Similar robustness trends are observed under l_2 attacks, where LightPure maintains stable recovery as the perturbation budget increases.

Table 1. Robustness evaluation under l_∞ adversarial attacks on driving area and lane line segmentation.

	Driving Area Segmentation			Lane Line Segmentation			#Params (M, defense+segm.)	FPS
Method	Acc	IoU	mIoU	Acc	IoU	mIoU		
ℓ_∞ attack, noise level $\boldsymbol{\epsilon_1 = 4/255}$								
No attack	0.974	0.861	0.915	0.705	0.262	0.623	$-+7.94M$	–
Attack	0.637	0.159	0.385	0.43	0.056	0.497	$-+7.94M$	–
Attack + DiffPure [24]	0.96	0.797	0.874	0.527	0.212	0.589	$552.81M + 7.94M$	0.13
ttack + OSCP [18]	0.957	0.75	0.849	0.566	0.203	0.593	$635M + 7.94M$	4.00
Attack + proposed LightPure	**0.966**	**0.818**	**0.888**	**0.69**	**0.232**	**0.607**	$0.04M + 7.94M$ (**99.99%**↓)	**17.03**
ℓ_∞ attack, noise level $\boldsymbol{\epsilon_2 = 8/255}$								
No attack	0.974	0.861	0.915	0.705	0.262	0.623	$-+7.94M$	–
Attack	0.553	0.092	0.331	0.336	0.038	0.483	$-+7.94M$	–
Attack + DiffPure [24]	0.958	0.789	0.87	0.521	0.212	0.598	$552.81M + 7.94M$	0.13
Attack + OSCP [18]	0.949	0.711	0.825	0.493	0.192	0.588	$635M + 7.94M$	4.00
Attack + proposed LightPure	**0.964**	**0.811**	**0.884**	**0.661**	**0.242**	**0.612**	$0.04M + 7.94M$(**99.99%**↓)	**17.03**

Table 2. Robustness evaluation under l_2 adversarial attacks on driving area and lane line segmentation.

	Driving Area Segmentation			Lane Line Segmentation			#Params (M, defense+segm.)	FPS
Method	Acc	IoU	mIoU	Acc	IoU	mIoU		
ℓ_2 attack, noise level $\boldsymbol{\epsilon_1 = 15}$								
No attack	0.974	0.861	0.915	0.705	0.262	0.623	$-+7.94M$	–
Attack	0.905	0.558	0.725	0.496	0.191	0.588	$-+7.94M$	–
Attack + DiffPure [24]	0.960	0.800	0.876	0.525	0.213	0.598	$552.81M + 7.94M$	0.13
Attack + OSCP [18]	0.966	0.780	0.869	0.586	**0.215**	**0.599**	$635M + 7.94M$	4.00
Attack + proposed LightPure	**0.967**	**0.823**	**0.892**	**0.637**	0.201	0.590	$0.04M + 7.94M$ (**99.99%**↓)	**17.03**
ℓ_2 attack, noise level $\boldsymbol{\epsilon_2 = 25}$								
No attack	0.974	0.861	0.915	0.705	0.262	0.623	$-+7.94M$	
Attack	0.886	0.472	0.673	0.372	0.167	0.577	$-+7.94M$	
Attack + DiffPure [24]	0.959	0.793	0.872	0.521	0.213	0.598	$552.81M + 7.94M$	0.13
Attack + OSCP [18]	0.961	0.760	0.856	0.527	0.210	0.598	$635M + 7.94M$	4.00
Attack + proposed LightPure	**0.965**	**0.808**	**0.883**	**0.661**	**0.219**	**0.600**	$0.04M + 7.94M$ (**99.99%**↓)	**17.03**

Diffusion-based purification methods also recover segmentation accuracy; however, their robustness gains come at the cost of extreme computational and memory overhead, as discussed next.

Efficiency and Deployment Considerations

From a deployment perspective, computational efficiency is critical. While diffusion-based purification methods achieve competitive robustness, they introduce hundreds of millions of parameters, often exceeding the size of the segmentation model itself. In contrast, LightPure introduces less than 0.5% additional parameters relative to the YOLOP backbone and achieves real-time inference performance. As shown in Table

1 and Table 2, LightPure operates at 17.03 FPS, outperforming diffusion-based defenses by several orders of magnitude in throughput.

3.1.4. Next Steps

Future work will focus on further strengthening the evaluation and efficiency of the proposed LightPure framework. Specifically, we plan to extend the comparative analysis to a broader set of adversarial defense techniques, beyond the methods currently considered, to provide a more comprehensive benchmarking. In addition, the evaluation will be expanded to multiple autonomous driving datasets to assess robustness and generalization across diverse environments and sensing conditions. The threat model will also be enriched by considering a wider range of adversarial attacks, including stronger white-box formulations and additional black-box attack scenarios. Finally, we will further optimize the proposed purification framework to improve computational efficiency, exploring reductions in model complexity and inference cost while maintaining adversarial robustness for real-time deployment.

3.2. VisionGuard: Diffusion-based Adversarial Purification

Adversarial purification[23,24] represents a powerful defence paradigm for securing vision-based perception systems. In contrast to Adversarial Training (AT)[25] , which requires costly retraining of classifiers and often struggles to generalize to unseen threats, purification methods operate as a pre-processing step, removing perturbations from inputs before they reach the downstream model. However, state-of-the-art purification based on Denoising Diffusion Probabilistic Models (DDPMs)[26] incurs substantial computational overhead due to iterative sampling, rendering it impractical for resource-constrained edge applications such as aerial drones. In this section, we investigate "VisionGuard," an efficient adversarial purification framework designed to enable robust diffusion-based defence on edge devices.

3.2.1. Methodology

To address the limitations of standard diffusion-based defences, VisionGuard alters the sampling paradigm by replacing the computationally expensive Markovian process of DDPMs with Denoising Diffusion Implicit Models (DDIMs). Unlike DDPMs, which require a rigid sequence of sampling steps (typically T=1000), DDIMs utilize a non-Markovian generative process. This formulation allows the purification process to define a sparse subset of timesteps, enabling the system to "skip" steps during the reverse diffusion process. Consequently, VisionGuard generates high-fidelity purified images with significantly fewer iterations (reducing the sampling steps from thousands to as few as 30) without retraining the noise prediction network.

[23] Changhao Shi, Chester Holtz, and Gal Mishne. Online adversarial purification based on self-supervision. arXiv preprint arXiv:2101.09387, 2021.
[24] Jongmin Yoon, Sung Ju Hwang, and Juho Lee. Adversarial purification with score based generative models. In International Conference on Machine Learning, pages 12062–12072. PMLR, 2021.
[25] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.
[26] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. arXiv preprint arXiv:2205.07460, 2022.

To further optimize edge deployment, the framework incorporates a specific knowledge of distillation and finetuning strategy. A compact "student" U-Net is trained to mimic the noise prediction output of a larger, pre-trained "teacher" model. By minimizing the Mean Squared Error (MSE) between the teacher and student noise predictions, robust generative capabilities are transferred to a lightweight architecture. This distilled model is subsequently finetuned on clean data to mitigate quality degradation.

3.2.2. Experimental Results

We evaluated VisionGuard on standard benchmarks (CIFAR-10) and high-resolution datasets (CelebA-HQ) using NVIDIA Jetson Xavier NX edge devices and NVIDIA A100 GPUs. Additionally, we have evaluated VisionGuard on real aerial images using the AID dataset.

Adversarial Robustness on CIFAR-10

We compared the robustness of VisionGuard against state-of-the-art defenses, including projected gradient descent (PGD) Adversarial Training (AT)[27] and standard DDPM purification. As presented in Table 3, the proposed approach demonstrates superior robustness compared to baseline methods.

Table 3. Standard accuracy and robust accuracy against various adversarial attacks on CIFAR-10, with a WideResNet-28-10 classifier.

Method	Standard Acc (%)	FGSM (%)	PGD (%)	CW (%)	APGD-CE (%)	APGD-T (%)	FAB-T (%)	SQUARE (%)
Regular Training	93.84	12.49	0.00	0.00	0.00	0.00	0.01	0.16
PGD AT	86.00	60.1	54.18	7.19	51.57	49.58	50.15	58.94
DDPM	87.16	78.6	83.14	85.76	84.52	84.59	86.41	84.39
VisionGuard (Ours)	89.83	68.5	82.38	88.16	84.88	85.56	89.12	86.79
Distilled VisionGuard (Ours)	80.63	51.2	63.64	75.67	69.06	70.48	78.38	73.18

While the distilled variant trades some accuracy for speed, it still significantly outperforms the undefended model and surpasses PGD AT against the formidable CW (L_2) attack. Notably, the standard VisionGuard achieves the highest robust accuracy against sophisticated gradient-based attacks, including APGD-CE (84.88%) and FAB-T (89.12%), as well as the L_2-based CW attack (88.16%), validating the efficacy of the implicit sampling trajectory.

Computational Efficiency on CIFAR-10

The primary contribution of VisionGuard is enabling these defenses on resource-constrained hardware. Table 4 details the inference speed and model complexity on an edge device (Jetson Xavier NX).

[27] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083, 2017.

Table 4. Inference speed on NVIDIA Jetson Xavier NX and A100 GPU for CIFAR-10.

Purification Method	Jetson Xavier NX		A100		GFLOPS	Parameter Number (M)
	Time per Frame (s)	FPS	Time per Frame (s)	FPS		
DDPM	4.6981	0.21	1.2568	0.8	6.24	35.75
VisionGuard (Ours)	0.216	5.14	0.0528	18.95	6.24	35.75
Distilled VisionGuard (Ours)	0.1371	8.11	0.0316	31.7	1.08	4.44

The baseline DDPM operates at a prohibitive 0.21 FPS on the Jetson device. In contrast, VisionGuard achieves a ~25x speedup. The Distilled finetuned variant further accelerates performance, achieving 8.11 FPS (a ~40x improvement over DDPM) while reducing the computational load (GFLOPS) by 82.7% and parameter count by 87.6%. This transforms adversarial purification from a theoretical concept into a practical runtime solution for edge robotics.

Adversarial Robustness on CelebA-HQ

We evaluated resilience against the adaptive BPDA+EOT attack, which attempts to approximate gradients through the purification process. As shown in Table 5, VisionGuard demonstrates exceptional stability.

Table 5. Standard accuracy and robust accuracy against BPDA+EOT on CelebA-HQ, with a ResNet-18[28]. We evaluate on the smiling attribute.

Method	Standard Accuracy	BPDA+EOT
NVAE[29]	93.55%	0.00%
GAN+OPT[30]	93.49%	3.41%
GAN+ENC+OPT[31]	93.68%	0.78%
GAN+ENC[32]	90.55%	40.40%
DDPM	89.78%	55.73%
VisionGuard (Ours)	92.42%	78.40%
Distilled VisionGuard (Ours)	87.15%	74.31%

VisionGuard achieves a robust accuracy of 78.40%, marking a +22.67% improvement over the DDPM baseline. This performance gap is attributed to the deterministic

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 770–778.
[29] Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. Advances in neural information processing systems, 33:19667–19679, 2020.
[30] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 8110–8119, 2020.
[31] Lucy Chai, Jun-Yan Zhu, Eli Shechtman, Phillip Isola, and Richard Zhang. Ensembling with deep generative views. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 14997–15007, 2021.
[32] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a style gan encoder for image-to-image translation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2287–2296, 2021.

sampling inherent to our implementation. Unlike the stochastic paths of DDPMs, the deterministic reverse process effectively masks the gradient landscape, rendering the BPDA gradient approximation ineffective. Even the lightweight Distilled model (74.31%) substantially outperforms the computationally heavier DDPM and GAN-based baselines.

Computational Efficiency on CelebA-HQ

Processing high-resolution images on edge devices presents a significant challenge. Table 6 quantifies the efficiency gains of our proposed method compared to standard DDPMs.

Table 6. Inference speed on NVIDIA Jetson Xavier NX and A100 GPU for CelebA-HQ.

Purification Method	Jetson Xavier NX		A100		GFLOPS	Parameter Number (M)
	Time per Frame (s)	FPS	Time per Frame (s)	FPS		
DDPM	214.4115	0.005	11.3171	0.09	249.36	113.67 M
VisionGuard (Ours)	7.5725	0.13	0.3405	2.94	249.36	113.67 M
Distilled VisionGuard (Ours)	3.1464	0.32	0.1958	5.11	91.97	20.31 M

Table 6 highlights that standard DDPM is unusable for high-resolution edge processing (0.005 FPS). However, the Distilled VisionGuard model provides a significant acceleration, achieving a 68x speedup on the Jetson edge device compared to DDPM, while maintaining the strong robust accuracy observed in Table 3. This reduces the GFLOPS by approximately 63%, and the parameter count by over 82%, making diffusion-based defense viable for larger images on edge hardware.

Adversarial Robustness on Aerial Image Dataset (AID)

To validate the applicability of VisionGuard in its primary target domain, we conducted evaluations using the AID dataset. We have evaluated robustness against the FGSM, PGD, and CW attacks. The comparisons can be seen in Table 7.

Table 7. Standard accuracy and robust accuracy against various adversarial attacks on the AID dataset.

	Standard Acc (%)	FGSM (%)	PGD (%)	CW (%)
No Defence	87.14%	5.91%	0.00%	0.00%
DDPM	71.15%	64.43%	65.91%	62.74%
VisionGuard	70.86%	60.18%	63.01%	64.00%

As illustrated in Table 7, the undefended model exhibits failure under iterative attacks, with accuracy dropping to 0.00% against PGD and CW. VisionGuard successfully mitigates these threats, restoring robust accuracy to 63.01% under PGD and 64.00% under CW. Crucially, VisionGuard demonstrates comparable performance with the computationally intensive DDPM baseline. Given the efficiency established in the previous sections, this result confirms that VisionGuard delivers the protective capabilities of standard diffusion models but at significantly faster speed. A visualization of the purification on the AID dataset is presented in Figure 3.2.

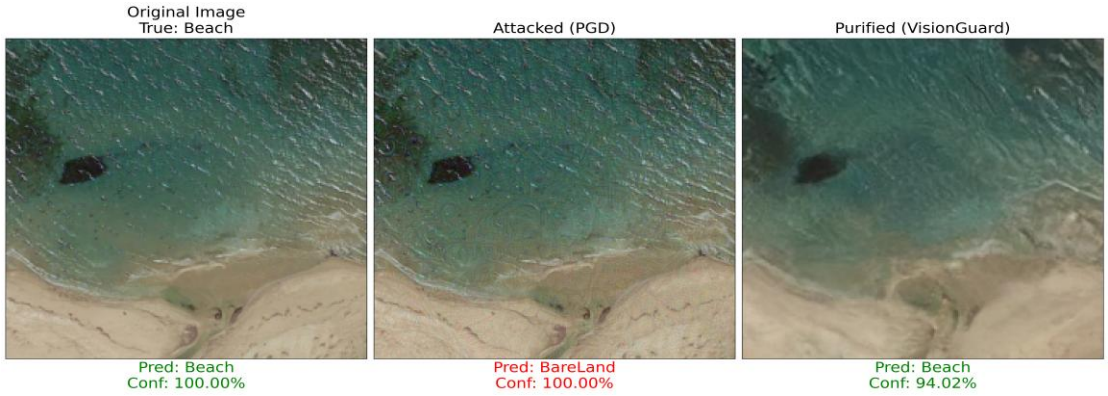


Figure 3.2: An example of adversarial purification using VisionGuard.

3.2.3. Next Steps

Our future research aims to further bridge the gap between robust diffusion-based defense and real-time edge constraints. We plan to investigate one-step adversarial purification methods, utilizing models such as Latent Consistency Models (LCMs)[33] to achieve single-step inference. Furthermore, we intend to integrate Low-Rank Adaptation (LoRA) [34] strategies to efficiently adapt pre-trained diffusion models to the aerial domain with minimal memory overhead. To enhance the fidelity of purified images, we will explore structural guidance mechanisms, such as edge detection, to preserve semantic details during the purification process. Finally, we will extend our evaluation to a broader range of aerial imaging datasets to ensure generalizability across different terrains and capture conditions

3.3. Adversarial Patch Threats in Aerial Imagery

Adversarial patch attacks represent a particularly severe threat for vision-based perception systems. In contrast to pixel-level adversarial noise, which is often imperceptible but fragile, adversarial patches are localized, structured perturbations that can physically realize and remain effective under changes in scale, viewpoint and illumination. This makes them especially concerning for aerial imagery applications, where cameras are mounted on UAVs or drones operating at varying altitudes, angles, and environmental conditions.

Previous studies have demonstrated that adversarial patches can reliably disrupt deep neural networks across a wide range of scenes and transformations, even when the patch content is simple or visually conspicuous[35]. In particular, it has been shown that a single patch can generalize across locations and viewing conditions, while other studies extended these findings to object detection architectures, highlighting their vulnerability to localized perturbations that suppress or mislead detections rather than

[33] S. Luo et al., “Latent Consistency Models: Synthesizing High-Resolution Images with Few-Step Inference,” 2023, arXiv. doi: 10.48550/ARXIV.2310.04378.
[34] E. J. Hu et al., “LoRA: Low-Rank Adaptation of Large Language Models,” 2021, arXiv. doi: 10.48550/ARXIV.2106.09685.
[35] T. Brown, D. Mane, A. Roy, M. Abadi, and J. Gilmer, “Adversarial Patch,” *arXiv:1712.09665*, 2017.

simply altering class labels[36,37,38]. These characteristics motivate a focused analysis of adversarial patch threats in aerial object detection pipelines.

In this section, we investigate the impact of adversarial patches on aerial object detection. The objective is twofold: first, to quantify the vulnerability of standard aerial object detection models to different patch configurations; and second, to establish strong baseline attack performance against which future defense methods can be developed.

3.3.1. Patch Types and Attack Setup

We evaluate adversarial patch attacks on a YOLOv5-based aerial object detection pipeline using the VisDrone dataset[39]. The detector is kept frozen throughout the experiments so that observed changes in performance can be attributed solely to patch application rather than model adaptation.

Patches are applied at object-level locations using ground-truth bounding boxes, simulating targeted physical placement on objects of interest. Patch size is defined in pixel dimensions and/or scaled relative to object size (depending on configuration), while patch application supports transformations, such as scaling, optional rotation, and spatial perturbations to better reflect aerial viewing conditions.

To capture a range of patch characteristics, we consider four representative patch types:

- *Random patches*, initialized with uniformly random pixel values.
- *Gray patches*, consisting of constant mid-level intensity values.
- *White patches*, corresponding to constant high-intensity values.
- *Checkerboard patches*, featuring alternating high-frequency patterns.

These patch types were selected to assess whether attack effectiveness depends on complicated texture optimization or can arise even from visually simple or structured patterns, as suggested prior to adversarial patch studies on object detectors.

3.3.2. Quantitative Impact on Detection Performance

Under clean (non-patched) conditions, the detector achieves near-perfect performance on the evaluated aerial scenes with Average Precision (AP) and Average Recall (AR) close to 1.0. This confirms that the baseline model performs strongly in nominal settings and provides a stable reference for evaluating adversarial impact.

When patches are introduced, detection performance degrades substantially across all patch types. *Random patches* of size 64x64 reduce AP to 0.178 and AR to 0.278 yielding an Attack Success Rate (ASR) of 0.472. *Gray patches* show even stronger

[36] S.-T. Chen, C. Cornelius, J. Martin, and D. H. Chau, “Robust Physical Adversarial Attacks on Faster R-CNN Object Detector,” *arXiv:1804.05810*, 2018.
[37] Y. Liu et al., “DPATCH: An Adversarial Patch Attack on Object Detectors,” *arXiv:1806.02299*, 2018.
[38] A. Thys, W. Van Ranst, and T. Goedemé, “Fooling Automated Surveillance Cameras: Adversarial Patches to Attack Person Detection,” *CVPR Workshops*, 2019.
[39] P. Zhu et al., "Detection and Tracking Meet Drones Challenge," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 11, pp. 7380-7399, 1 Nov. 2022, doi: 10.1109/TPAMI.2021.3119563.

disruption, increasing ASR to 0.577 despite their visually simple appearance. White and checkerboard patches, evaluated at a 32x32 resolution, also cause significant performance drops, with ASR values of 0.303 and 0.434 respectively.

These findings reinforce the practical risk of patch-based attacks in aerial imagery: strong attack success can be achieved even with visually simple patterns, consistent with observations that localized perturbations can interfere with detection confidence and post-processing, rather than merely changing predicted classes. A summary of quantitative results is provided in Table 7.

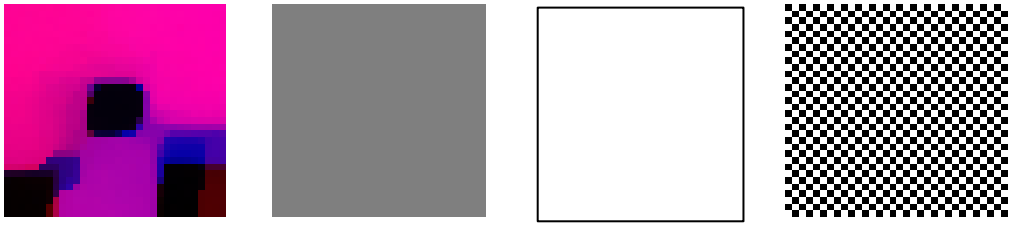
Table 7. Object detection performance under adversarial patch attacks on aerial imagery. Results are reported for clean images and for different patch types in terms of Average Precision (AP), Average Recall (AR), and Attack Success Rate (ASR)

Patch type	AP	AR	ASR
Clean	0.993	0.996	-
Random (64x64, random placement, rotate patch)	0.178	0.278	0.472
Gray (64x64, random placement, rotate patch)	0.165	0.254	0.577
White (32x32, No random placement, No patch rotation)	0.356	0.414	0.303
Checkerboard (32x32, No random placement, No patch rotation)	0.343	0.397	0.434

3.3.3. Qualitative Effects and Visual Analysis

Qualitative inspection of patched scenes reveals consistent failure modes. In dense aerial scenes, patches can lead to complete suppression of detections for small and medium objects, particularly when patch placement overlaps salient object regions. In other cases, detections persist but with substantially reduced confidence, leading to their filtering during non-maximum suppression.

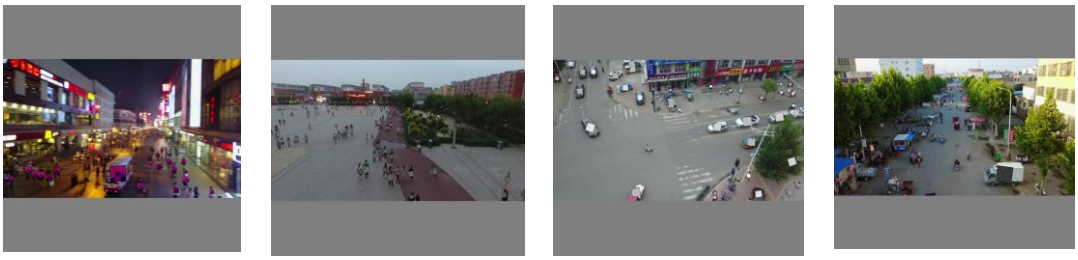
These effects are particularly pronounced in aerial imagery because objects often occupy relatively few pixels and exhibit large-scale variability, making detection pipelines more sensitive to localized feature disruptions. Representative examples for the different patch types are illustrated in Figure 3.2.



(a) Random (b) Gray (c) White (d) Checkerboard

Figure 3.2: Examples of different patch types.

Figure 3.3 presents representative examples of adversarial patch application on aerial images for the different patch configurations considered. The examples illustrate how localized visual perturbations affect detection outcomes in practice.



(a) Random (b) Gray (c) White (d) Checkerboard

Figure 3.3: Representative examples of adversarial patch application in aerial imagery for different patch types.

The results demonstrate that adversarial patches can significantly compromise aerial object detection performance, achieving high ASR without requiring any modification to the detector architecture at deployment time. Importantly, the effectiveness of gray and structured checkerboard patches suggests that successful attacks do not necessarily require sophisticated texture optimization, which lowers the barrier for potential adversaries. These results serve as a reproducible baseline for adversarial patch threats in aerial imagery, and they motivate the development of complementary resilience mechanisms, including patch-aware training techniques, patch detection approaches, and context-aware consistency checks across frames or modalities, in line with broader robustness objectives.

3.3.4. Next Steps

Future work will extend the current analysis by investigating gradient-based optimization of adversarial patches and evaluating their transferability across different object detection architectures and aerial datasets. In parallel, robustness-oriented approaches will be explored, including patch-aware training strategies and lightweight inference-time mechanisms for detecting or mitigating patch-induced perturbations. Additional aspects such as object scale, scene density, and aerial-specific constraints will also be considered to better characterize patch effectiveness under realistic operating conditions.

3.4. Saliency-Guided Recovery from Adversarial Perturbations

Adversarial patch attacks are known to cause object detectors to make misleading predictions, resulting in partially or completely missed objects in the scene. Presently, recovery methods often require extensive training to ensure the attacked patches are effectively removed without loss of performance in unattacked scenarios. This leads to defense techniques having ineffective generalization to out-of-distribution attacks or data domains. Furthermore, such techniques require a significant pretraining budget to be effective when deployed in downstream applications. Therefore, there is significant potential in developing adversarial defense techniques that are training-free and can be applied to a range of datasets and attack types.

3.4.1. The SaliSAM pipeline

Towards this goal, we present SaliSAM - a fully automatic training-free recovery method for detecting adversarial patches in images that can defend against patches of different sizes, shapes and multiple patches. SaliSAM does not require any training for the attack recovery module and hence can generalize effectively across multiple datasets and diverse scenarios. SaliSAM uses a saliency-based binary mask estimation step for coarse localization of the adversarial patch followed by a Segment Anything model to extract fine-grained masks from the input image. Finally, the SAM masks with a high saliency value are used to accurately detect and inpaint the adversarial patch.

SaliSAM is a saliency-guided approach for training-free, automatic recovery of adversarial patch attacks in CNNs. As illustrated in Figure 3.4, the SaliSAM pipeline consists of two parallel streams: the Saliency Branch and the SAM Branch. In the Saliency Branch, the input attacked image is passed through the victim model to extract feature maps, which are then used to extract a binary saliency mask using an ensemble of saliency thresholds. The SAM branch, on the other hand, is used to extract dense segmentation masks of the input attacked image. Finally, the saliency mask is used to filter overlapping SAM masks for fine-grained localization of the adversarial patch. Finally, the localized patch mask is used to inpaint the recovery of the patch-removed image.

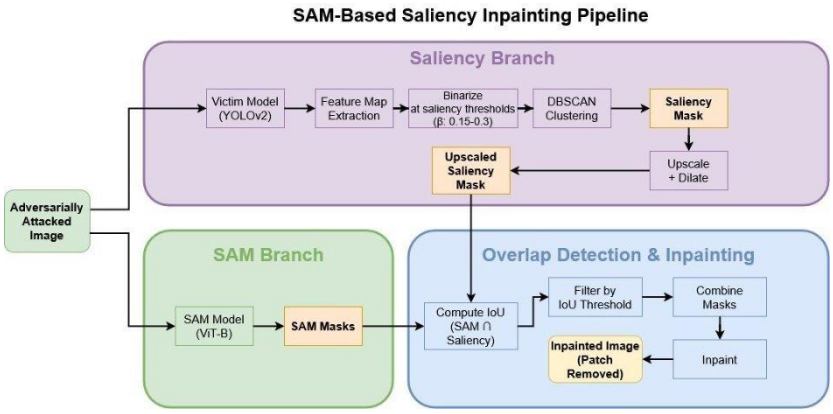


Figure 3.4: The SaliSAM pipeline

3.4.2. Experimental setup

In this section, experimental setup is described for the evaluation of the SaliSAM pipeline, focusing on the utilized datasets and the adopted performance metrics.

Datasets Used
To validate the performance of SaliSAM, we experiment on two challenging object detection datasets: the INRIA Detection Dataset[40] and the Pascal VOC Detection dataset[41].

INRIA Person Detection Dataset: The INRIA Detection dataset is an object detection dataset with images and corresponding bounding box annotations for detecting objects in a scene. We use the official test split containing 288 images and focus on the 'person' class for the patch attacks. We implement the single patch, double patch, multi-object and triangular patch attacks for this set of images resulting in the INRIA-1, INRIA-2, INRIA-MO and INRIA-T attacked variants of the dataset, each with 288 images.

Pascal VOC Detection Dataset: The Pascal VOC Detection Dataset is a popular object detection dataset. We use the official test split of the 2007 version of the dataset in our experiments. We select a subset of 500 images from this test split to prepare the attacked datasets, resulting in the VOC-1, VOC-2, VOC-MO and VOC-T attacked datasets.

Evaluation Metrics
We adopt the Recovery Rate (RR) and Lost Predictions Rate (LPR) metrics to evaluate the performance of SaliSAM.

Recovery Rate (RR) is the ratio between the number of recovered inputs and the total number of effectively attacked inputs. For a dataset of attacked images X_A, the recovery rate of a defense technique R is calculated as:

$$RR = \frac{N_{rec}}{N_{atk}} \quad (6)$$

where $N_{rec}$is the number of images successfully recovered by the defense technique and $N_{atk}$is the total number of effectively attacked images in the dataset X_A.

Similarly, for a defense technique R, the **Lost Prediction Rate (LPR)** is the ratio of the number of clean images for which applying R leads to incorrect predictions over the total number of clean images. For a clean dataset X_C and an image x_i from this dataset, the Lost Prediction Rate is calculated as:

$$LPR = \frac{N_{lost}}{N_{clean}} \quad (7)$$

[40] https://universe.roboflow.com/pascal-to-yolo-8yygq/inria-person-detection-dataset
[41] M. Everingham et al., "The pascal visual object classes (VOC) challenge", International Journal of computer vision 88.2 (2010): 303-338.

where N_{lost} is the number of images for which $R(x_i)$ leads to an incorrect prediction and N_{clean} is the total number of clean images in the dataset X_C.

3.4.3. Results and Discussion

In our experiments, it was seen that SaliSAM was able to consistently outperform Saluitl – the present state-of-the-art patch recovery technique – across different attack types on both the Pascal VOC and INRIA datasets. In Table 8, we detail the RR vs. LPR tradeoff performance of SaliSAM compared to Saluitl across different types of patch attacks. SaliSAM is able to achieve comparable or higher performance than Saliuitl while being an entirely training free method.

Also, in Figure 3.5, we demonstrate some qualitative examples of the bounding boxes predicted by the YOLOv2 victim model on attacked, Saliuitl-recovered and SaliSAM-recovered images. It can be seen that images recovered by SaliSAM have significantly less original information loss compared to those recovered by Saliuitl. Also, the YOLOv2 model can make cleaner bounding box predictions and fewer false positives on images recovered by SaliSAM compared to those recovered by Saluitl.

Table 8. Quantitative evaluations for the Recovery Rate (RR) vs. Lost Prediction Rate (LPR) trade-offs in single (1), double (2), triangular (T) and multi-object (MO) patch attacks. Best and second-best scores are highlighted as bold and underlined respectively, based on the difference, RR − LPR.

Attack	INRIA-1	INRIA-2	INRIA-T	INRIA-MO	VOC-1	VOC-2	VOC-T	VOC-MO
RR/LPR								
Saliuitl	0.7812/0.0	**0.8368/0.0**	0.8509/0.0	0.7639.0.0	0.4615/0.0121	0.3866/0.0121	0.7085/0.0121	0.2854/0.0121
SaliSAM (ours)	**0.8576/0.0104**	0.8160/0.0104	**0.8888/0.0104**	**0.7847/0.0104**	**0.5627/0.0243**	**0.4674/0.0243**	**0.7490/0.0243**	**0.3684/0.0243**

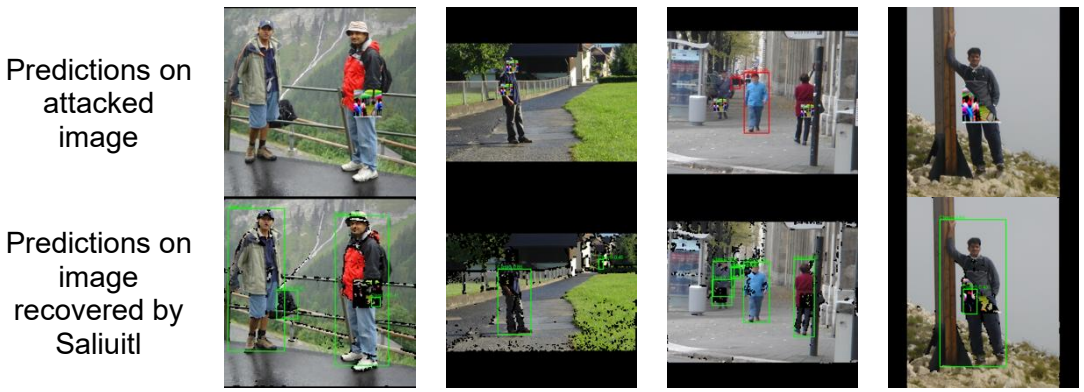


Figure 3.5: Qualitative comparisons of the recovery performance of SaliSAM. Across all attacks, images recovered by SaliSAM can yield better predictions from the victim model (YOLOv2). Also, images recovered by SaliSAM are characterized by a lesser loss of original information than those from Saliuitl.

3.4.4. Next Steps

In future work, the SaliSAM pipeline will be extended with the following:

1. The present SaliSAM pipeline is a passive method that applies the patch detection and segmentation steps to all input samples. Future work will focus on implementing an attack detection component that can identify if an input image consists of adversarial patch before performing the saliency mask extraction and patch segmentation steps. This should avoid unnecessary computational overhead and potential performance loss in unattacked scenarios.
2. The segmentation branch presently employs a Segment Anything model for fine-grained localization of adversarial patches in the entire input image. This could be further improved by allowing the SAM module to only focus on salient regions in the image, further improving processing times for each input sample.
3. Additional benchmarking with more efficient segmentation backbones for the segmentation branch could allow for choosing the most effective SaliSAM pipeline to be used under different hardware configurations.

3.5. Adversarial Resiliency for LiDAR: Benefits on semantic Segmentation

LiDAR-based perception is a core component of modern autonomous systems, providing precise geometric information that supports environment understanding, obstacle detection, and semantic segmentation. In particular, LiDAR semantic segmentation plays a critical role in safety-relevant decision-making by assigning semantic labels to the surrounding scene, enabling higher-level reasoning such as free-space estimation, object interaction, and motion planning. Despite its importance, recent studies have shown that LiDAR-based perception pipelines are vulnerable to adversarial perturbations, where carefully crafted but subtle input manipulations can significantly degrade segmentation performance without being easily detectable.

Adversarial attacks on LiDAR data exploit the sensitivity of deep learning models to small, structured perturbations. These attacks can lead to incorrect semantic predictions, such as misclassifying road surfaces, vehicles, or pedestrians, thereby compromising the reliability of autonomous systems. Unlike random sensor noise, adversarial perturbations are intentionally optimized to maximize model error, making them particularly dangerous in safety-critical or adversarial environments

3.5.1. Lightweight Deep Unrolling Based Adversarial Purification for LiDAR Segmentation (DU-AP)

Existing adversarial attacks on LiDAR perception systems can be broadly categorized into point perturbation, point injection, and point removal attacks. Point perturbation attacks introduce minimal spatial displacements to existing points, while injection attacks add synthetic points to confuse perception, and removal attacks selectively eliminate salient points critical for correct prediction.[42] Although these attack types have been extensively studied for 3D object detection, their impact on semantic segmentation has gained increasing attention due to the central role segmentation plays in downstream autonomy tasks[43].

To counter such threats, several defence strategies have been explored in the literature. Adversarial training improves robustness by retraining models on adversarial examples, but it requires access to labelled data, incurs substantial computational cost, and often generalizes poorly to unseen attack strategies.[44,45,46] More recently, input purification approaches have emerged, where adversarial noise is removed from the sensor input prior to inference. Many of these methods rely on generative models, such as Generative Adversarial Networks (GANs)[47,48,49,50] or diffusion models[51,52], which iteratively reconstruct clean inputs from perturbed data. While generative purification methods demonstrate strong denoising capabilities, they are typically computationally expensive and introduce significant latency. In many cases, the defense model is substantially larger than the segmentation network it protects, making real-time deployment on embedded automotive platforms impractical.

---

[42] Zhang, Y., Hou, J. and Yuan, Y. (2024). *A comprehensive study of the robustness for LiDAR-based 3D object detectors against adversarial attacks*. International Journal of Computer Vision, 132(5), pp. 1592–1624.

[43] Zhu, Y., Miao, C., Hajiaghajani, F., Huai, M., Su, L. and Qiao, C. (2021). *Adversarial attacks against LiDAR semantic segmentation in autonomous driving*. In: Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems, pp. 329–342

[44] Lehner, A., Gasperini, S., Marcos-Ramiro, A., Schmidt, M., Mahani, M.-A.N., Navab, N., Busam, B. and Tombari, F. (2022). *3D-VField: Adversarial augmentation of point clouds for domain generalization in 3D object detection*. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 17295–17304.

[45] Gu, J., Zhao, H., Tresp, V. and Torr, P.H.S. (2022). *SegPGD: An effective and efficient adversarial attack for evaluating and boosting segmentation robustness*. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer, pp. 308–325.

[46] Zhou, Q., Lei, M., Zhi, P., Zhao, R., Shen, J. and Yong, B. (2022). *Towards improving the anti-attack capability of the RangeNet++*. In: Proceedings of the Asian Conference on Computer Vision (ACCV), pp. 56–67.

[47] Jin, G., Shen, S., Zhang, D., Dai, F. and Zhang, Y. (2019). *APE-GAN: Adversarial perturbation elimination with GAN*. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 3842–3846.

[48] Cheng, G., Sun, X., Li, K., Guo, L. and Han, J. (2021). *Perturbation-seeking generative adversarial networks: A defense framework for remote sensing image scene classification*. IEEE Transactions on Geoscience and Remote Sensing, 60, pp. 1–11.

[49] Lin, G., Li, C., Zhang, J., Tanaka, T. and Zhao, Q. (2024). *Adversarial training on purification (ATOP): Advancing both robustness and generalization*. arXiv preprint, arXiv:2401.16352.

[50] Wang, Y., Liao, X., Cui, W. and Yang, Y. (2025). *Defending against and generating adversarial examples together with generative adversarial networks*. Scientific Reports, 15(1), p. 12994.

[51] Cai, M., Wang, X., Sohel, F. and Lei, H. (2025). *LiDAR-SPD: Improving adversarial robustness of 3D object detection via spherical projection and diffusion*. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 1–5.

[52] Cai, M., Wang, X., Sohel, F. and Lei, H. (2024). *Diffusion models-based purification for common corruptions on robust 3D object detection*. Sensors, 24(16), p. 5440.

A key observation in recent LiDAR perception research is that many state-of-the-art semantic segmentation pipelines do not operate directly on raw 3D point clouds. Instead, they project the point cloud into a structured 2D range-view representation, where each pixel corresponds to a LiDAR beam and azimuth angle. This representation enables efficient convolutional processing and is widely adopted due to its favorable trade-off between accuracy and computational cost.

However, most adversarial attacks and defenses have been designed for the 3D point cloud domain. When adversarial perturbations crafted in 3D are projected into the 2D range view, quantization effects and geometric distortions often reduce their effectiveness. This mismatch between attack formulation and model input representation limits both the realism of the threat model and the effectiveness of existing defenses for range-view segmentation networks.

This gap highlights the need for adversarial resiliency mechanisms that are explicitly tailored to the data representations used in practical LiDAR segmentation systems, rather than relying on defenses designed for fundamentally different processing pipelines.

3.5.1.1. Model-based purification framework: The DU-AP approach

Modern LiDAR semantic segmentation pipelines frequently operate on 2D range-view representations, where the 3D point cloud is projected onto a structured image defined over sensor channels and azimuth angles. While this representation enables efficient convolutional processing, it also defines the effective attack surface of the segmentation model. Adversarial perturbations designed in the raw 3D point cloud domain may lose effectiveness after projection due to quantization and geometric distortions and therefore do not accurately capture the threat model faced by range-view segmentation networks.

To reflect realistic attack conditions, adversarial perturbations are considered directly in the range-view domain. Let $Y_c \in R^{C\times M}$ denote a clean LiDAR range image, where C is the number of LiDAR channels and M the horizontal angular resolution. An adversarial example Y_n is generated by adding a bounded perturbation Δ_r to the clean input: $Y_n = Y_c + \Delta_r$. The objective of the adversarial attack is to degrade the segmentation performance of a pretrained model $f(\cdot)$, while ensuring that the perturbation remains small and physically plausible. This can be expressed through the following constrained optimization problem:

$$\min_{\Delta_r} -L_{seg}\left(f(Y_c + \Delta_r), Z\right) + \lambda\|\Delta_r\|_{F,}^2 \text{ s.t } |\Delta_r(i,j)| \le \varepsilon \qquad (8)$$

where $L_{seg}(\cdot)$ denotes the segmentation loss with respect to the ground-truth labels Z, ε controls the maximum allowable perturbation per pixel, and λ balances attack effectiveness against imperceptibility.

In practice, the above problem is solved using Projected Gradient Descent (PGD)[53], yielding adversarial range images that are highly effective against range-view

[53] Madry, A., Makelov, A., Schmidt, L., Tsipras, D. and Vladu, A. (2017). *Towards deep learning models resistant to adversarial attacks*. arXiv preprint, arXiv:1706.06083.

segmentation models. By operating directly on the 2D range-view representation, this attack formulation avoids projection-induced artifacts and provides a realistic and challenging input for evaluating adversarial defense mechanisms.

The resulting adversarially perturbed range image Y_n serves as the input to the proposed purification framework described next.

Deep Unrolled Adversarial Purification (DU-AP)
Given the adversarially perturbed range image Y_n obtained through the range-view attack formulation described above, the objective of DU-AP (Deep Unrolled Adversarial Purification) is to recover a purified range image Y_r such that the segmentation output is restored as closely as possible to that produced by the clean input Y_c.

Adversarial purification is framed as a model-based denoising problem that explicitly exploits both learned priors and structural properties of LiDAR range images. In contrast to purely data-driven generative approaches, DU-AP incorporates domain-specific regularization terms that reflect the physical acquisition characteristics of LiDAR sensors.

Purification objective
The recovery of the purified range image Y_r from the adversarial input Y_n is formulated as the following optimization problem:

$$arg\min_{Y_r} \frac{1}{2}\|Y_n - Y_r\|_F^2 + \lambda R(Y_r) + \mu\|\nabla_h Y_r\|_F^2 \quad (9)$$

The first term enforces data consistency, ensuring that the purified output remains close to the observed input. The second term $R(\cdot)$ denotes a learnable regularization that promotes statistical properties of clean range images. The third term introduces a horizontal smoothness prior, encouraging consistency between neighboring azimuth directions, which naturally correspond to adjacent LiDAR beams.

Iterative purification strategy
To efficiently solve the above optimization problem, DU-AP adopts a model-based iterative scheme inspired by classical variable-splitting methods. The purification process alternates between two complementary operations:

1. Data-consistency update, which refines the current estimate while enforcing smoothness along the horizontal (azimuth) dimension, and
2. Denoising step, which removes structured adversarial artifacts using a compact learnable module.

The data-consistency update is implemented through a regularized gradient descent step:

$$Y_r^{(k+1)} = Y_r^{(k)} - n\left(-\left(Y_n - Y_r^{(k)}\right) + bB^{(k)} + \mu\nabla_h^T\nabla_h Y_r^{(k)}\right) \quad (10)$$

Here, n denotes the step size, and the operator $\nabla_h^{\mathrm{T}}\nabla_h$ is implemented via a learnable 1D circular convolution, enabling the model to capture the periodic, ring-like structure inherent in LiDAR range images.

Following this update, a denoising operation is applied to suppress structured adversarial artifacts:

$$B^{(k+1)} = \mathcal{G}_\theta\left(Y_r^{(k+1)}\right) \tag{11}$$

where $\mathcal{G}_\theta$ is a lightweight convolutional neural network that acts as a learnable proximal operator, suppressing adversarial perturbations while preserving semantic content.

Deep unrolling and network realization
Rather than executing a large number of iterative updates at inference time, DU-AP adopts a Deep Unrolling strategy, unfolding a fixed number K of purification iterations into a feed-forward architecture. Each unrolled layer corresponds to one iteration of the underlying optimization process, resulting in a K-layer purification network with clear algorithmic interpretation (Figure 3.6).

This design provides a favorable balance between robustness and efficiency: it preserves interpretability, limits computational overhead, and enables DU-AP to function as a lightweight purification front-end for LiDAR semantic segmentation systems operating under real-time and embedded constraints.

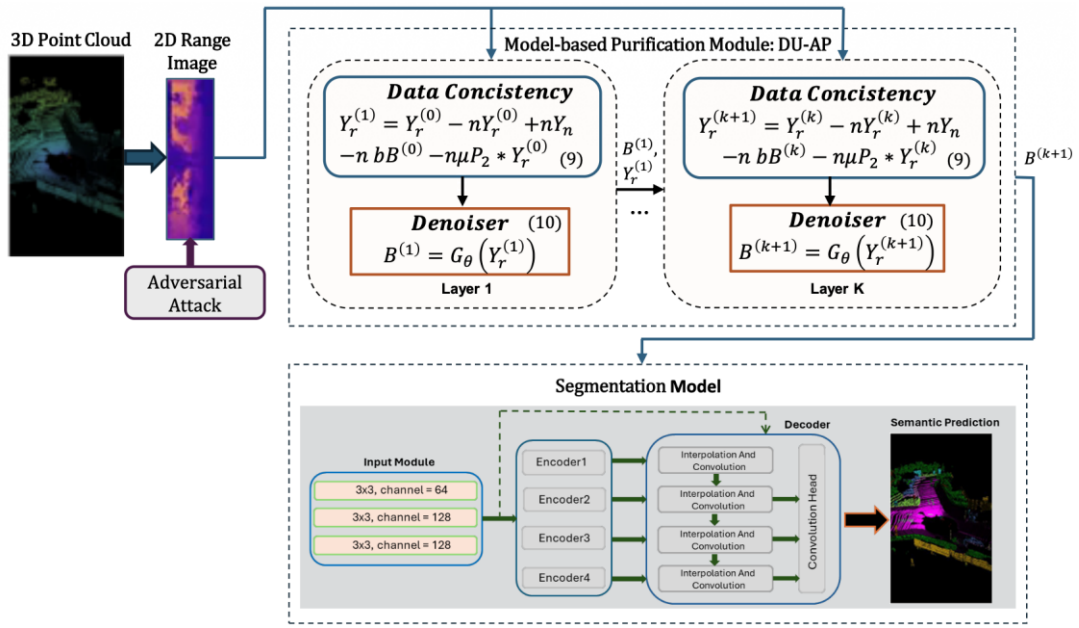


Figure 3.6: Illustration of the proposed model-based purification framework (DU-AP).

3.5.1.2. Experimental Evaluation

This section evaluates the effectiveness and practical relevance of DU-AP in improving adversarial robustness for LiDAR semantic segmentation. The evaluation focuses on three key aspects: (i) the impact of adversarial perturbations on range-view segmentation, (ii) the robustness gains provided by DU-AP on standard benchmarks, and (iii) the feasibility of deployment on real automotive hardware. DU-AP is compared

against a range-view–oriented robustness method[54] enhances the anti-attack capability through architectural and training-level modifications, and (ii) Adversarial Training on Purification (ATop)[55], a training-based defense that couples adversarial training with input purification.

Datasets and Experimental Setup

Experiments are conducted on two widely used LiDAR semantic segmentation benchmarks: SemanticKITTI[56] and SemanticPOSS[57], two widely used LiDAR segmentation benchmarks with different sensor configurations and resolutions. Adversarial examples are generated using PGD in the 2D range-view domain. Segmentation performance is measured using Intersection-over-Union (IoU).

Adversarial Impact on Range-View Segmentation

Table 9 compares adversarial perturbations crafted in the 3D point cloud domain and directly in the 2D range-view domain. Perturbations applied in the range-view domain result in substantially larger performance degradation, highlighting the observation that adversarial perturbations designed in the 3D domain are partially mitigated by the projection process due to quantization and geometric distortions, whereas perturbations applied directly in the range-view domain align with the true input representation of the segmentation network.

Table 9. Comparison of adversarial attack effectiveness in 2D vs. 3D domains for range-view segmentation.

	Original	Attack 3D	Attack 2D
IoU	0.527	0.444 (**–15.9%**)	0.365 (**–30.9%**)

Table 10 and Table 11 summarize the segmentation performance on SemanticPOSS and SemanticKITTI, respectively, under increasing adversarial perturbation strengths. Adversarial attacks cause significant degradation in segmentation accuracy, while DU-AP consistently recovers a large portion of the lost performance across all perturbation levels.

Table 10. SemanticPOSS: Performance comparison under adversarial attack and defenses. Parameter counts are shown as (defense) + (segmentation) in millions (M).

[54] Zhou, Q., Lei, M., Zhi, P., Zhao, R., Shen, J. and Yong, B. (2022). Towards improving the anti-attack capability of RangeNet++. In: Proceedings of the Asian Conference on Computer Vision (ACCV), pp. 56–67.

[55] Lin, G., Li, C., Zhang, J., Tanaka, T. and Zhao, Q. (2024). *Adversarial training on purification (ATOP): Advancing both robustness and generalization*. arXiv preprint, arXiv:2401.16352.

[56] Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C. and Gall, J. (2019) 'SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences', *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9297–9307.

[57] Pan, Y., Gao, B., Mei, J., Geng, S., Li, C. and Zhao, H. (2020) 'SemanticPOSS: A point cloud dataset with large quantity of dynamic instances', *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pp. 687–693.

IoU	ϵ=3	ϵ=6	ϵ=9	Params (M, defense+segm.)
No Attack	0.527	0.527	0.527	– + 4M
Only Attack	0.398	0.365	0.344	– + 4M
Attack & proposed DU-AP	**0.471**	**0.460**	**0.451**	0.05M + 4M
Attack & Method	0.449	0.431	0.432	– + 4M
Attack & ATOP	0.458	0.445	0.438	5M + 4M

Table 11. SemanticKITTI: Performance comparison under adversarial attack and defense.

IoU	ϵ=3	ϵ=6	ϵ=9	Parameters (M, defense+segm.)
No Attack	0.647	0.647	0.647	– + 4M
Only Attack	0.501	0.469	0.417	– + 4M
Attack & proposed DU-AP	**0.609**	**0.593**	**0.575**	0.05M + 4M
Attack & Method	0.564	0.557	0.531	– + 4M
Attack & ATOP	0.587	0.574	0.561	5M + 4M

Compared to alternative defense strategies, DU-AP achieves superior robustness while introducing minimal computational overhead. In contrast, GAN-based purification methods, such as ATOP[58], require significantly larger models, and adversarial retraining exhibits weaker recovery. The parameter efficiency of DU-AP, adding less than 1% additional parameters relative to the segmentation backbone, enabling real-time performance.

Class-wise results on SemanticPOSS (please see Table 12) further indicate that DU-AP improves robustness across both large and small object categories, including classes that are particularly sensitive to adversarial perturbations.

Table 12. Class-wise IoU on SemanticPOSS under Adversarial Attack $\boldsymbol{\varepsilon = 6}$.

IoU	avg	person	rider	car	trunk	plants	sign	pole	trash	building	stone	fence	bike	ground
No Attack	0.527	0.765	0.271	0.787	0.3	0.747	0.231	0.361	0.422	0.815	0.282	0.503	0.55	0.812
Only Attack	0.365	0.671	0.111	0.417	0.124	0.642	0.063	0.255	0.016	0.738	0.031	0.430	0.496	0.748
Attack & proposed DU-AP	0.460	0.743	0.204	0.547	0.251	0.708	0.174	0.316	0.304	0.787	0.153	0.463	0.542	0.791
Attack & Method	0.431	0.698	0.147	0.671	0.247	0.681	0.064	0.298	0.122	0.768	0.246	0.399	0.469	0.793
Attack & ATOP	0.445	0.698	0.163	0.528	0.238	0.696	0.162	0.302	0.271	0.774	0.164	0.471	0.529	0.790

Real-World Deployment and Practical Validation

To assess practical feasibility, DU-AP is deployed on a demo vehicle equipped with an NVIDIA Jetson Orin and a 16-channel Velodyne LiDAR. Real-world experiments are conducted under realistic operating conditions, using pseudo ground-truth labels derived from the segmentation model's predictions on clean data.

Real-world experiments confirm that adversarial attacks degrade segmentation performance, while DU-AP restores semantic coherence (please see Table 13).

[58] Lin, G., Li, C., Zhang, J., Tanaka, T. and Zhao, Q. (2024) 'Adversarial training on purification (ATOP): Advancing both robustness and generalization', *arXiv preprint*, arXiv:2401.16352.

Table 13. Demo vehicle: Performance comparison. Evaluation is based on pseudo ground truth labels derived from the segmentation network's predictions on clean data.

IoU	ϵ=1	ϵ=2	ϵ=3	Params (M, defense+segm.)
Only Attack	0.437	0.361	0.325	– + 4M
Attack & proposed DU-AP	**0.541**	**0.437**	**0.401**	0.05M + 4M
Attack & method	0.461	0.384	0.357	– + 4M
Attack & ATOP	0.501	0.412	0.387	5M + 4M

From a deployment perspective, computational efficiency is critical. While GAN-based purification methods achieve reasonable robustness, their large parameter count, and inference cost make them impractical for real-time use on embedded automotive platforms. DU-AP, by contrast, introduces over 99% fewer parameters, confirming its suitability for integration into real-world autonomous perception pipelines.

3.5.1.3. Next steps

Future work will focus on further consolidating the robustness and efficiency of the proposed DU-AP framework. This includes extending the experimental evaluation to a broader set of adversarial defense methods and additional LiDAR datasets, enabling a more comprehensive assessment of generalization. In parallel, the adversarial threat model will be expanded to include stronger white-box attacks, and additional black-box scenarios. Finally, further architectural refinements will be explored to improve computational efficiency, with the goal of minimizing latency and resource consumption for real-time embedded deployment.

3.5.2. Imbalance-aware Learning as a Robustness Strategy

Robustness in 3D point cloud semantic segmentation remains a critical challenge for deploying deep learning systems in safety-critical applications[59]. Unlike 2D image classification, where pixel-level perturbations have been extensively studied, 3D point clouds present unique vulnerabilities due to their unstructured nature and direct geometric representation. The absence of a regular grid structure means that perturbations to point coordinates can fundamentally alter the geometric relationships that neural networks rely upon for semantic understanding.

A particularly understudied aspect of robustness in 3D segmentation is the interplay between class imbalance and model vulnerability to adversarial perturbations. Real-world LiDAR datasets exhibit severe class imbalance, with majority classes comprising 40-60% of points while critical minority classes (e.g., poles, power lines, vehicles, pedestrians, etc.) constitute less than 6% of the data[60]. This imbalance not only affects standard performance metrics but also shapes the optimization landscape, with profound implications for model robustness. This section argues that understanding

[59] Y. Liu, *et al*, "A Review of Semantic Segmentation for Large-scale Point Cloud Data," 2023 9th International Conference on Big Data and Information Analytics (BigDIA), Haikou, China, 2023, pp. 662-671.

[60] Y. Pan, *et al.*, "Understanding the Challenges When 3D Semantic Segmentation Faces Class Imbalanced and OOD Data," in IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 7, pp. 6955-6970, July 2023.

and manipulating the loss landscape through imbalance-aware learning strategies constitutes a fundamental robustness mechanism.

3.5.2.1. The loss landscape framework

The optimization landscape for 3D point cloud segmentation can be characterized by examining local curvature properties at converged model parameters θ^*. These geometric properties (quantified through Hessian eigenvalue spectra) reveal whether a trained model occupies a sharp, narrow minimum or a broad, flat basin in parameter space. This distinction is critical because flat minima are strongly associated with improved generalization and robustness, while sharp minima indicate fragile solutions that perform poorly under distribution shifts or adversarial perturbations.

For a trained segmentation model f_θ processing point clouds $P = \{p_i\}^N \; with \; p_i \in \mathbb{R}^{(3+d)}$ (coordinates plus optional features), the loss landscape around the converged solution θ^* determines how the model responds to both natural variations and adversarial manipulations of input geometry. The fundamental question is: *how does class imbalance shape this landscape, and can we leverage this understanding to improve robustness?*

For our assessment, we used the following methods:

- Uniform (uni): standard cross-entropy with equal weight ($w_c = 1$) for all classes (serves as the baseline).
- Inverse frequency (invf)**:** weights inversely proportional to class size: $w_c = N/n_c$, where N is total points and n_c is points in class c.
- Class-balanced (cb)[61]: uses effective number of samples: $w_c = (1-\beta)/(1-\beta^{n_c})$ with $\beta = 0.9$.
- Inverse logarithm (invl): logarithmic damping of frequency: $w_c = 1/log(n_c)$.
- Inverse power (invp): power-law scaling: $w_c = 1/n_c^\gamma$ with $\gamma = 0.1$.
- Complementary frequency (comf): weight as proportion of absent samples: $w_c = 1 - n_c/N$.
- Focal loss (FL)[62]: modulates standard cross-entropy by down-weighting well-classified examples: $L = -(1-p_y)^\gamma \; log(p_y)$ with $\gamma = 1$.
- LDAM (Label-Distribution-Aware Margin)[63]: enforces larger decision margins for minority classes: $\Delta_c \propto 1/n_c^{1/4}$.
- LADJ (Logit Adjustment)[64]: adjustment of logits based on class priors: $\tilde{z}_c = z_c - \tau \; log(\pi_c)$ with $\tau = 0.3$.

[61] Y. Cui, et al., "Class-Balanced Loss Based on Effective Number of Samples," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 9260-9269.

[62] T. -Y. Lin, et al., "Focal Loss for Dense Object Detection," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2999-3007.

[63] K. Cao, et al., “Learning imbalanced datasets with label-distribution-aware margin loss,” in Advances in Neural Information Processing Systems, vol. 32, H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, Inc., 2019.

[64] A. K. Menon, et al., “Long-tail learning via logit adjustment,” in Proceedings of the International Conference on Learning Representations (ICLR), 2021.

- BalSoft (Balanced Softmax)[65]: incorporates class frequencies directly into softmax normalization, rebalancing posterior probabilities.
- Seesaw Loss[66]: Dynamically balances mitigation factors (reducing penalties for tail classes) and compensation factors (increasing penalties for misclassifications).

Figure 3.7 presents the segmentation results when applying these methods in DALES[67] (upper panel) and S3DIS[68] (lower panel) datasets, using KPConv[69]. On DALES, uni achieves 80.05% mIoU, competitive with the best performing methods (invp: 80.86%, invl: 80.69%, comf: 80.72%, LDAM: 80.63%), while extreme reweighting (invf) and BalSoft significantly reduce performance. On S3DIS, all methods cluster tightly within 62.92-64.85 % mIoU (1.93% range). The last row shows the performance range across methods, revealing that minority classes (e.g., DALES *poles*: 36.08%, S3DIS *window*: 10.43%) exhibit greater variance, while majority classes remain stable.

Method	mean	*ground*	*vegetation*	*cars*	*trucks*	*power lines*	*fences*	*poles*	*buildings*
uni	80.047	96.507	93.781	85.143	42.650	94.029	61.092	72.262	94.915
invf	67.780	95.885	91.111	71.534	32.470	86.406	31.041	40.163	93.632
cb	78.534	96.445	93.474	85.103	43.480	94.763	57.071	63.121	94.816
invl	80.692	96.488	93.846	84.828	43.504	94.476	62.659	74.936	94.802
invp	80.861	96.518	93.795	85.137	44.374	94.647	63.169	74.299	94.945
comf	80.715	96.474	93.759	85.192	43.103	94.646	63.280	74.335	94.932
FL	80.039	96.495	93.767	85.025	43.339	93.790	62.402	70.609	94.884
LDAM	80.629	96.510	93.828	85.260	44.550	94.510	62.524	72.894	94.957
LADJ	79.598	96.561	93.665	84.088	42.359	94.277	59.860	70.928	95.047
BalSoft	68.136	96.126	91.771	74.583	22.637	93.752	33.658	38.856	93.708
Seesaw	80.360	96.538	93.826	85.091	43.641	93.990	62.984	71.885	94.924
Range	13.081	0.676	2.735	13.726	21.913	8.357	32.239	36.080	1.415

[65] J. Ren, et al., “Balanced meta-softmax for long-tailed visual recognition,” in Advances in Neural Information Processing Systems, vol. 33, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds. Red Hook, NY, USA: Curran Associates, Inc., 2020, pp. 4175–4186.
[66] J. Wang et al., "Seesaw Loss for Long-Tailed Instance Segmentation," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 2021, pp. 9690-9699,
[67] N. Varney, *et al.*, "DALES: A Large-scale Aerial LiDAR Data Set for Semantic Segmentation," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 2020, pp. 717-726.
[68] I. Armeni et al., "3D Semantic Parsing of Large-Scale Indoor Spaces," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 1534-1543.
[69] H. Thomas, et al., "KPConv: Flexible and Deformable Convolution for Point Clouds," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 6410-6419.

Method	mean	*ceiling*	*floor*	*wall*	*beam*	*column*	*window*	*door*	*chair*	*table*	*bookcase*	*sofa*	*board*	*clutter*
uni	63.097	93.693	98.514	80.862	0.000	21.341	43.934	61.597	87.347	79.061	71.098	65.612	59.560	57.642
invf	63.683	92.264	98.314	80.297	0.000	25.512	46.554	60.456	87.337	79.534	71.026	71.432	60.282	54.871
cb	63.556	92.649	98.392	80.499	0.000	24.290	45.111	62.371	87.402	78.244	70.891	67.834	62.327	56.221
invl	63.534	93.449	98.498	80.746	0.000	23.703	45.545	59.198	87.668	79.424	71.799	67.024	60.248	58.635
invp	62.915	92.995	98.416	80.498	0.000	23.474	46.459	60.386	86.864	78.853	70.282	60.821	61.380	57.465
comf	63.534	92.937	98.398	81.069	0.000	21.561	46.451	66.563	87.381	79.507	70.915	62.936	61.760	56.469
FL	63.247	92.213	98.394	79.701	0.000	22.118	44.992	60.418	87.755	79.353	70.239	67.789	62.686	56.555
LDAM	63.268	92.684	98.476	80.578	0.000	24.958	45.116	62.027	87.850	78.675	71.300	63.466	60.798	56.553
LADJ	63.999	92.834	98.405	81.447	0.000	24.748	49.772	62.241	87.156	78.881	71.915	64.341	63.757	56.492
BalSoft	64.848	93.138	98.383	82.602	0.000	28.378	54.368	64.392	86.916	77.966	71.283	66.948	63.054	55.598
Seesaw	63.610	92.922	98.453	80.586	0.000	21.489	47.755	62.613	87.691	78.743	70.777	67.915	62.434	55.549
Range	1.933	1.480	0.200	2.901	0.000	7.037	10.434	7.365	0.986	1.568	1.676	10.611	4.197	3.764

Figure 3.7: Segmentation results when applying these methods in DALES (upper panel) and S3DIS (lower panel) datasets.

3.5.2.2. Dataset-dependent landscape topologies

The empirical analysis across the two diverse datasets reveals that class imbalance ratio fundamentally determines landscape topology (Figure 3.8, Figure 3.9):

Extreme imbalance (DALES with 641:1 ratio)

- Anisotropic curvature structure: Hessian eigenvalue spectra[70] show a dominant first eigenvalue (λ_1) with rapid decay, indicating one sharp direction with relatively flat orthogonal directions.
- Narrow basin geometry: Methods exhibiting high decision boundary variability (>0.020) relative to standard cross-entropy with uniform weighting suffer severe performance degradation (>12% mIoU reduction).
- Strong negative correlation (Spearman $r_s = -0.874, p < 0.001$) between decision boundary[71] deviation and performance, suggesting that the uniform weighting solution occupies a favorable but confined basin.

Moderate imbalance (S3DIS with 56:1 ratio)

- Isotropic curvature structure: Eigenvalues λ_1 through λ_6 maintain similar magnitudes before declining, revealing multiple directions with comparable sharpness.
- Flat plateau geometry: All methods cluster within 1.93% mIoU despite decision boundary variability ranging from 0.047 to 0.066.
- Weak positive correlation (Spearman $r_s = 0.768, p = 0.009$) between decision boundary deviation and performance, indicating that diverse boundary configurations yield similar results.

[70] H. Li, et al. , "Visualizing the loss landscape of neural nets," in Advances in Neural Information Processing Systems, vol. 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, Inc., 2018.
[71] S. Lei, *et al.*, "Understanding Deep Learning via Decision Boundary," in IEEE Transactions on Neural Networks and Learning Systems, vol. 36, no. 1, pp. 1533-1544, Jan. 2025, doi: 10.1109/TNNLS.2023.3326654.

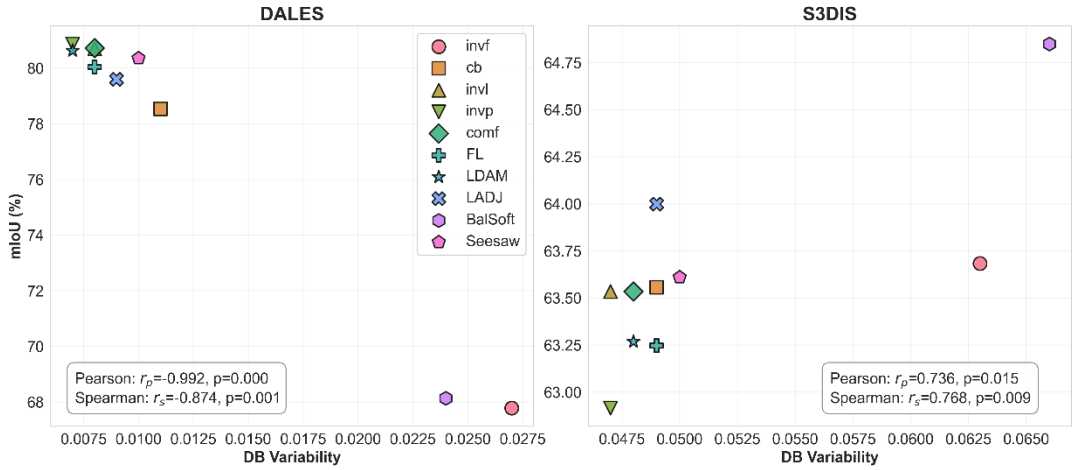


Figure 3.8: Decision boundary variability vs. mIoU on DALES (left) and S3DIS (right). Each point represents one method; DB variability measures decision boundary divergence from uniform weighting (uni). DALES exhibits a strong negative correlation, with large deviations causing significant performance degradation (invf, BalSoft: >12% mIoU drop). S3DIS exhibits a weak positive correlation with all methods clustered within ≈2% mIoU.

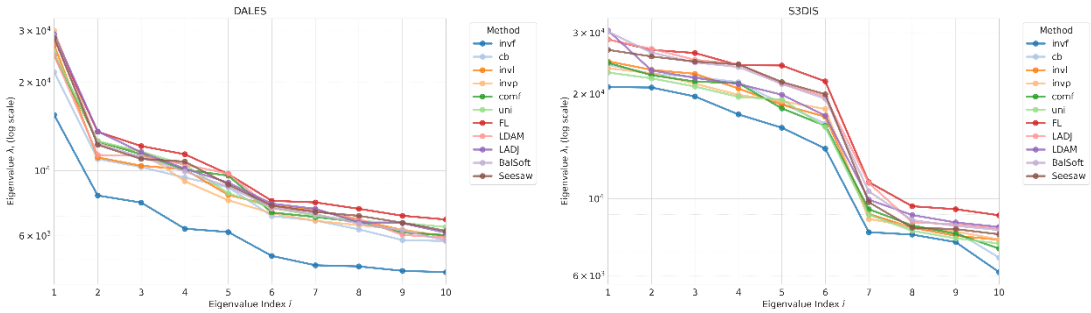


Figure 3.9: Hessian eigenvalue spectra (log scale) for all methods. DALES exhibits anisotropic curvature, while S3DIS shows more isotropic curvature.

3.5.2.3. The precision-recall trade-off and geometric consistency

Analysis of confusion matrices reveals why aggressive reweighting fails under extreme imbalance and why specialized methods cannot consistently outperform uniform weighting. The fundamental issue is geometric inconsistency in false positives:

- Uniform weighting (DALES): Achieves balanced error rates (0.15% majority-to-minority, 12.78% minority-to-majority misclassifications).
- Aggressive reweighting (inverse frequency) and balanced-softmax loss: Successfully reduce minority-to-majority errors to ~1.6% (improving recall) but inflates majority-to-minority errors by 10× to ~1.5% (degrading precision).

In 3D point clouds where semantic classes exhibit strong geometric structure (vertical poles, horizontal ground planes, volumetric vegetation), aggressive reweighting causes models to misclassify geometrically similar majority classes as minority classes. This precision collapse indicates that the model has learned to over-prioritize features of the minority class at the expense of maintaining coherent geometric relationships. Such solutions are inherently vulnerable to geometric perturbations, as

they rely on brittle features that can be easily manipulated through coordinate-level adversarial attacks.

3.5.2.4. Robustness implications of landscape topology

The connection between landscape topology and robustness emerges from several theoretical considerations:

1. Flat minima generalize better: Solutions in broad basins are less sensitive to perturbations in both parameter space and input space, providing natural resistance to adversarial attacks.
2. Geometric consistency as a robustness prior: Methods that maintain low majority-to-minority misclassification rates preserve geometric plausibility, making it harder for adversaries to induce coordinate perturbations.
3. Low decision boundary variability relative to geometrically sensible baselines (uniform weighting) indicates that the model has learned stable semantic boundaries aligned with true geometric structure.
4. Extreme imbalance creates anisotropic landscapes with sharp directions that adversaries can exploit, while moderate imbalance produces more isotropic geometry with no single vulnerable direction.

3.5.2.5. Adversarial attacks on point cloud geometry

To empirically validate the connection between imbalance-aware landscape topology and robustness, we employ adversarial attacks specifically designed to perturb the geometric coordinates of point clouds while preserving any additional features (e.g., RGB values in S3DIS). This focus on geometry-only perturbations is crucial because it: (a) directly tests the model's reliance on learned geometric relationships between points; (b) simulates realistic sensor noise and calibration errors in LiDAR systems; and (c) avoids confounding effects from feature-space perturbations that may not be physically realizable.

Attack Definitions and Implementation

Gaussian noise attack represents the simplest form of geometric perturbation, simulating random sensor measurement errors. For each point $p_i = (x_i, y_i, z_i)$ in the point cloud, we add independent Gaussian noise to each coordinate:

$$p_i' = p_i + \varepsilon \cdot n_i, \text{ where } n_i \sim N(0, I_3) \quad (12)$$

The noise magnitude ε is scaled relative to the point cloud's spatial extent to ensure perturbations are meaningful but not visually obvious. This attack perturbs only the (x, y, z) coordinates, leaving any additional features unchanged. While Gaussian noise is non-adversarial (i.e., not gradient-guided), it serves as a baseline for measuring inherent model robustness to geometric uncertainty.

Fast Gradient Sign Method (FGSM) is a white-box adversarial attack that generates perturbations by taking a single gradient step in the direction that maximally increases the loss function. For point cloud segmentation, we compute:

$$\delta = \varepsilon \cdot sign(\nabla_p L(f_\theta(p), y_{true})) \quad (13)$$

where L is the segmentation loss, f_θ is the trained model, and y_{true} are the ground truth labels. The perturbation δ is then applied to the point coordinates:

$$p' = p + \delta \quad (14)$$

Critically, the gradient ∇_p is computed only with respect to the geometric coordinates (x, y, z), not with respect to any additional features. This ensures that the attack manipulates only the spatial configuration of points.

Projected Gradient Descent (PGD) is an iterative extension of FGSM that represents a much stronger adversary. It performs multiple gradient steps, each followed by projection back onto an ε-ball around the original point coordinates:

$$p^{t+1} = \Pi_{\{||\delta||_\infty \leq \varepsilon\}}(p^t + \alpha \cdot sign(\nabla_p L(f_\theta(p^t), y_{true}))) \quad (15)$$

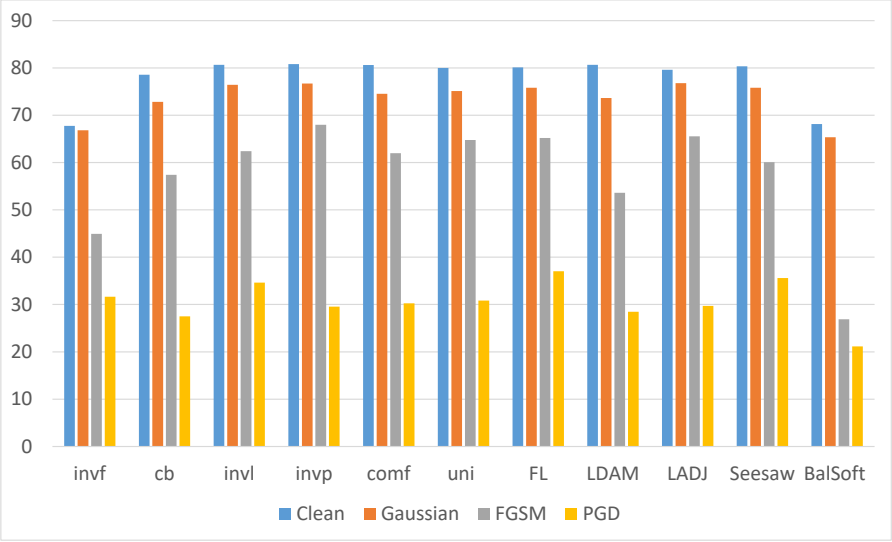
where α is the step size, typically set to $\varepsilon/4$, and Π denotes projection. The projection ensures that perturbations remain bounded and physically plausible (e.g., preventing points from moving arbitrarily far from their original positions).

As before, gradients are computed and applied only to the (x, y, z) geometric coordinates. PGD is considered one of the strongest first-order adversarial attacks because it explores the local loss landscape more thoroughly than FGSM, finding perturbations that more reliably induce misclassifications.

The attacks are configured to $\varepsilon = 0.2cm$ and for PGD we used 10 iterations.

3.5.2.6. Imbalance strategies under adversarial attacks

Under adversarial attack, the performance gap between different imbalance mitigation strategies becomes dramatically more pronounced than under clean evaluation. The key finding is that methods that maintain low decision boundary variability and preserve geometric consistency (low majority-to-minority misclassification) demonstrate significantly better robustness.



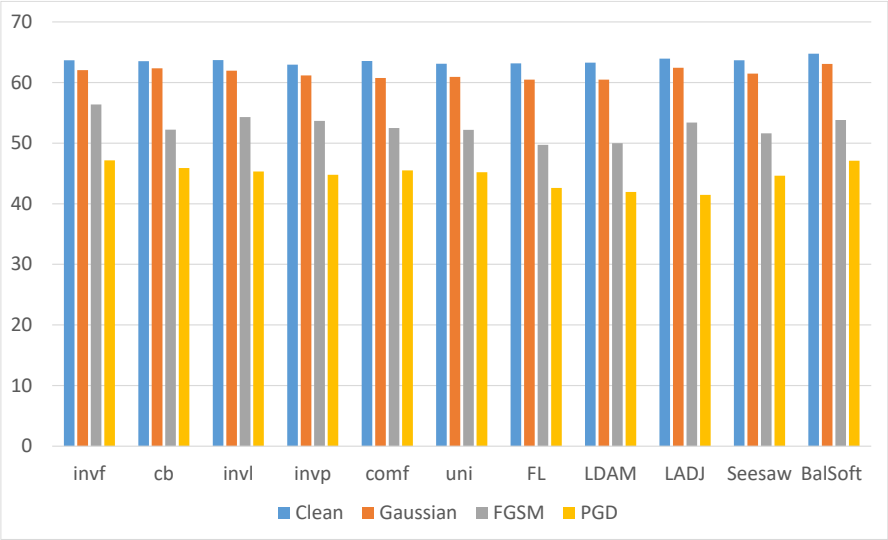


Figure 3.10: Performance under adversarial attacks for DALES (upper panel) and S3DIS (lower panel) datasets.

3.5.2.7. Conclusion and future steps

Our analysis establishes a direct mechanistic connection between class imbalance mitigation strategies, loss landscape topology, and adversarial robustness in 3D point cloud segmentation:

- Extreme imbalance (641:1) creates narrow, anisotropic basins with limited solution space, while moderate imbalance (56:1) produces flat, isotropic plateaus with multiple viable solutions.
- All imbalance-aware methods converge to regions with similar local curvature within each dataset. This implies that the data distribution itself constrains the achievable solution space more strongly than loss function design.
- Methods that preserve low majority-to-minority misclassification rates (maintaining geometric plausibility) demonstrate superior resistance to coordinate-level adversarial attacks. Aggressive reweighting sacrifices precision for recall, creating brittle decision boundaries that amplify adversarial vulnerability.

These findings have immediate practical implications for designing robust 3D segmentation systems:

- Leverage landscape topology based on imbalance characteristics of the dataset, i.e., prefer mild reweighting strategies over aggressive inverse frequency weighting for extreme imbalance scenarios (>100:1); and for moderate scenarios (20:1 to 100:1) topology allows more flexibility in method selection without catastrophic robustness penalties.
- Methods should explicitly penalize geometrically implausible predictions to maintain consistency under perturbation
- Clean performance metrics alone are insufficient; models should be evaluated under PGD as a robustness certification step.

As next steps, we will examine:

- How do different network architectures (transformer-based models, graph convolutions, sparse convolutions) shape landscape topology under class imbalance?
- The transition zone between extreme (>500:1) and moderate (<100:1) imbalance to systematically investigate optimal strategy selection criteria.
- The applicability of segmentation networks in edge devices and how their robustness is affected.

3.6. Visual Dataset Augmentation workflow and Poisoning Pipeline for the Detection of Attacks

3.6.1. Visual Dataset Augmentation techniques

Visual dataset augmentation is the practice of synthetically expanding a training set by transforming existing images to mimic the variability a model will face in the real world, without collecting new data. It uses label-preserving operations (flips, crops, color/exposure shifts), photometric corruptions (noise, blur, compression), and environment-driven effects (fog, rain, snow) that helps:

- Expanding data diversity without costly recollection or annotation.
- Improving robustness to common corruptions (noise, blur, weather, compression, brightness/contrast).
- Reducing overfitting by exposing models to many appearance variants.
- Mimicking deployment conditions (e.g., foggy mornings, rain streaks, low light, compression artifacts) so decision boundaries stabilize.

For UGVs in particular, where operating environments vary widely (early/late light, dust, weather, motion) and may include adversarial interference (e.g., glare or injected noise),a data augmentation process is essential to train models that retain performance as conditions degrade.

All augmentation techniques are based on imgaug[72], a lightweight, flexible Python library for image/video augmentation.

3.6.2. List of Augmentators

Currently we are focused on two families of augmentations:

- Common corruption & photometric changes (random noise, blur, brightness, contrast, grayscale, flips, crops, compression/pixelation).
- Weather-like conditions (clouds/fog/rain/snow) and combined weather-like conditions.

Table 14 summarizes the series of augmentators that have been developed.

Table 14. Summary of the developed augmentators.

	Family	Augmentator	Purpose
1	Noise/Blur/Exposure	gaussian_noise	Adds pixel-wise zero-mean noise (σ scaled by severity) to mimic sensor/ISP

[72] https://imgaug.readthedocs.io/en/latest/

			noise, RF interference, or low-light amplification artifacts. Useful for testing noise robustness.
2		gaussian_blur	Applies Gaussian blur approximating a point-spread function (defocus, slight misfocus, or mild atmospheric scatter). Helps models cope with soft images and lens imperfections.
3		brightness	Multiplies image brightness uniformly to emulate exposure drift, auto-gain quirks, sunrise/sunset light, or headlight/glare effects. Stresses robustness to luminance changes.
4		contrast / linear_contrast	Linearly expands/compresses intensity differences to simulate haze, fog, or harsh lighting. Ensures features remain detectable across low/high contrast scenes.
5		grayscale	Removes chroma information (keeps luminance only). Useful when color is unreliable or misleading, or to force reliance on shape/texture cues.
6		jpegCompression	Re-encodes at low quality to introduce blocking, ringing, and banding,common in low-bandwidth links or onboard storage compression.
7		pixelate	Pixelates by shrinking then enlarging, producing block artifacts akin to digital zoom or streaming resolution drops.
8		saturate	Shifts hue/saturation to mimic sensor color drift, white-balance errors, or environmental color casts (e.g., dusk/LEDs). Tests color invariance.
9		motionblur	Directional blur to simulate camera/platform motion or moving subjects during exposure (UGV jolts, turns, rough terrain
10		flip (H/V)	Mirrors geometry to balance left/right or up/down biases. Preserves labels for most tasks and reduces viewpoint overfitting.
11		crop_random	Independently crops small margins from each edge, perturbing framing/centering. Encourages robustness to partial views and off-center subjects.
12		crop_center	Removes a uniform border from all sides, emphasizing central content and simulating tighter FOV or digital zoom without resampling.
13	Weather-like	clouds	Adds low-frequency cloud texture with alpha blending to dim sky/ambient light and flatten shadows. Emulates overcast days.
14		fog	Fog/haze effect (broad, soft veil) that reduces contrast and desaturates distant regions. Mimics scattering in humid or misty conditions.

15		rain	Adds slanted, semi-transparent streaks; often paired with motion blur and slight darkening. Tests robustness under wet, dynamic conditions.
16		snowflakes	Overlays small bright flakes (density/size by severity) that partially occlude features and alter local contrast, including sleet-like pellets when combined with motion.
17		snowy_landscape	Boosts light areas and suppresses darker tones to emulate snow cover/whiteout where edges and textures collapse.
18		overcast_haze	Simulates a dull, overcast day with a soft, milky haze.
19		morning_fog_bank	Models a ground-hugging fog bank rolling through the scene
20		drizzle / spatter	Adds fine, semi-transparent droplets or streaklets that accumulate on the image.
21		heavy_rain	Simulates intense rainfall with visible streaks, global dimming, and motion-induced smear.
22		pellet_snow (sleet)	Represents compact, fast “pellet” flakes (sleet) that look like tiny bright dots with short streaks.
23		whiteout_blizzard	Emulates a near-whiteout: dense flakes, strong fog veil, and scene whitening that collapses edge contrast.

Each augmentator supports severity levels (1–5) or is performed using static values. Latter parameters are set so that severity progresses monotonically from subtle (s1) to heavy (s5) while remaining realistic for UGV scenes. Table 15 lists the main augmenters developed and how their severities are controlled.

Table 15. Main augmenters and configuring parameters.

Augmentator	Parameter (s)	s=1	s=2	s=3	s=4	s=5
gaussian_noise	σ	0–3	0–6	0–12	0–24	0–40
gaussian_blur	σ	0.2–0.4	0.3–0.6	0.5–0.9	0.7–1.2	1.0–1.6
brightness	Multiply factor	0.90–1.10	0.80–1.20	0.70–1.30	0.60–1.40	0.50–1.50
linear_contrast	Factor	0.9–1.1	0.8–1.2	0.7–1.3	0.6–1.4	0.5–1.5
saturate	Hue/Sat offset	−5…0	−10…−3	−15…−5	−20…−8	−25…−10
grayscale	α (to gray)	1.0 (static)				
jpegCompression	JPEG quality	90–85	80–70	60–50	40–30	25–15
pixelate	Downscale factor	0.9–0.8	0.8–0.6	0.6–0.45	0.45–0.35	0.35–0.25
motionblur	kernel size *k*	5–7	7–9	9–11	11–15	15–21
flip (H/V)		static on/off				
crop_random	per-side crop % (max)	0–5%	0–10%	0–20%	0–30%	0–40%

crop_center	center-crop kept area	95–100%	90–100%	80–100%	70–100%	60–100%
overcast_haze	Cloud α (BlendAlpha)	0.30–0.50	0.50–0.70	0.60–0.85	0.75–0.95	0.90–1.00
	Fog veil α (BlendAlpha)	0.20–0.30	0.25–0.40	0.35–0.55	0.50–0.70	0.65–0.85
	Contrast factor	0.95–1.00	0.90–0.98	0.85–0.95	0.80–0.92	0.75–0.90
	Blur σ	0.0–0.3	0.2–0.5	0.3–0.7	0.4–1.0	0.6–1.2
morning_fog_bank	Fog α (low-freq clouds)	0.25–0.40	0.35–0.55	0.45–0.65	0.60–0.80	0.75–0.95
	Contrast factor	0.95–1.00	0.90–0.98	0.85–0.95	0.80–0.92	0.75–0.90
	Blur σ	0.0–0.3	0.2–0.5	0.3–0.7	0.5–1.0	0.8–1.5
drizzle / spatter	Spatter severity					
heavy_rain	Rain severity	s=1	s=2	s=3	s=4	s=5
	MotionBlur *k*	7–9	9–11	11–13	13–17	17–21
	Contrast factor	0.95–1.00	0.90–0.98	0.85–0.95	0.80–0.92	0.75–0.90
	Darken multiply	0.95–1.00	0.92–0.98	0.90–0.97	0.88–0.96	0.85–0.95
pellet_snow (sleet)	Snow severity (if available)	s=1	s=2	s=3	s=4	s=5
	MotionBlur *k*	3–5	5–7	7–9	9–11	11–13
	Desat (Hue/Sat)	−5…0	−10…−3	−15…−5	−20…−8	−25…−10
	Contrast factor	0.98–1.00	0.95–0.99	0.90–0.97	0.85–0.95	0.80–0.92
	Darken multiply	0.98–1.00	0.96–0.99	0.94–0.98	0.92–0.97	0.90–0.96
whiteout_blizzard	Snow severity (dense)	s=1	s=2	s=3	s=4	s=5
	Fog α (BlendAlpha)	0.50–0.70	0.65–0.85	0.75–0.92	0.85–0.97	0.90–1.00
	Snowy landscape α	0.10–0.20	0.15–0.30	0.20–0.45	0.30–0.60	0.40–0.80
	Contrast factor	0.90–0.98	0.85–0.95	0.80–0.92	0.70–0.88	0.60–0.85
	Blur σ	0.0–0.4	0.2–0.6	0.4–0.9	0.6–1.2	0.8–1.6

3.6.3. Augmentation Workflow

Each augmentator is packaged as a separate Docker service [73](one image per effect). All services share a common host directory layout:

[73] https://www.docker.com/

- shared/input_images/ – the source images (mounted read-only in the container).
- shared/output_augmented_images/ – where each service writes its results.

This isolation makes it trivial to compose, scale, and reproduce the experiments: On start, the service:

1. Parses configuration from environment variables (augmentor name, severity levels, sampling mode, etc.).
2. Discovers input images via glob patterns (such as jpg, jpeg, png, bmp, tif, and tiff) under the input directory.
3. Optionally samples the set (first N or a random N images) to accelerate iteration.
4. Builds a severity-specific augmenter and processes all selected images.
5. Writes outputs in the respectively output folder among with per output logs files metrics.
6. Exits cleanly, with no restart after the batch completes.

Because each augmentator is its own container, they can run in parallel (different services) without stepping on each other, while they can pin different dependency stacks per augmentator if needed. Table 16 represents a typical compose service for the augmentator, while Table 17 represents the runtime and environmental variables used for the augmenter service.

Table 16. Compose service for the augmentator.

```
<name_of_the_ augmentator>:
  build:
    context: <path_to_the_context_of_the_ augmentator>
    dockerfile: Dockerfile
  container_name: <name_of_the_container_of_the_augmentator>
  volumes:
    - ./shared/input_images:/app/input_images:ro
    - ./shared/output_augmented_images:/app/output_augmented_images
  environment:
    AUG: <name_of_the_ augmentator>
    SEVERITY: "all"       # or "1,3,5" or "4"
    MAX_IMAGES: "50"      # 0 for all
    SAMPLE_MODE: "random" # or "first"
  restart: "no"
```

Table 17. Runtime and environmental variables used for the augmenter service.

Variable	Type / Allowed Values	Default	Purpose
AUG	string	augmentor name (e.g., overcast_haze)	Used for naming the output subtree of the output folders.
SEVERITY	"all" or 1–5 or single 1..5	"all"	Which severity levels to run. "all" expands to all levels of severity.

MAX_IMAGES	integer	0	If 0, process all discovered images; otherwise process only the N images
SAMPLE_MODE	"first" or "random"	"first"	If MAX_IMAGES > 0, choose the first N by sorted filename or pick N unique random files.

3.6.4. Visual Attack detection & Resilient Decision-making in CAVs

Perceptual Image Hashing

A digital image consists of a grid of pixels, and each pixel is defined by three numbers representing the intensity of the red, green, and blue (RGB) color channels. In standard 8-bit images, each channel ranges from 0 to 255. For instance, a bright red pixel is encoded as (255, 0, 0), a faint blue pixel as (0, 0, 64), black as (0, 0, 0), medium grey as (128, 128, 128), and pure white as (255, 255, 255). A 12-megapixel camera sensor typically captures an image of roughly 3000 × 4000 pixels. Because each of these pixels can take 256 values across three channels, the number of possible images this sensor can produce is astronomically large, far beyond the estimated 10^{24} stars in the universe.

Perceptual hashing aims to replicate how humans judge whether two images depict the same visual content, rather than relying on direct pixel-by-pixel comparisons. It works by condensing the enormous pixel-level information into a compact, distinctive, and visually meaningful fingerprint that remains stable even when the image undergoes modifications such as compression, color adjustments, cropping, rotation, or the addition of logos or text, changes that alter pixel values but not the essential scene. At the same time, generating and comparing these perceptual hashes must be computationally lightweight to handle billions of images processed every day.

Visual perception is fundamental to the operation of Connected and Autonomous Vehicles (CAVs), enabling essential functions, such as object detection, lane keeping, obstacle avoidance, and overall situational awareness. These capabilities depend on the accurate interpretation of key contextual factors, including environmental conditions, urban infrastructure, and expected system performance. Vision-based sensors, primarily cameras, often complemented by LiDAR (Light Detection and Ranging), serve as the core input mechanisms upon which higher-level decision-making and control processes are built. As a result, maintaining the integrity and trustworthiness of visual data is crucial for ensuring safe and reliable autonomous navigation.

In contrast to cryptographic hash functions, which operate directly on raw bytes and yield entirely different outputs even when an image is only minimally altered, perceptual image hashing focuses on extracting stable visual characteristics. This enables images that appear alike to humans to produce similar hash values, while visually unrelated images generate hash outputs with no meaningful correlation[74].

Hashing Techniques

Hashing techniques provide compact and visually meaningful representations of images, enabling efficient comparison based on content rather than raw pixel data.

[74] Ferenčak, M., Grd, P. and Tomičić, I., 2023, May. The impact of image processing on perceptual hash values. In *2023 46th MIPRO ICT and Electronics Convention (MIPRO)* (pp. 1070-1075). IEEE.

Among the most widely used approaches are average hash (aHash), differential hash (dHash), perceptual hash (pHash), and wavelet hash (wHash), each capturing different aspects of the underlying visual structure.

Average hashing (aHash) simplifies an image by converting it to grayscale, resizing it to a small, fixed grid, and computing whether each pixel is above or below the mean intensity; this produces a binary pattern that reflects the image's overall luminance distribution. The aHash summarizes an image's global appearance by emphasising low-frequency information and suppressing fine-grained details. The algorithm first applies a blur, scales the image to an 8x8 resolution, and converts it to grayscale (YCbCr). It then computes the mean intensity value across all pixels and assigns each pixel a bit of 1 if its intensity is equal to or greater than this average, and 0 otherwise. Reading these binary values in row-major order yields a compact 64-bit hash that serves as a basis for similarity assessment using a chosen distance metric.

Differential hashing (dHash) instead focuses on the gradients within the image: after shrinking it to a compact grayscale grid, it records whether each pixel is brighter than its neighbour, effectively encoding edge information and local transitions that remain stable under minor transformations. dHash generates a compact 64-bit descriptor by encoding how pixel intensities change across the image. The method begins by converting the image to grayscale and scaling it down to an 8x8 matrix. It then compares each pixel with the one directly to its right along every row. When the neighbouring pixel has a higher intensity, the algorithm assigns a bit value of 1; otherwise, it assigns 0. The concatenation of these binary comparisons across all rows forms a distinctive 64-bit hash that characterizes the image's local gradient patterns.

Perceptual hashing (pHash) builds on the idea of structural similarity by applying a discrete cosine transform (DCT) to the image and capturing the dominant frequency components, creating a signature that is particularly robust against noise, compression, and small geometric distortions. pHash produces a 64-bit hash by analysing an image in the frequency domain. The image is first reduced to 32×32 pixels, which is sufficient to retain the dominant low- and mid-frequency components, and then converted to grayscale. A 2D Discrete Cosine Transform (DCT) is applied to the resized image, producing a matrix of frequency coefficients. These coefficients encode the essential frequency characteristics of the image, enabling the generation of a compact and robust perceptual hash.

Wavelet hashing (wHash) goes further by leveraging wavelet transforms to decompose the image into multi-scale frequency bands, allowing it to capture both coarse and fine-grained visual details while maintaining resilience to blurring, lighting changes, and other natural variations. Together, these hashing methods enable a spectrum of content-based comparison strategies: from simple luminance-driven fingerprints to sophisticated frequency-domain encodings. When combined, they offer a powerful toolkit for detecting visual similarity, identifying tampering, and supporting resilient image verification across diverse real-world scenarios such as autonomous systems, digital forensics, and large-scale multimedia platforms.

Hamming Distance

Hamming Distance is traditionally used to measure the difference between two-character sequences. In the context of perceptual image similarity, it is employed to compare the hash values of image frames by determining how many positions differ between two or more equal-length binary strings. In essence, it counts the number of bit mismatches between the hashes. This metric is widely applied in error detection

and correction, information theory, and image comparison tasks[75,76]. When two corresponding bits are identical, the distance contributes a value of 0; when they differ, it contributes a value of 1. The threshold plays a crucial role in determining the accuracy of similarity detection within image alteration schemes. It defines the cutoff used to interpret Hamming Distance values and decides whether two data items should be considered similar[77].

Hamming Distance is a fundamental metric used to quantify the dissimilarity between two characters or binary sequences of equal length. Within the domain of perceptual image similarity, it provides a simple yet powerful way to compare the binary hash representations generated by algorithms, such as aHash, dHash, pHash, or wHash. By counting the number of bit positions at which the two hashes differ, Hamming Distance effectively measures how much the underlying visual content of two images diverges. A value of 0 indicates identical hashes, and thus high likelihood of identical or near-identical visual content, while larger values correspond to progressively greater differences in scene structure or appearance. Because each bit of mismatch contributes a value of 1, the metric is computationally lightweight and well suited for high-volume or real-time applications. Hamming Distance has deep roots in error detection, coding theory, and data integrity validation, making it a natural fit for tasks requiring robust and explainable similarity assessments. Crucially, its usefulness in image comparison depends on the selection of an appropriate threshold, which determines the maximum allowable distance for two hashes to be considered similar. Setting this cutoff too low may cause benign variations, such as compression, minor noise, or slight perspective changes, to be misinterpreted as differences, while setting it too high risks overlooking meaningful alterations or tampering. Thus, threshold calibration becomes an essential step for reliable similarity detection in visual integrity checking, anomaly detection, and tamper-resistant image processing systems.

Attack Vectors and Experimental Set up

The developed visual augamentors are intentionally repurposed as controlled visual attack vectors that simulated both benign environmental degradation and adversarial interference affecting image integrity in CAVs. Rather than assuming a single threat model, the augmentations span a continuum of attacks, from naturally occurring disturbances to aggressive conditions that can compromise perception pipelines.

Common corruption and photometric augmentations represent low-level signal-space attacks. Gaussian noise emulates sensor noise amplification, RF interference, or deliberate noise injection, while Gaussian and motion blur model defocus, platform vibration, or induced camera shake. Brightness, contrast, saturation, and grayscale transformations reflect exposure to manipulation, glare, white-balance drift, or international chromatic distortion. Compression artifacts and pixelation simulated bandwidth-constrained transmission, storage degradation, or adversarial down

[75] Cheng, S., Tang, Z., Zeng, S., Cui, X. and Li, T., 2024. Pfdup: Practical fuzzy deduplication for encrypted multimedia data. *Journal of Industrial Information Integration*, *40*, p.100613.

[76] Tang, Z., Zeng, S., Li, T., Cheng, S. and Zheng, H., 2023, October. Fuzzy deduplication scheme supporting pre-verification of label consistency. In *International Conference on Provable Security* (pp. 365-384). Cham: Springer Nature Switzerland.

[77] Ionescu, M. and Ralescu, A., 2004, July. Fuzzy hamming distance in a content-based image retrieval system. In *2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No. 04CH37542)* (Vol. 3, pp. 1721-1726). IEEE.

sampling, all of which may subtly alter pixel statistics without immediately raising alarms. Spatial operations, such as flips and crops, represent viewpoint perturbations or partial occlusions that stress geometric consistency assumptions.

The weather-like augmentations operate as higher-level, context-aware attack vectors. Fog, haze, rain, snow, and overcast conditions degrade visibility by reducing contrast, desaturating colors, and partially occluding scene elements. More extreme composites, such as heavy rain, sleet, or whiteout blizzards, deliberately collapse edge information and global scene structure, approximating worst-case operational environments. These effects are particularly relevant for UGV deployments, where perception systems must remain reliable under rapidly changing weather, low-light transitions, dust, or aerosol interference. Importantly, all attacks are parameterized across severity levels (s1-s5), enabling a systematic escalation from subtle perturbations to near-failure visual conditions.

By treating each augmentator as an attack primitive, the framework supports repeatable and interpretable evaluation of how visual integrity degrades under progressively harsher conditions, without relying on ad hoc or manually altered data.

The experimental setup evaluates visual integrity preservation and compromise detection by coupling the augmentation-based attack vectors with perceptual image hashing and Hamming Distance analysis. For each original image, a reference set of perceptual hashes is computed using aHash, pHash, dhash, and wHash. These hashes serve as compact visual fingerprints representing the expected, uncompromised visual content.

Each augmentator container processes the same base images across the configured severity levels, generating attacked image variants in an isolated and reproducible manner. For every augmented output, perceptual hashes are recomputed and directly compared against the reference hashes of the original image. The Hamming Distance between corresponding hashes quantifies the degree of visual divergence induced by the attack. Since perceptual hashes are designed to be invariant to benign transformations while remaining sensitive to structural or semantic changes, the resulting distance distributions provide a principled indicator of integrity degradation.

Severity-aware analysis is central to the setup. At low severities, most augmentations are expected to produce small Hamming Distances, reflecting acceptable visual variation that may trigger minor integrity alarms. As severity increases, distances are expected to grow monotonically, eventually crossing predefined thresholds that signal potential compromise, excessive degradation, or adversarial manipulation. This allows the identification of attack-specific breaking points, conditions under which perceptual similarity can no longer be reliably preserved.

The containerized workflow ensures full experimental isolation and scalability. Each augmentator runs as an independent Docker service, processes a controlled subset of images, logs severity parameters and runtime metadata, and exits deterministically after completion. This design allows parallel execution of attack families, straightforward ablation studies, and reproducible benchmarking across datasets and hash functions.

These attack vectors and experimental setup establish a closed-loop integrity evaluation framework. Augmentations emulate realistic and adversarial visual attacks,

while perceptual hashing combined with Hamming Distance acts as a lightweight detection and quantification mechanism. This pairing enables systematic assessment of how visual perception degrades, when integrity can still be trusted, and when downstream decision making in CAVs and UGVs becomes unreliable. As a result, the framework directly supports resilient AI-enabled perception by providing measurable, explainable, and severity-aware indicators of visual compromise under real-world and adversarial conditions.

3.6.5. Next steps

The current study establishes a lightweight and interpretable framework for visual integrity monitoring in CAV environments using perceptual hashing and Hamming Distance under both adversarial and environmental attack vectors. Building upon these findings, several research directions are identified for future work.

The first extension involves the adaptive calibration of detection thresholds. While fixed thresholds were sufficient to demonstrate the sensitivity of different hashing techniques, future work will explore dynamic, context-aware thresholds that account for attack type, severity progression, and environmental conditions. This would allow the system to distinguish more effectively between benign degradation and malicious manipulation, reducing false positives during adverse but legitimate operating conditions.

Second, the framework can be extended toward multi-hash fusion and confidence scoring. Instead of treating phash, aHash, dHash, and wHash independently, future work will investigate their combined use through weighted aggregation or learning-based fusion. Such an approach could provide a single integrity of confidence score, improving robustness against attack-specific weakness of individual hashing methods.

Third, the visual integrity indicators derived from Hamming Distance can be integrated into AI-driven decision-making and resilience mechanisms. Rather than performing binary accept-reject decisions, future systems may leverage integrity scores to trigger adaptive responses, such as sensor re-acquisition, cross-modal verification (e.g., LiDAR or radar), or graceful degradation of autonomy levels. This integrity aligns with safety-critical requirements in autonomous navigation.

Finally, future experimentation will extend the evaluation to temporal consistency and video streams, where successive frames can be jointly analyzed to detect gradual integrity drift or persistent tampering patterns. This direction is particularly relevant for real-time CAV deployments, where isolated frame-level decisions may be insufficient to capture long-term perception compromise.

4. Robustness Through Multi-X Context Awareness

Section 4 examines how robustness can be improved by exploiting Multi-X context awareness, i.e., by combining complementary sources of contextual information rather than relying on a single model output or a single sensing stream. In GuardAI, “Multi-X” spans multiple views (e.g., ground vs. birds-eye view perspectives), multiple modalities (e.g., RGB and infrared), multiple tasks (e.g., jointly checking outputs from different perception tasks), and multiple agents (e.g., cross-validating observations and confidence across a team). The underlying idea is that many failures (whether caused by harsh operational conditions, distribution shifts, or adversarial manipulation) tend to be less consistent across X-dimensions; leveraging these complementary cues enables detection of inconsistencies, graceful degradation, and more reliable downstream decisions.

Following this principle, the section first focuses on cross-view geo-localization (CVGL) for robust pose estimation, where matching ground views to satellite/drone imagery provides an alternative localization anchor when GNSS is degraded, spoofed, or otherwise unreliable, and extends this to UAV pose refinement under realistic navigation uncertainties. It then introduces context-awareness mechanisms that strengthen trustworthiness at runtime, including multi-task consistency checks for attack/anomaly indications and the infrared modality as a fallback for RGB-IR sensing when the visible channel becomes unreliable. Finally, the section transitions to multi-agent context awareness, presenting a methodology for perception effectiveness and action scoring that incorporates visual integrity assessment (e.g., similarity/perceptual-hash based checks) and aggregates agent-level evidence into event-level and overall situational awareness scores—supporting coordinated, resilient operation even when some agents or sensing streams are partially compromised.

4.1. Robust pose estimation via Cross-view Geo-localization (CVGL)

Cross-View Geo-Localization (CVGL) provides a robust solution for estimating the absolute geographic pose of autonomous systems by matching Ground-View or Aerial-View Camera Images to Geo-referenced Satellite Imagery. Specifically, CVGL leverages visual invariants and distinctive features, enabling it to maintain robust functionality in challenging scenarios, such as high localization error in GNSS measurements, or spoofing GNSS data. This capability fundamentally contrasts with traditional GNSS localization, which is highly susceptible to signal degradation, multipath effects, and vulnerabilities stemming from radio-frequency interference and spoofing attacks. In UAV applications, where precise localization is critical, CVGL complements existing systems like Visual-Inertial Odometry (VIO) by providing absolute position updates that correct drift and detect GNSS anomalies. The CVGL framework enables the system robustness by fusing visual cues with noisy or compromised GNSS data, reducing position uncertainty from 5–10 meters to sub-meter levels. As a result, CVGL operates as both a redundancy layer and a resilience mechanism, enabling reliable navigation in dynamic, degraded, or adversarial environments.

4.1.1. Ground View and Satellite Images for Robust Cross-View Geo-Localization

Pose estimation-based Cross-View Geo-Localization (CVGL) approaches estimate the agent's precise 3-Degrees of Freedom (DoF) pose (lateral, longitude and yaw) by refining an initial coarse pose typically obtained from GNSS or ground-view camera images. Unlike CVGL retrieval-based approaches that identify the closest match from a database, pose estimation methods to operate in a continuous pose space, allowing for fine-grained localization. This enables significantly higher accuracy, often achieving sub-meter or even centimeter-level precision, which is critical for autonomous vehicle applications where spatial awareness is essential.

The Cross-View Geo-Localization (CVGL) framework using Satellite and Ground-View Images typically utilizes transformer-based architectures, diffusion models, and foundation models or encompasses diverse set of pipelines to enhance localization accuracy, robustness and efficiency. Recent developments such as SAIG-D[78] and GeoDTR[79] leverage diffusion models and geometry-guided transformers, respectively, to improve cross-view matching through realistic view synthesis and learnable ground-to-satellite transformations. Sample4Geo[80] expands on this by generating multiple plausible satellite images to increase robustness under visual ambiguity. For autonomous systems, foundation models such as BEV-integrated localization[81] and temporal filtering[82] improve spatial understanding and stability in dynamic environments. Accordingly, the CVGL framework introduced by Wang et.al [83] is employed to facilitate precise pose estimation. It operates with flexible numbers of onboard cameras and demonstrates strong generalization to environmental changes requiring only geo-poses as ground truth without needing dense annotations. Furthermore, by explicitly identifying and filtering out moving objects and temporary structures, it maintains accuracy even in busy urban environments with significant traffic and pedestrian activity.

The Cross-View Geo-Localization scheme[84] is a fine-grained self-localization method in lateral, longitude and yaw using multiple onboard cameras and satellite imagery, leveraging vision data to refine a noisy initial pose estimate. The system considers GNSS-tagged satellite images and an initial coarse pose estimation (typically from raw GNSS/Odometry data) as inputs, and outputs a highly accurate refined pose specified

---

[78] Zhu, Y., Chen, S., Lu, X. and Chen, J., 2023. Cross-view image synthesis from a single image with progressive parallel GAN. *IEEE Transactions on Geoscience and Remote Sensing*, *61*, pp.1-13.
[79] Wang, H., Zhang, L., & Liu, M. (2024). BEV-guided cross-view geo-localization for autonomous driving. *IROS*.
[80] Deuser, F., Habel, K. and Oswald, N., 2023. Sample4geo: Hard negative sampling for cross-view geo-localisation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 16847-16856).
[81] Ye, J., He, J., Li, W., Lv, Z., Lin, Y., Yu, J., Yang, H. and He, C., 2025. Leveraging BEV paradigm for ground-to-aerial image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 28451-28461).
[82] Wang, S., Nguyen, C., Liu, J., Zhang, Y., Muthu, S., Maken, F.A., Zhang, K. and Li, H., 2024. View from above: Orthogonal-view aware cross-view localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 14843-14852).
[83] Wang, S., Zhang, Y., Perincherry, A., Vora, A. and Li, H., 2023. View consistent purification for accurate cross-view localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 8197-8206).
[84] Wang, Shan, et al. "View consistent purification for accurate cross-view localization." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023.

by its global position x, y and orientation yaw. This refined pose improves upon the coarse estimate, enabling sub-meter localization accuracy even in GPS-degraded or adversarial environments.

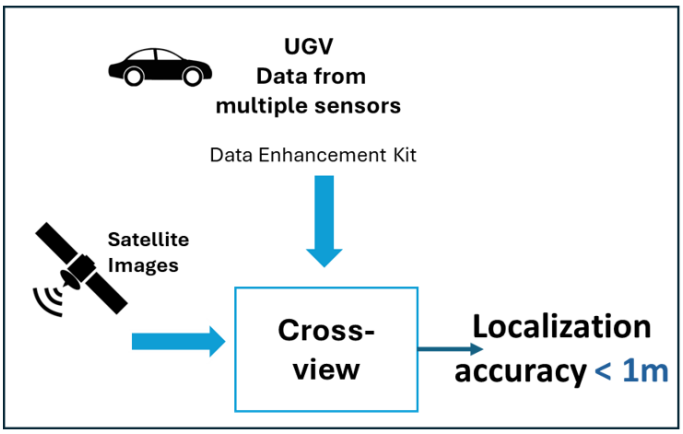


Figure 4.1: Cross-View Geo-Localization framework through Satellite and Ground-View Images achieving high localization accuracy.

The method consists of the Spatial-Aware Feature Confidence Extractor (SAFCE) outputting feature maps (F) containing visual descriptors, View-consistent confidence maps (V) indicating which features are geometrically consistent across views and On-ground confidence maps (O) identifying features that correspond to ground-level structures rather than elevated objects. Additionally, it consists of the View-consistent On-ground Key-point Detection (VOKD) module to fuse the confidence maps to identify the most reliable features. It selects top-k confident features from ground-view images and search their corresponding features in satellite maps, effectively filtering out unreliable matches. Finally, it designs the Residual-based Pose Refinement Block (RPRB) module for iterative pose optimization to refine the vehicle's position and orientation estimate using the selected confident features and their weights.

Due to the incorporation of a spatial embedding approach, this framework leverages camera intrinsic and extrinsic information to reduce matching ambiguity. By encoding geometric constraints from camera calibration parameters, ambiguous visual matches are addressed which is critical under overlapping field-of-view coverage from multiple cameras.

Experiments have been performed on the Ford Dataset[85] to evaluate the localization accuracy of the pipeline[83]. This dataset covers one full year of recordings (2017-2018) and consists of strong seasonal variations such as summer, fall, winter (snow), cloudy conditions. Furthermore, it includes diverse driving environments along a 66 km route in Michigan on highways, city centers, campus roads, airports, and suburban roads. The dataset is time-synchronized with high-resolution raw sensor data in ROS bag format. The included sensors are LiDAR, undistorted camera images, IMU accelerations with angular velocities, Global Positioning System (GPS) data, and time. The dataset further provides necessary reference information, including the raw sensor

[85] Agarwal, S., Vora, A., Pandey, G., Williams, W., Kourous, H. and McBride, J., 2020. Ford multi-AV seasonal dataset. *The International Journal of Robotics Research*, *39*(12), pp.1367-1376.

streams, prior localized positions, and ground-truth poses required for objective performance assessment. Each route in the dataset is described by the recording date, the vehicle and the route such as 2017-08-04-V2-Log4. To assess system robustness and sensitivity to environmental shifts, two distinct sequences representing significantly different driving conditions were analyzed for localization accuracy and the study was performed applying uniform noise of near-zero variance.

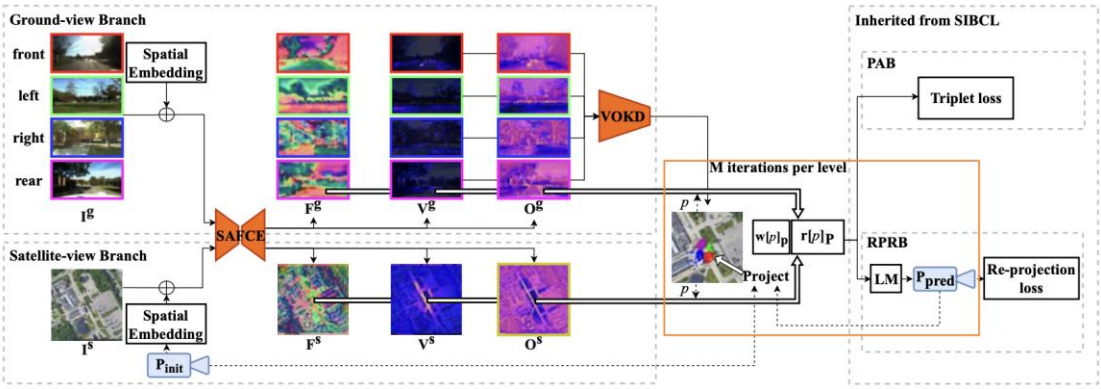


Figure 4.2: Architectural Diagram of the CVGL framework[84]

The evaluation of this pipeline was conducted using the metrics of Normalized Error, Translation Error, and Precision-Recall metric. Specifically, the normalized error metric measures the percentage reduction of errors (translational, latitudinal, and longitudinal) compared to their initial values:

$$\Delta error = \frac{initial - final}{initial} \times 100 \tag{16}$$

The translational error is computed via the L2 norm of the latitude and longitude deviations with respect to the ground-truth pose:

$$t = \sqrt{(\Delta lat)^2 + (\Delta lon)^2} \tag{17}$$

And finally, the Precision-Recall metric evaluates the fraction of predictions that fall within a specified localization threshold δ. Thresholds are defined at 0.5m, 1m, 2m, 5 m, enabling assessment of both coarse and fine-grained localization:

$$Precision(\delta) = \frac{\text{Number of predictions with error } \leq \delta}{\text{Total number of predictions}} \tag{18}$$

Firstly, the 2017-08-04-V2-Log4 route sequence was used for training and the 2017-10-26-V2-Log4 for evaluation studying the affection of the initial pose error to the precision as well as the times of iterations in the optimization problem to refine pose. Figure 4.3 describes the reduction in the translation, longitude and latitude errors under different number of initial errors and number of iterations. With moderate initial offsets, the optimizer reliably lowers the translation error often by more than 50%, while with the initial error up to 15, the optimizer often fails to reliably reach the correct alignment, so the final position estimate still has a large error. Indicative results showcase that for initial error 5m, the method reduces the final error by 71%, in 8m by 66%, 10m by 50%, while for 15m the error is reduced to 20%. Figure 4.4 reports the Precision–Recall

values at different distance thresholds (0.5 m, 1 m, 2 m, and 5 m) with varying the initial error across values {2,4,5,8,10,15,20,30} meters, and varying the number of optimizer iterations across {2,5,10,20,40,60}. As illustrated, the system is highly sensitive to the small initial errors with the optimizer reducing the localization error, although in higher initial errors the performance degrades. Moreover, Figure 4.5 illustrates the Cumulative Distribution Functions of localization accuracy across the testing sequence and indicates that a majority of estimates are within 1–2 m of the ground truth, while under poor initialization the error distribution shifts and develops a long tail toward much larger errors.

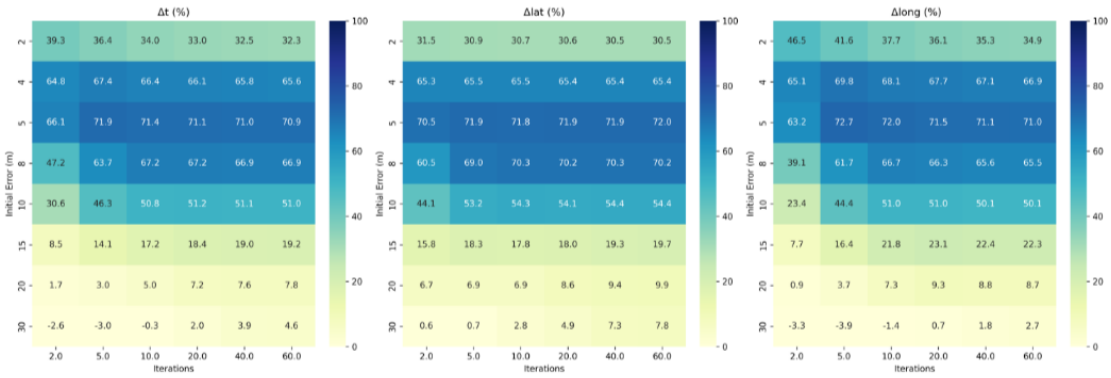


Figure 4.3: Translational Error Reduction across various initial errors and optimizer's iterations.

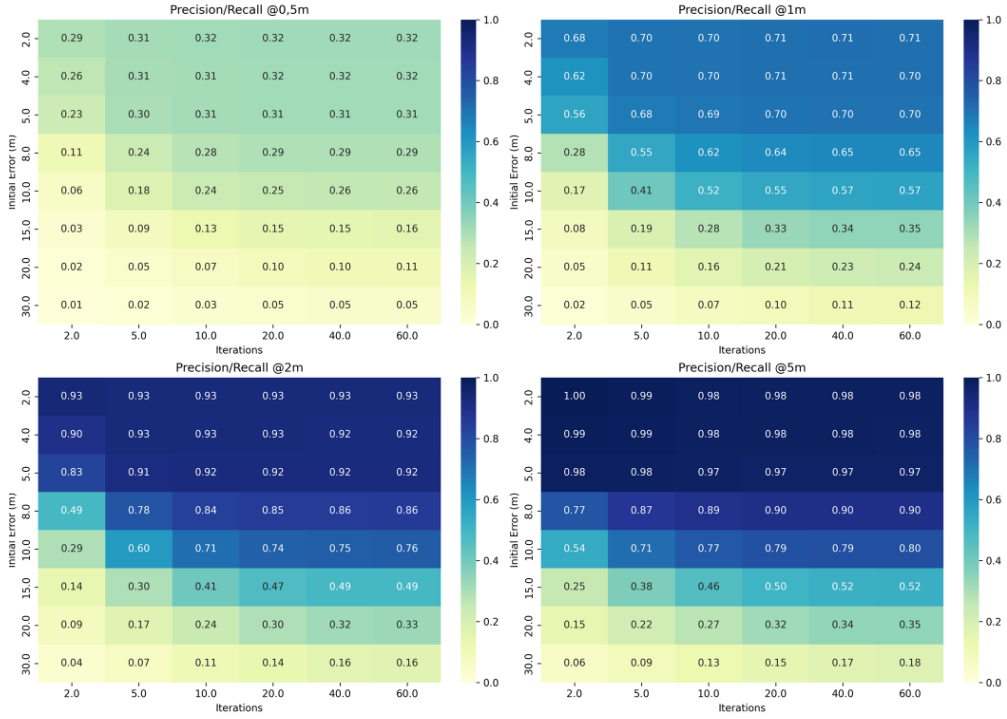


Figure 4.4: Precision/Recall Heatmap plots of @0.5m, @1m, @2m and @5m demonstrating the number of optimizer's iterations along with the initial error (m).

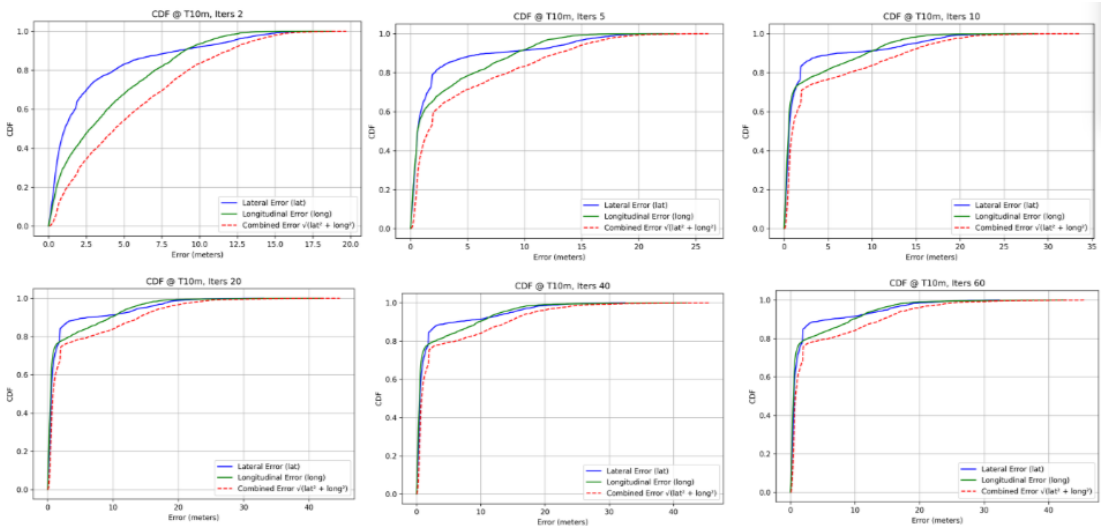


Figure 4.5: Cumulative Distribution Functions (CDF) of final translation errors.

Moreover, another study was performed using the 2017-08-04-V2-Log4 route sequence for training and the 2017-10-26-V2-Log5 for evaluation analyzing the system performance under different environments. Figure 4.6 describes the reduction in the translation, longitude and latitude errors under different number of initial errors and number of iterations. In case of small initial errors, the optimizer achieves high localization accuracy, although the improvements are less than in the first study. Figure 4.6 presents the percentage of the error reduction in the Iterations-Initial Error Plot, indicating that for 5m initial error, the final pose error is reduced by 49.9% and for 15m by 19.5% after 5 iterations. Hence, the high initial errors over 15 meters demonstrate high final pose estimation errors and unreliable convergence. Figure 4.7 reports the Precision–Recall values at different distance thresholds (0.5m, 1m, 2m, and 5m) with varying the initial error across values {2,4,5,8,10,15,20,30} meters, and varying the number of optimizer iterations across {2,5,10,20,40,60}. The system achieves high precision at coarse thresholds (2 m and 5 m) but recall at smaller thresholds (0.5m and 1m) is reduced. Even with more optimization steps, the method rarely recovers from large initial offsets, highlighting strong sensitivity to environmental variation.

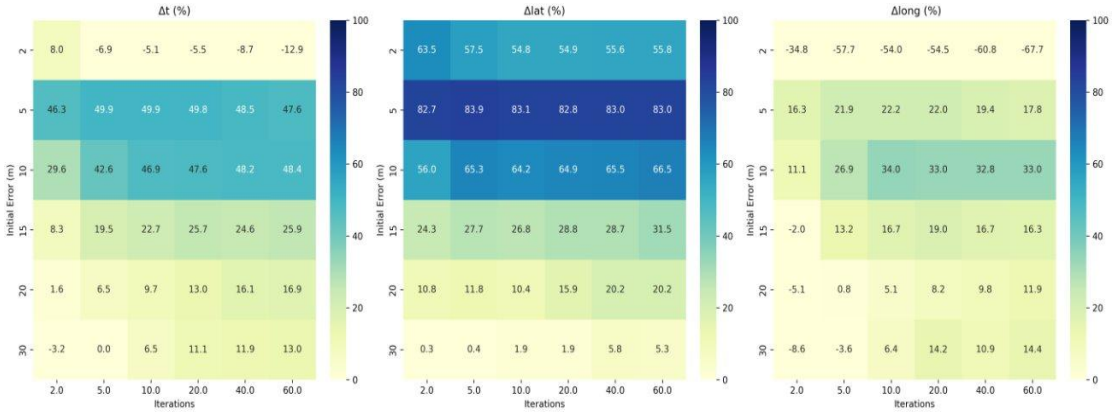


Figure 4.6: Translational Error Reduction across various initial errors and optimizer's iterations.

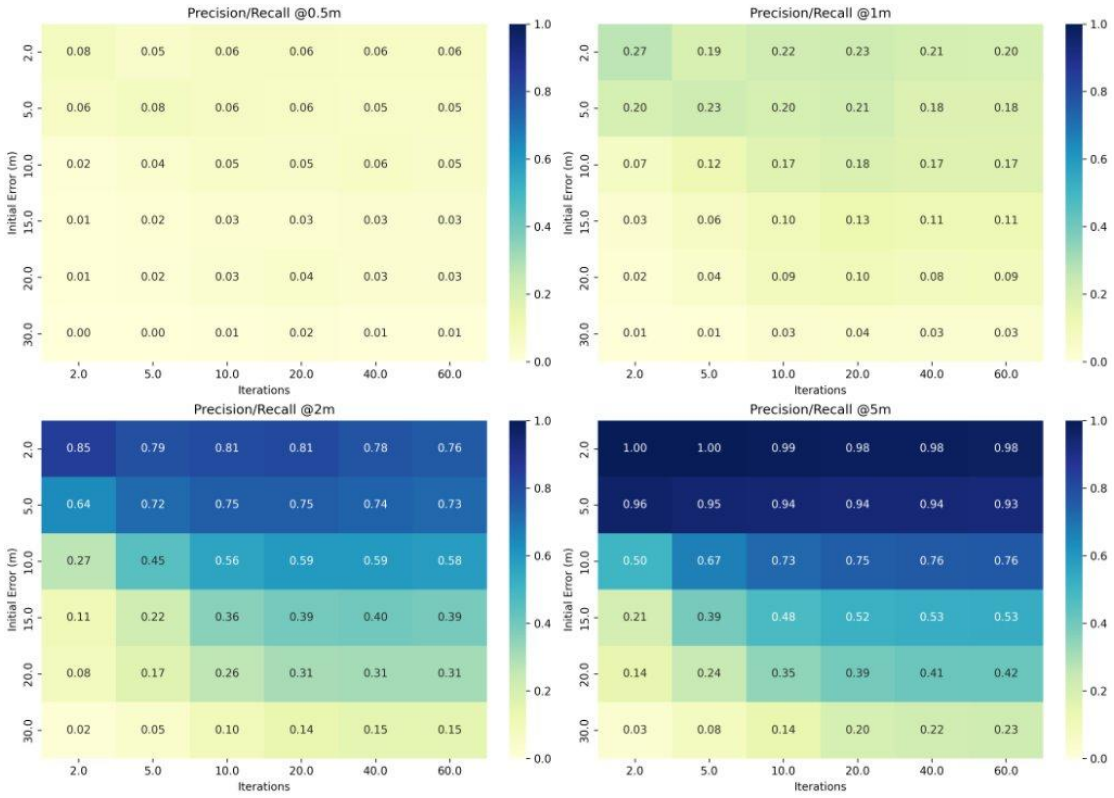


Figure 4.7: Precision-Recall Heatmap plots of @0.5m, @1m, @2m and @5m demonstrating the number of optimizer's iterations along with the initial error (m)

Overall, this method lacks generalization of environmental variations. Introducing a depth-estimation technique such as Lift-Splat-Shoot (LSS)[86] can improve the localization by relaxing the overly strict flat-ground and zero-lift assumptions through explicit, geometrically consistent modeling of the ground plane and camera height ("lift"). While the current framework implicitly assumes that all correspondences lie on a perfectly planar ground at a fixed heigh, making it sensitive to elevation errors and camera pitch, the LSS approach incorporates the plane-induced homography and lift parameters into the loss, allowing small deviations from ideal flatness and better alignment under realistic camera mounting and terrain variations. By grounding the loss in a more physically accurate model, it is also expected to improve generalization across different scenes, camera setups, and road profiles not seen during training. This approach will be designed for the next phase and reduce the localization estimation error.

4.1.2. Cross-View Geo-Localization for UAV Pose Refinement

Unmanned Aerial Vehicles (UAVs) increasingly rely on Global Navigation Satellite Systems (GNSS) for localization and navigation in safety-critical applications. However, GNSS-based positioning is inherently limited in accuracy and reliability. Under nominal conditions, consumer-grade GNSS receivers exhibit positioning errors

[86] Lu, Shu-Wei, Yi-Hsuan Tsai, and Yi-Ting Chen. "Toward Real-world BEV Perception: Depth Uncertainty Estimation via Gaussian Splatting." *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025.

in the range of 2-5 meters due to atmospheric delays, clock errors, and orbital uncertainties. In challenging environments, these errors escalate significantly.

Multipath propagation, where satellite signals reflect off buildings, terrain, or other surfaces before reaching the receiver, is a primary source of degradation in urban and semi-urban environments. Multipath-induced errors can reach 5-10 meters or more, depending on the geometry of surrounding structures and the satellite constellation visibility. This level of uncertainty is unacceptable for applications requiring precise trajectory following, automated landing, or coordinated multi-agent operations.

Beyond unintentional degradation, GNSS systems are vulnerable to deliberate attacks. GNSS spoofing involves broadcasting counterfeit satellite signals that induce controlled position errors in the target receiver, potentially displacing the reported position by 5-15 meters or more without triggering standard integrity checks[87]. Such attacks pose severe threats to autonomous UAV operations, as they can cause vehicles to deviate from intended flight paths, enter restricted airspace, or collide with obstacles. The accessibility of software-defined radio platforms has made spoofing attacks increasingly feasible, elevating the urgency of developing resilient localization methods.

Cross-View Geo-Localization (CVGL) offers a solution by matching images captured from UAV-mounted cameras against geo-referenced satellite imagery. Unlike GNSS, visual localization is immune to radio-frequency interference and spoofing attacks, providing an independent position estimate that can be used to detect anomalies in GNSS measurements or serve as a fallback during signal denial. By leveraging the abundance of publicly available satellite imagery, CVGL can provide absolute position corrections without requiring pre-mapped environments or suffering from the drift inherent to visual-inertial odometry systems.

Within the GuardAI framework, CVGL serves as a context-aware mechanism to enhance the robustness of UAV pose estimation. The objective is to refine noisy or potentially compromised GNSS estimates using visual evidence, reducing position uncertainty from the typical 5-10 meter range to sub-meter levels where matching conditions permit. This capability contributes to the overall resilience of edge AI systems operating on autonomous platforms in contested or degraded environments.

Existing approaches can be broadly categorized into two main paradigms: Visual-Inertial Odometry (VIO) systems and image retrieval-based localization methods.

Visual-Inertial Navigation Systems such as VINS-Fusion[88],[89] have demonstrated robust performance for relative pose estimation by tightly coupling camera and IMU measurements through optimization-based approaches. VINS-Fusion achieves accurate self-localization by performing sliding window optimization over visual features and pre-integrated IMU measurements. However, VIO methods inherently

[87] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. O'Hanlon, and P. M. Kintner, "Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer," Proceedings of the ION GNSS Conference, 2008

[88] Tong Qin et al., “A General Optimization-based Framework for Local Odometry Estimation with Multiple Sensors”, arXiv:1901.03638, 2019.

[89] Tong Qin et al., “A General Optimization-based Framework for Global Pose Estimation with Multiple Sensors”, arXiv:1901.03642, 2019.

suffer from drift accumulation over time, as they estimate relative motion rather than absolute position. This limitation motivates the need for complementary absolute localization mechanisms such as CVGL.

Image retrieval-based geo-localization methods have evolved considerably with the advent of deep learning. Early works focused on ground-to-satellite matching using learned feature representations[90], where the goal is to identify the correct satellite image from a database given a ground-level query. While these approaches achieve high recall rates on benchmark datasets such as CVUSA and University-1652, they are primarily designed for coarse place recognition rather than precise metric localization.

Recent advances in dense feature matching have enabled more precise localization. The MASt3R framework[91] introduced a 3D-grounded approach to image matching by treating matching as a 3D reconstruction task, it achieves superior performance on challenging cross-view scenarios compared to traditional 2D feature matchers. This capability is particularly relevant for drone-to-satellite matching, where viewpoint discrepancies and scale variations are substantial.

Framework Description

The proposed CVGL pipeline integrates dense feature matching with geometric pose optimization to refine noisy GNSS estimates using satellite imagery. The framework operates in several stages, as illustrated in Figure 4.8.

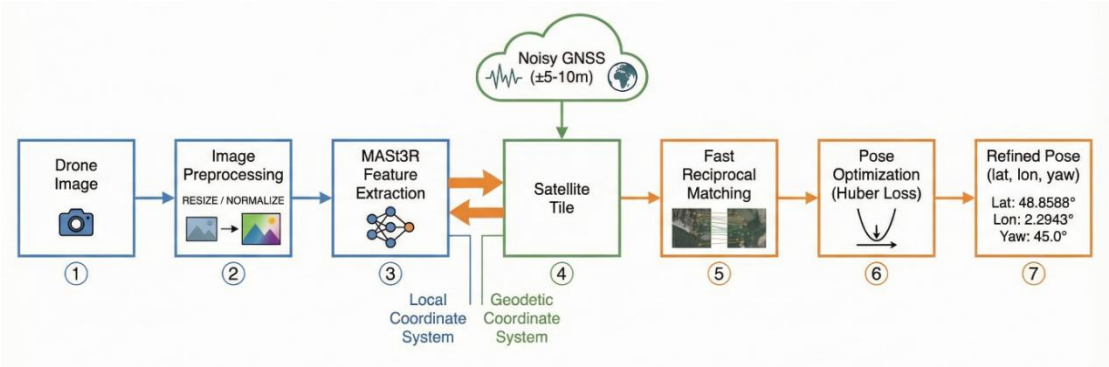


Figure 4.8: Overview of the Cross-View Geo-Localization pipeline for UAV pose refinement. The system takes as input a drone image, noisy GNSS coordinates, and IMU-derived yaw estimate. A satellite tile is retrieved from a mapping service based on the approximate GNSS position. Dense feature matching using MASt3R establishes correspondences between the drone and satellite views. Finally, a bounded least-squares optimization refines the pose estimate by minimizing reprojection residuals.

Input Processing

Given a drone-captured image and an initial (noisy) GNSS position estimate, the system first retrieves a corresponding satellite tile centered on the approximate location. The satellite imagery is obtained through web mapping services at a fixed

[90] L. Zheng, Y. Zheng, M. Cai, and R. Wei, "University-1652: A Multi-view Multi-source Benchmark for Drone-based Geo-localization," Proceedings of the 28th ACM International Conference on Multimedia, pp. 1395-1403, 2020.

[91] V. Leroy, Y. Cabon, and J. Revaud, "Grounding Image Matching in 3D with MASt3R," European Conference on Computer Vision (ECCV), 2024.

zoom level, providing a ground sample distance (GSD). The tile dimensions are chosen to ensure that the true position lies within the retrieved imagery despite GNSS errors of up to 10-15 meters. The drone image is resized to a compatible resolution while adjusting the camera's intrinsic matrix accordingly to preserve geometric consistency.

Dense Feature Matching

The core of the pipeline employs MASt3R, a state-of-the-art dense matching network based on Vision Transformers and it produces dense local descriptors that enable robust matching even between images with significant appearance and viewpoint differences. The network processes both the drone and satellite images simultaneously, outputting dense descriptor maps and confidence scores for each view. Correspondences are established using fast reciprocal nearest-neighbor matching in the descriptor space. This approach ensures that matches are mutually consistent, such as the drone pixel is matched to a satellite pixel only if that satellite pixel also selects the same drone pixel as its nearest neighbor. This reciprocity constraint significantly reduces false matches arising from repetitive textures common in aerial imagery.

Geometric Pose Optimization

The final stage formulates pose refinement as a bounded nonlinear least-squares problem. Given the matched pixel correspondences between drone and satellite images, we optimize for corrections to the initial pose estimate (yaw angle and planar translation) that minimize the reprojection residuals.

The transformation from drone image coordinates to satellite image coordinates is modelled as an affine mapping that accounts for the relative scale difference (determined by the ratio of ground sample distances), rotation (drone heading), and translation. For a drone pixel coordinate (u_d, v_d), the predicted satellite coordinate (u_s, v_s) is given by (19):

$$p_s = \tilde{s} \cdot R(\psi) \cdot A \cdot (p_d - c_d) + c_s + t \quad (19)$$

where R(ψ) is the 2D rotation matrix for heading angle ψ, A captures the GSD ratio between drone and satellite views, c_d and c_s are the image centers, t is the translation in pixels, and s̃ is an empirical scale factor.

The optimization minimizes the sum of squared residuals between predicted and observed satellite pixel locations. To handle outliers from mismatches, we employ a Huber loss function which provides robustness to gross errors while maintaining efficiency near the optimum.

Dataset

Experimental evaluation was conducted using a drone imagery dataset provided by the KIOS Research and Innovation Center of Excellence at the University of Cyprus (UCY) in Autumn 2025. The dataset comprises UAV images captured with a DJI platform equipped with RTK-GNSS for ground-truth positioning and calibrated camera intrinsics.

Each image in the dataset is accompanied by metadata including precise RTK-derived coordinates (latitude, longitude, altitude), body and gimbal yaw angles, and timestamps. The flight trajectories cover university campus environments with diverse

features, including buildings, roads, vegetation, and open areas. We are in ongoing communication with UCY for further dataset expansions to include additional environments and flight conditions.

To simulate realistic GNSS degradation scenarios representative of multipath or spoofing conditions, we inject artificial noise into the ground-truth positions following the PureACL protocol: uniform random perturbations of ±15° in yaw and ±10 meters in horizontal position. This noise model reflects the magnitude of errors observed in real-world GNSS-challenged environments and provides a controlled setting to evaluate the CVGL system's correction capabilities.

Preliminary Results

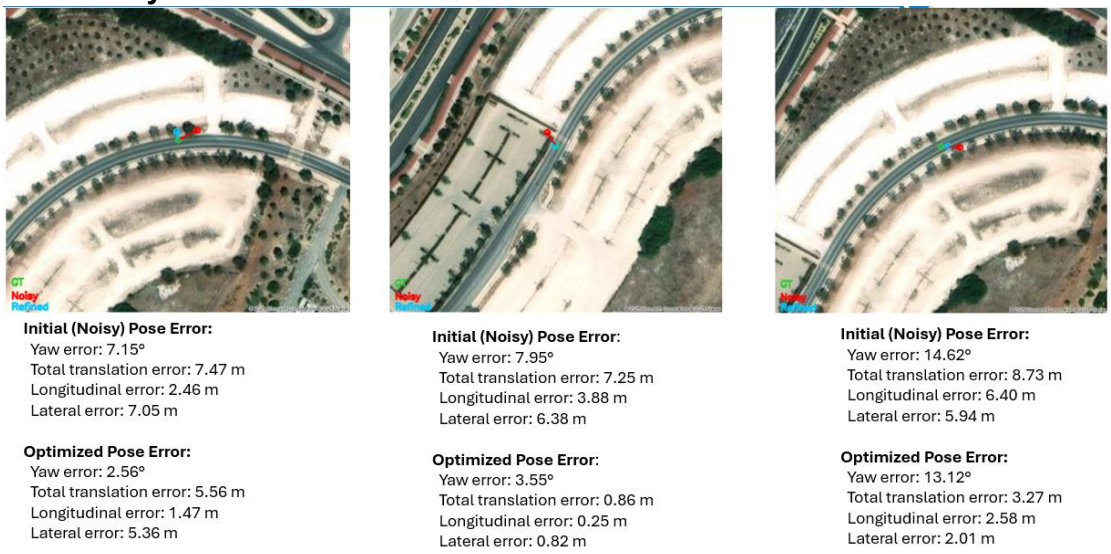


Figure 4.9: Cross-View Geo-Localization results on three test cases from the UCY drone dataset. Each panel shows the satellite reference tile with marked positions: noisy GNSS estimate (red), ground truth (blue), and refined pose (green). Below each image, quantitative errors are reported for both the initial noisy pose and the optimized result.

The results demonstrate consistent improvement across all test cases (as illustrated in Figure 4.9). In the first example (left panel), the initial noisy GNSS position exhibited a translation error of 7.47 meters and yaw error of 7.15°. After optimization, these were reduced to 5.56 meters and 2.56° respectively. The second case (center panel) demonstrates more substantial improvement, with translation error decreasing from 7.25 meters to 0.86 meters, achieving sub-meter accuracy typical of RTK-GNSS systems but obtained purely through visual means.

The third case (right panel) illustrates the system's behavior under larger initial errors (8.73 meters translation, 14.62° yaw). While the optimization successfully reduces the translation error to 3.27 meters, the yaw correction is more limited (13.12° residual error), suggesting the presence of challenging matching conditions or local minima in the optimization landscape.

Across the evaluated samples, the average translation error reduction was approximately 50%, with particularly strong performance in longitudinal positioning (along the flight direction). These preliminary results validate the feasibility of the proposed approach for mitigating the 5-10 meter positioning errors characteristic of

multipath-affected or spoofed GNSS signals, enhancing UAV localization robustness within the GuardAI framework.

4.1.3. Next steps

The method for ground-view and satellite image based cross-view localization that was presented in Section 4.1.1 lacks generalization of environmental variations. Introducing a depth-estimation technique such as Lift-Splat-Shoot (LSS) can improve the localization by relaxing the overly strict flat-ground and zero-lift assumptions through explicit, geometrically consistent modeling of the ground plane and camera height ("lift"). While the current framework implicitly assumes that all correspondences lie on a perfectly planar ground at a fixed heigh, making it sensitive to elevation errors and camera pitch, the LSS approach incorporates the plane-induced homography and lift parameters into the loss, allowing small deviations from ideal flatness and better alignment under realistic camera mounting and terrain variations. By grounding the loss in a more physically accurate model, it is also expected to improve generalization across different scenes, camera setups, and road profiles not seen during training. This approach will be designed for the next phase and reduce the localization estimation error.

When BEV images from drones are used along with satellite images (see Section 4.1.2), several extensions can further enhance the framework's accuracy and robustness. Incorporating temporal information through particle filtering would enable sequential pose tracking across frames, maintaining a probabilistic distribution over UAV positions that evolves with new observations. This would mitigate isolated matching failures and provide smoother trajectory estimates by leveraging motion constraints. The refined CVGL pose estimates should be directly integrated as absolute measurements into tightly coupled SLAM frameworks such as VINS-Fusion, where visual-inertial odometry provides high-frequency relative motion while CVGL periodically corrects drift accumulation. Finally, the optimization stage can benefit from multi-scale coarse-to-fine refinement strategies and adaptive robust loss functions tailored to cross-view matching error distributions. Incorporating altitude constraints from barometric sensors and exploring learned uncertainty estimates for match confidence weighting may yield further improvements in challenging matching scenarios.

4.2. Multi-task Consistency Checks for Attack Detection

Modern visual perception systems increasingly rely on multi-task architectures, where multiple computer vision tasks are executed in parallel on the same sensory input. Typical examples include object detection and instance or semantic segmentation, which provide complementary information about scene structure. Such architectures are commonly implemented using shared-backbone designs, as exemplified by Mask R-CNN[92]. While this paradigm improves performance and robustness under normal conditions, it also introduces new vulnerabilities in adversarial settings.

Most adversarial robustness studies focus on evaluating performance degradation within individual tasks, commonly using task-specific metrics such as mean Average

[92] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. Proceedings of the IEEE International Conference on Computer Vision (ICCV).

Precision (mAP) for object detection or segmentation[93,94]. However, such metrics do not capture whether the internal semantic coherence between different perception task is preserved. In safety-critical deployment, adversarial perturbations may lead to inconsistent interpretations of the same scene across tasks, even when individual task outputs appear plausible in isolation.

4.2.1. Multi-task Consistency as an Adversarial Signal

Recent research has shown that adversarial perturbations can disrupt the consistency between related vision tasks, motivating the use of multi-task consistency as complementary signal for adversarial attack detection[95]. In particular, object detection and instance segmentation provide closely related but independently learned representations of scene structure. Under normal conditions, bounding boxes predicted by an object detector are expected to align spatially and semantically with instance masks produced by a segmentation model. This alignment arises naturally from shared visual representations and correlated task objectives. Adversarial perturbations can break this alignment, leading to measurable discrepancies between task outputs even when task-level performance degradation is limited.

4.2.2. Consistency Measurement between Detection and Segmentation

Multi-task consistency checks explicitly quantify the semantic agreement between detection and segmentation predictions. For each detected object, the spatial region defined by its bounding box is examined within the corresponding segmentation output. If the dominant semantic label inside the bounding box matches the class predicted by the object detector, the prediction is considered consistent; otherwise, it is marked as inconsistent. By aggregating consistency of information across all detected objects in an image, an image-level multi-task consistency score is obtained. This score reflects the degree of semantic agreement between the outputs of different tasks operating on the same input. Clean inputs have typically high consistency scores, indicating coherent interpretations across tasks. In contrast, adversarial inputs often produce significantly lower consistency scores, revealing disrupted cross-task agreements that may not be captured by conventional task-level metrics.

4.2.3. Adversarial Scenarios and Evaluation Protocol

The effectiveness of multi-task consistency checks is evaluated under an adversarial threat model using gradient-inspired scored-based perturbations. Both single-step and iterative attacks are considered, and multiple perturbation strengths are evaluated to assess robustness across a wide range of adversarial intensities. Adversarial perturbations are generated by optimizing the loss of a single task while being applied to the shared input image. This allows the analysis of how task-specific adversarial manipulations propagate across tasks and affect cross-task agreements. The evaluation is performed on matched clean-adversarial image pairs to ensure consistency and comparability. Detection performance is assessed by comparing the

[93] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). *Explaining and Harnessing Adversarial Examples*. International Conference on Learning Representations (ICLR).
[94] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). *Towards Deep Learning Models Resistant to Adversarial Attacks*. International Conference on Learning Representations (ICLR).
[95] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). *Mask R-CNN*. Proceedings of the IEEE International Conference on Computer Vision (ICCV).

distributions of consistency scores for clean and adversarial inputs. Threshold-independent metrics such as the AUC-ROC are used to quantify the effectiveness of consistency-based detection across different attack configurations.

4.2.4. Key Observations and Relevance

Experimental results demonstrate that multi-task consistency is a sensitive indicator of adversarial perturbations. In many cases, adversarial inputs lead to a significant drop in consistency scores even when task-level accuracy metrics, such as mAP remain relatively high. This highlights the complementary nature of consistency-based analysis compared to traditional robustness evaluation approaches. Overall, multi-task consistency checks provide a lightweight and model-agnostic mechanism for detecting adversarial perturbations in multi-task perception systems. The approach does not require retraining, architectural modifications, or additional supervision, making it suitable for integration into existing perception pipelines and safety-critical applications.

4.2.5. Next Steps

Several important directions remain open for future investigation. First, extending the proposed consistency analysis to additional perception tasks, such as depth estimation or optical flow, could further enhance detection robustness in more complex multi-task perception systems. Second, the incorporation of adaptive or learned consistency thresholds may help account for scene-dependent variability and improve reliability across diverse environmental conditions. Third, and most critically, future work should consider adaptive adversarial threat models in which the attacker explicitly optimizes perturbations to preserve cross-task agreement while degrading task-level performance. Evaluating multi-task consistency under such adaptive attacks would provide deeper insight into the fundamental limits of consistency-based detection and help identify potential failure modes.

4.3. Infrared Modality as a Fallback for RGB-IR Sensors

4.3.1. Introduction

Modern autonomous systems increasingly rely on visual perception for critical decision-making tasks ranging from autonomous navigation to surveillance and threat detection. However, single-modality vision systems present a fundamental vulnerability: adversarial perturbations, sensor degradation, or environmental conditions (fog, low-light, glare) that compromise one sensing modality can catastrophically degrade system performance. Multimodal sensing architectures, particularly RGB-infrared fusion, offer a promising defensive strategy by exploiting the orthogonal failure modes of complementary sensors. While visible-spectrum cameras excel in texture-rich, well-lit environments, infrared sensors maintain functionality under low-light conditions and demonstrate differential resilience to certain adversarial perturbations. The strategic value of such sensor diversity lies not merely in redundancy, but in intelligent fallback mechanisms that dynamically prioritize the more reliable modality under adversarial conditions. This defensive paradigm transforms multimodal perception from a performance enhancement into a security primitive, enabling graceful degradation rather than catastrophic failure when individual sensing channels are compromised. Understanding the robustness characteristics of

multimodal fusion under adversarial conditions is therefore critical for deploying trustworthy autonomous systems in contested or unpredictable environments.

4.3.2. Multimodal Defense for Object Detection

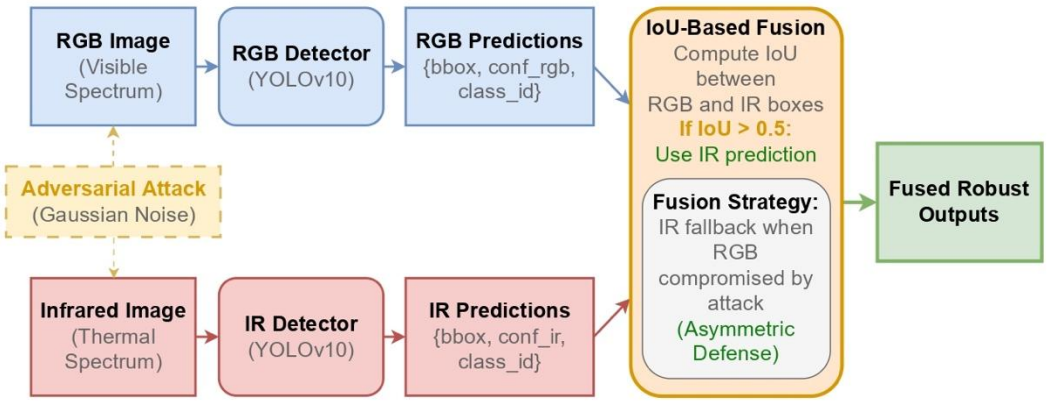


Figure 4.10: The overall pipeline of the infrared fallback mechanism when multi-modal object detection systems are compromised under adversarial conditions.

This work presents a defense framework (shown in Figure 4.10) designed to evaluate the resilience of multimodal object detection systems under adversarial conditions, with a particular focus on RGB-infrared sensor fusion as a defensive strategy. The system implements two primary attack methodologies: Gaussian noise perturbation and Cross-Modal Patch Attack (CMPA), both configured through a modular attack configuration system that enables dataset-specific and modality-specific parameterization. The defense strategy is to fall back to the infrared modality, by means of fusing predictions, when adversarial attacks affect both the visible and infrared sensors. The fusion is carried out by an IoU theshold: when the IoU between predicted bounding boxes from both modalities overlap by more than the threshold, the RGB predictions are retained. On the other hand, when there is not enough overlap, bounding boxes from both modalities are retained.

The Gaussian noise attack employs additive stochastic perturbations with configurable blending coefficients (α for visible, $\alpha_{inf}$for infrared) to simulate sensor degradation. The implementation supports both RGB and grayscale noise patterns, with the latter specifically tailored for infrared imagery. Critically, the framework recognizes differential vulnerability across modalities; for a given noise intensity, the visible imagery achieves higher detection degradation than the infrared imagery, reflecting the inherent robustness characteristics of thermal sensors.

The CMPA implementation represents a more sophisticated approach, utilizing differential evolution optimization to generate adversarial patches that simultaneously degrade detection performance across both RGB and infrared modalities. The attack employs a fitness function that jointly optimizes the attack success rate on both sensor streams, computing $r_{attack} = min(r_{inf}, r_{vis})$ to ensure cross-modal effectiveness. This optimization process generates spline-based patch masks that are semantically meaningful and transferable across modalities, making them particularly challenging even for multi-modal sensors.

The defensive strategy leverages multimodal fusion to provide robustness against single-modality and multi-modal attacks. The RGB-infrared combined inference pipeline implements an IoU-based prediction fusion strategy where detections from both modalities are intelligently merged. The fusion algorithm operates on a replacement principle: when detections from RGB and infrared streams exhibit significant spatial overlap (IoU ≥ 0.5), the infrared prediction takes precedence, effectively allowing the thermal modality to serve as a fallback when the visible spectrum is compromised. When there is not enough overlap (IoU < 0.5), both RGB and infrared predictions are retained. This asymmetric fusion strategy is motivated by experimental observations that infrared sensors demonstrate greater resilience to both additive noise perturbations and cross-modal patch attacks.

The unified model wrapper provides a critical abstraction layer that enables seamless integration of heterogeneous YOLO architectures (YOLOv3, YOLOv10) within the evaluation framework. The DetectionResult dataclass standardizes output formats across model types, providing bounding boxes in multiple coordinate systems (xyxy, xywh, normalized, absolute) and facilitating consistent metric computation. This architectural design enables systematic comparison of attack effectiveness across different detector families and training paradigms.

The evaluation methodology employs standard detection metrics (precision, recall, mAP50, mAP50-95) with experimental validation conducted on the LLVIP (Low-Light Visible-Infrared Paired) and DroneVehicle datasets. Results demonstrate that the multimodal fusion strategy substantially mitigates performance degradation under adversarial conditions: when Gaussian noise attacks reduce RGB detection mAP to 20.34%, the fused system achieves a mAP score of 28.38% by leveraging robust infrared predictions. However, the CMPA attack, specifically optimized for cross-modal degradation, achieves suppression of both individual and fused detection systems, highlighting the critical need for adversarially-robust fusion strategies beyond simple IoU-based merging. Despite this, it is observed that the IR modality experiences relatively less degradation than RGB modality.

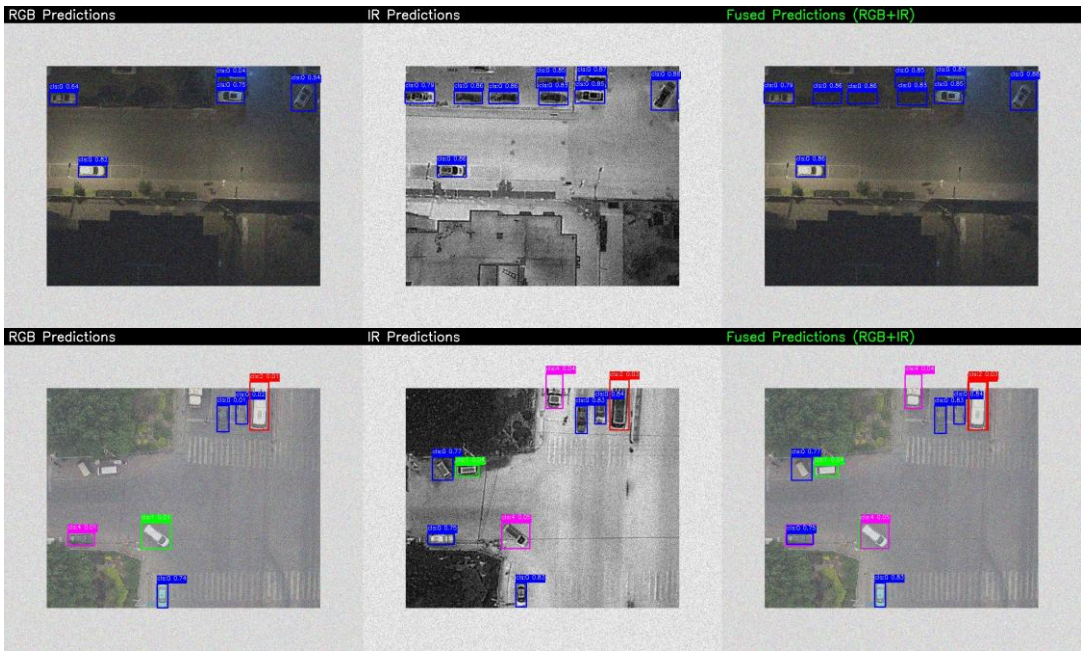


Figure 4.11: Comparison of the prediction performance of the RGB vs IR modalities under adversarial gaussian noise. The IR modality can allow for detecting more objects accurately than the RGB model.

In Figure 4.11, we demonstrate the qualitative comparisons between the RGB and IR modalities when they are attacked by Gaussian noise. The IR modality is much more robust to the gaussian noise while also being able to retain performance under low light conditions. Similarly, under the cross-modal patch attack, we can see that the IR modality allows detecting more accurate bounding boxes than the RGB modality as shown in Figure 4.12.

Table 18: Quantitative comparison between the detection performance of the predictions from the RGB and IR modalities from the DroneVehicle dataset.

Modality	*Precision*	*Recall*	*mAP50*	*mAP50-95*
RGB	60.19	59.83	31.95	20.34
IR	**66.47**	91.15	45.27	27.45
RGB+IR	51.78	**93.27**	**46.49**	**28.38**

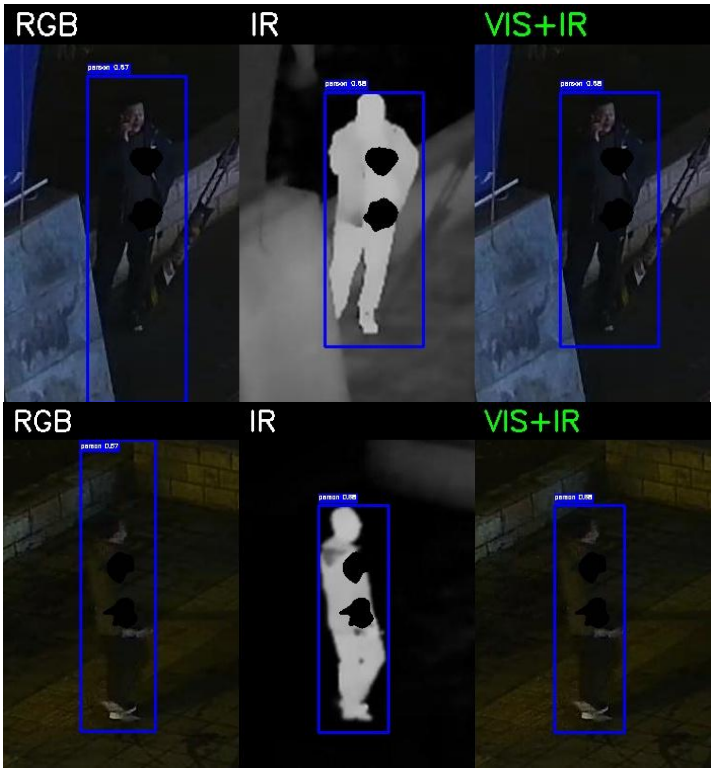


Figure 4.12: Comparison of the prediction performance of the RGB vs IR modalities with adversarial CMPA patches. The IR modality can allow for detecting person bounding boxes more accurately than the RGB model.

This framework provides valuable insights for developing resilient autonomous systems that operate in contested environments, demonstrating both the defensive value of sensor diversity and the sophisticated adversarial capabilities required to compromise multimodal perception pipelines. The modular architecture supports extensibility for future attack methodologies and alternative fusion strategies, establishing a foundation for continued research in adversarial robustness of multimodal perception systems.

4.3.3. Next Steps

To further advance the approach presented in Section 4.3, the following directions have been identified:

1. In the experiments conducted so far, it is seen that the infrared modality is generally more robust to simple additive noise attacks as well as jointly optimized cross-modal patch attacks. Further benchmarking with more diverse noise attacks such as FGSM, PGD, etc. as well as adversarial patches (digital and physical) is required to comprehensively evaluate the behavior of the RGB and IR modalities under different adversarial scenarios.
2. The present defense framework assumes that the input is always adversarially perturbed for the sake of evaluations. Following this, it is required to implement active attack detection modules that can identify if an input image is under adversarial attack. This would also allow us to evaluate the potential performance impact when the defensive fusion strategy is applied on clean input samples.

4.4. Multi-agent Perception Effectiveness and Action Scoring Methodology

In a multi-agent environment, perceptual hashing and Hamming Distance provide a lightweight but reliable mechanism for detecting visual inconsistencies, tampering, or perturbations in sensor data. When an agent processes an image, the perceptual hash acts as a compact signature of the scene content, and the Hamming Distance between the reference hash and the observed hash determines the degree of visual deviation. A small distance indicates stable and trustworthy visual information, whereas a large distance suggests that the image may have altered, corrupted, or adversarially manipulated. This information becomes a critical input to the agent's Perception Effectiveness (PE) score, because perception is only meaningful if the underlying sensor data can be trusted.

Formally, the normalized Hamming Distance is converted into a Visual Integrity Score (VI). This score directly reflects the reliability of the image stream: values near 1 indicate intact, high-quality scene perception, while values approaching 0 signal compromised imagery. The system can incorporate this measure in two ways. First, VI may be fused into the PE metric as a modality-specific confidence factor, reducing the perception score when an agent bases its reasoning on a corrupted visual input. Second, the binary tampering decision (based on a Hamming Distance threshold), can influence the agent's success indicator, altering the Action Score whenever the agent fails to detect manipulated imagery or incorrectly flags benign images as compromised.

At the multi-agent level, compromised-image detection also influences collective awareness. If multiple agents agree on the visual integrity of a scene, cross-agent consensus (CS) increases; if some agents detect tampering and others do not, CS decreases, affecting the Event Score. Similarly, the robustness factor (RB) benefits when agents maintain stable performance under degraded or adversarial visual conditions. When image tampering is detected early, agents can adjust their reasoning and avoid propagating corrupted information, improving both temporal coherence (TC) and context relevance (CR).

Thus, perceptual hashing and Hamming Distance do more than detect compromised frames, they actively shape the situational awareness score by modulating individual perception reliability, penalizing misinterpretation of tampered imagery, and strengthening collective consistency and resilience. This directly enhances the system's ability to maintain accurate, coordinated, and trustworthy awareness in adversarial or degraded sensing environments.

4.4.1. Contextual Awareness in Heterogeneous Multi-Agent Systems

In a multi-agent situational awareness environment, the reliability of sensory inputs is fundamental to how effectively agents can perceive, understand, and react to evolving events. Modern cyber-physical and autonomous systems increasingly depend on visual data streams, making them vulnerable to distortion, noise, or deliberate adversarial manipulation. To address this challenge, the system incorporates perceptual hashing and Hamming Distance analysis as a lightweight yet robust mechanism for assessing the integrity of visual information. A perceptual hash captures the underlying scene structure in a compact signature, and the Hamming Distance between a reference hash and an observed hash quantifies their perceptual difference. Small distances indicate trustworthy imagery, while large distances reveal potential tampering or degradation.

This capability directly reinforces the situational awareness scoring mechanisms. The normalized Hamming Distance is converted into a visual integrity score that forms part of the agent's Perception Effectiveness (PE), ensuring that perception quality reflects not only classification accuracy and detection performance but also the trustworthiness of the underlying sensory data. In cases where tampering is detected, or when an agent fails to detect it, the resulting confidence penalty influences the agent's overall Action Score and, consequently, the collective Multi-Agent Event Score. At the group level, consistent detection of compromised imagery strengthens cross-agent consensus and robustness, while divergent or delayed detections signal weak coordination or degraded sensing, lowering the system's overall situational awareness.

By embedding image integrity assessment within the perception and aggregation layers, the model provides a more realistic and resilient representation of how agents operate in contested or uncertain environments. This integration ensures that situational awareness is not simply a function of performance metrics, but a comprehensive measure of trustworthy perception, coordinated interpretation, and collective resilience, core attributes of an effective multi-agent awareness ecosystem.

4.4.2. Multi-agent Situational Awareness Scoring System

Overview of the Situational Awareness Scoring Approach

The main aim of this part includes the presentation of a comprehensive multi-agent SA scoring model aimed at evaluating how effectively individual agents and groups of agents perceive, interpret, and respond to events.

Agent-level Action Score Model

The score related to the agents' actions may be considered as one of the fundamental blocks of the proposed framework. The Action Score (A) may represent the SA exhibited by an individual agent during a specific event. Each agent computes its A independently, allowing fine-gained analysis of agent behavior while preserving modularity and scalability. The A captures the extent to which an agent responds in a timely, accurate, and contextually appropriate manner.

Formally, A combines four main components. The Time metric (T) captures reaction speed and timeliness, reflecting how quickly the agent perceives and responds to relevant events. Timely responses are essential for effective awareness, particularly in fast-evolving or safety and critical environments. The Perception Effectiveness metric (PE) reflects the quality of the agent’s sensing and interpretation capabilities, incorporating accuracy, latency, false positives, and the trustworthiness of the sensed data. Finally, a successful indicator ensures that only actions that successfully address the task or event contribute positively to SA. The use of configurable weights allows the A formulation to be adapted to different operational priorities.

Perception Effectiveness and Visual Integrity Assessment

PE plays a central role in SA, as incorrect or untrustworthy perception undermines all subsequent reasoning and decision-making. In the proposed model, PE is computed as a composite metric that combines classical detection performance indicators with explicit assessment of data integrity.

Traditional perception metrics include Detection rate (DR), false positive rate (FPR), response time (RT), and accuracy, each normalized to a common scale. These metrics quantify how reliably and efficiently an agent detects and classifies relevant events. However, high numerical performance alone does not guarantee trustworthy perception if the underlying sensory data is degraded, corrupted, or adversarially manipulated.

To address this limitation, the model integrates Visual Integrity assessment factor (VI), based on perceptual hashing and Hamming Distance. Perceptual hashes provide compact representations of image content, enabling efficient comparison between frames. The hamming Distance between these hashes, quantifies perceptual deviation, allowing the system to detect potential image tampering, corruption or unexpected alterations. This distance is normalized and transformed into a visual integrity score that directly influences the PE metric.

When compromised or manipulated imagery is detected, perception confidence is reduced accordingly. This reduction propagates to the A, ensuring that agents operating on unreliable visual inputs are appropriately penalized. By explicitly modelling VI, the framework enforces the principles that SA must be grouped in trustworthy, especially in adversarial or safety critical environments.

Multi-Agent Event-level Awareness Aggregation

While individual A provides valuable insights into agent-level behavior, SA in multi-agent systems emerges from collective performance and coordination. To capture this, the framework introduces the Multi-Agent Event Score (Ee), which aggregates individual A for all agents involved in a given event.

The aggregation process begins by combining individual A using agent specific trust or role weights, allowing more reliable or authoritative agents to have greater influence. The resulting base score is then adjusted through a set of collective factors that capture group-level awareness properties. Cross-agent consensus (CS) measures agreement among agents regarding perception and interpretation of the event. Temporal coherence (TC) reflects synchronization and alignment in response timing. Context consistency (CC) evaluates whether agent actions align with the broader scenario context, while context relevance (CR) assesses the appropriateness of responses relative to mission objectives. Robustness (RB) captures the system's ability to maintain awareness under sensor degradation, noise, or adversarial interference. Finally, an event awareness indicator (aware) ensures that only events that are detected and addressed contribute to SA.

This aggregation layer captures emergent properties that cannot be inferred from individual scores alone, such as coordinated understanding, resilience, and shared situational interpretation.

Overall Situational Awareness Score

At the highest level. The framework computes the overall SA score of all the agents involved by aggregating event-level scores across a mission, scenario, or evaluation period. Each event-level score is weighted according to its significance or criticality, allowing the model to reflect operational priorities and mission relevance.

The resulting overall SA value provides a compact yet expressive measure of collective awareness and resilience. It enables systematic comparison between different system configurations, agent teams, scenarios, or environmental conditions. As such, it supports both offline evaluation and longitudinal analysis of SA trends over time.

Model Implementation and Architecture

The proposed scoring framework has been implemented as a modular Python-based scoring engine, closely aligned with the mathematical model. The implementation consists of three main components: an agent-level scoring module responsible for computing A score, and event-level aggregation module that computes multi-agent event scores, and the overall aggregation layer that produces the final SA value.

This modular architecture ensures reproducibility, extensibility, and ease of integration with existing monitoring systems. The direct correspondence between mathematical definitions and software components enables transparent validation and facilitates future extensions, such as adaptive weighting, learning-based calibration, or additional sensing modalities.

Visualization and model Representation

To enhance interoperability and communication, a visual representation of the scoring framework is provided. A high-level conceptual diagram illustrates the hierarchical flow from A to Ee and ultimately to SA. Additional diagrams depict the multi-agent

aggregation process, showing how consensus, temporal coherence, and robustness factors influence event-level scoring.

These visualizations support understanding among both technical and non-technical stakeholders and provide a clear mapping between conceptual design, mathematical formulation, and implementation.

Action-Level Multi-Metric Fusion

To reflect the reliability of perception and the fusion of heterogeneous performance indicators, the Action Score is redefined as:

$$A = (w_T T + w_{PE} PE + w_X X) \cdot success \quad (20)$$

Where:

- T denotes the normalized time and effort efficiency of the action,
- PE is the Perception Error metric capturing sensor or cognitive uncertainty,
- X represents the extended performance indicators derived from the log classifier and best practices,
- w_T, w_{PE} and w_X are the respective weights with $w_T + w_{PE} + w_X = 1$, $success \in [0,1]$ indicates whether the action achieved its operational objective.

The performance dimension X incorporates both sensor performance and integrity indicators, enabling the system to operate without relying on synthetic or dummy data. The explicit inclusion of *PE* allows penalization when an agent acts upon degraded, poisoned, or misleading observations.

Event-Level Multi-Agent Awareness

At the event level, collective awareness is computed through the weighted contribution of all agents participating in event e, as:

$$E = \left(\sum_{a=1}^{N} w_a A_{a,e} \right) CS^{\lambda_1} TC^{\lambda_2} CC^{\lambda_3} CR^{\lambda_4} RB^{\lambda_1} \cdot aware \quad (21)$$

With:

- w_a, agent *a* trust or role weight assigned by the moderator or service broker,
- $A_{a,e,}$ the action score of agents a related to event *e*,
- CS: cross-agent consensus measuring agreement on perceived event attributes,
- TC: Temporal coherence evaluating stability of perception across time windows,
- CC: *Context consistency reflecting logical compatibility with the exercise narrative,*
- CR: Context relevance indicating the importance of the event for mission goals,
- RB: *Robustness to degradation or attack assessing tolerance to missing or manipulated data,*
- λ_i: influence parameters controlling the impact of each dimension,
- $aware \in \{0,1\}$ that indicates whether the event was detected

The exponents λ_i enables nonlinear control of the scoring behavior. Values $\lambda_i > 1$ amplify the effect of the corresponding factor, while $\lambda_i < 1$ attenuate it. This mechanism allows the AI-moderator to adapt to exercise difficulty and provide continuous forecasting of individual and team performance.

The event model E_e is extended to incorporate new dimensions reflecting:

- **Collective perception coherence**: alignment of agents on shared sensor, views,
- **Perception robustness**: ability to maintain awareness under EPSS-driven dynamic risk conditions,
- **Multi-agent knowledge sharing** enabled by federated cyber-range orchestration

These enhancements transform the awareness engine into a unified evaluator suitable for federated providers and heterogeneous human/ machine teams.

4.4.3. Next Steps

The proposed multi-agent SA scoring method establishes a structured and interpretable foundation for assessing individual and collective awareness under both nominal and adversarial conditions. Building on this framework, several directions are identified for future work.

As a first extension concerns adaptive and context-aware weight calibration. While the current formulation relies on configurable but static weights and influence parameters, future work will investigate adaptive strategies in which weight are dynamically adjusted based on scenario difficulty, environmental conditions, or detected integrity degradation. This would allow the SA to respond more sensitively to evolving operational contexts and adversarial pressure.

Second, the integration of temporal awareness modelling represents a natural profession of the event-level aggregation. Rather than evaluating events in isolation, future work will explore temporal smoothing and trend analysis of A, E, and overall SA scores to capture awareness drift, delayed reactions, and recovery patterns over time. Such temporal modelling is particularly relevant for long-running missions and continuous cyber-physical exercises.

Third, the tight coupling between visual integrity assessment and decision support will be further developed. While the current framework propagates integrity degradation through PE and action scores, future extensions will explicitly link integrity-aware SA outputs to adaptive system responses, such as confidence-weighted decision-making, sensor redundancy activation, or escalation to human moderators. This step will transform SA scoring from an evaluating mechanism into an operational resilience enabler.

In addition, future work may focus on multi-modal and cross-agent integrity reasoning, extending visual integrity assessment to incorporate corroboration from non-visual sensors (e.g., LiDAR, telemetry, logs) and inter-agent knowledge sharing.

5. Anomaly Detection Framework

5.1. CAV Adversarial Attack Simulator

Anomaly detection is the task of identifying observations or patterns that deviate from expected or normal behavior. The foundational approach to anomaly detection involves building appropriate datasets of normal operation and abnormal operation to effectively train various detection methods. To enhance this process, efforts focus on developing Generative AI and data augmentation techniques that predict future states and learn to identify deviations from normal operation. Furthermore, the investigation includes deep learning-based autoencoders to accurately model the normal data distribution, allowing for the identification of abnormal instances that lie outside the training distribution, ideally without incurring additional computational costs. This holistic strategy is rounded out by exploring approaches for anomaly detection by correlating data from different sources to improve reliability and robustness.

Real-world testing of critical edge cases, particularly those involving intentional adversarial attacks, is inherently costly, time-consuming, and presents significant safety and legal challenges. The prohibitive nature of such physical testing underscores the need for alternative, highly controlled evaluation environments. This imperative is met by high-fidelity simulators, which are identified as essential tools for the rigorous, repeatable, and systematic evaluation of Autonomous Vehicle (AV) performance. The proposed CAV Adversarial Attack Simulator[96] can conduct comprehensive testing under precisely controlled adversarial conditions by utilizing a high-fidelity simulator. A crucial secondary role of the simulator is to facilitate the building of appropriate datasets, encompassing both normal and abnormal operational data, which are subsequently used to train diverse approaches for anomaly detection within the AV systems. This CAV simulator pipeline is outlined in Figure 5.1.

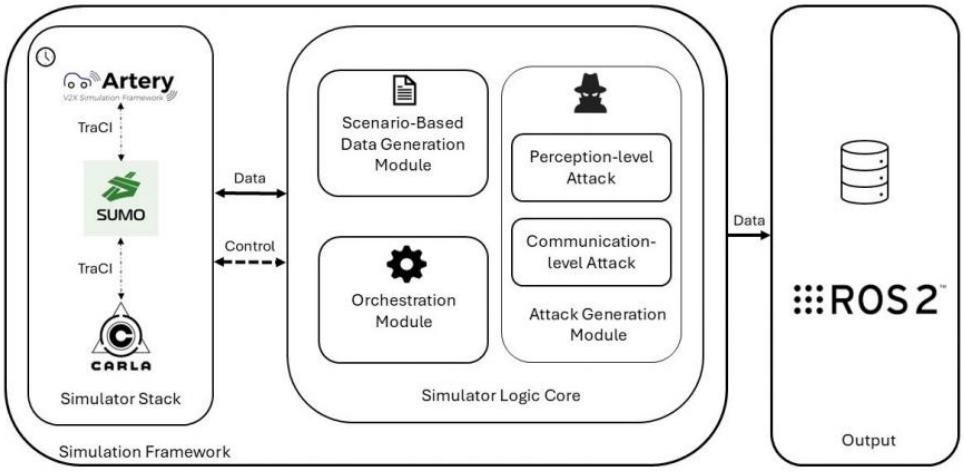


Figure 5.1. CAV Adversarial Attack Simulator pipeline for Anomaly Detection Frameworks.

The CAV Adversarial Attack Simulator is built upon an integrated framework designed to support the sophisticated simulation of cooperative autonomous vehicle (CAV)

[96] C. Anagnostopoulos et al., "Integrated Simulation Framework for Adversarial Attacks on Autonomous Vehicles", in Proc. of IEEE International Smart Cities Conference, 2025

systems. The framework utilizes a specialized component stack where CARLA is employed to render the physical environment and generate realistic sensor data, including LiDAR and camera streams. Concurrently, SUMO manages the traffic simulation component, which is crucial for implementing and testing complex traffic scenarios. Furthermore, Artery/OMNeT++ is incorporated to simulate Vehicle-to-Everything (V2X) communication based on the established ETSI ITS-G5 protocols. This integration of tools is essential for supporting the high-fidelity simulation requirements of CAV systems.

The core operational intelligence of this simulator is the Scenario Logic Core (SLC) Module operating as the central governor of the simulation's functionality. This SLC Module integrates three essential, highly coordinated components designed to enable rigorous and comprehensive testing of Connected and Autonomous Vehicles (CAVs): the Scenario-Based Data Generation, Orchestration, and Attack Generation modules. Firstly, the Scenario-Based Data Generation module ensures consistency by parsing a unified scenario description file to automatically initialize the entire simulation environment and establish the baseline experimental conditions. It is noted that the Scenario-Based Data Generation Module serves as the initialization component for the simulator environment. This module is designed to parse a single, unified configuration file to automatically initialize all simulators involved in the experiment. The configuration file is comprehensive, defining general simulation parameters such as the duration and time step. Crucially, it also includes simulator-specific settings, such as configuration parameters for CARLA (weather and sensor configurations), SUMO (traffic routes and vehicle behavior), and Artery (transmission power and emission parameters). This standardized file ensures the coherent and automatic setup of the entire simulation environment for consistent experimentation. This setup is then managed by the Orchestration module, which is critically tasked with synchronizing and controlling the diverse elements of the CARLA, SUMO, and Artery simulators enabling coherent execution throughout the test. The evaluation culminates with the Attack Generation module, which provides critical security and robustness testing capabilities through its integrated perception attack engine and communication-level adversarial attack engine. More specifically, the Attack Generation Module is a critical component of the system, designed with the central purpose of providing a unified environment to generate adversarial attacks within the simulation framework. This module supports two distinct adversarial attack domains, the Perception-Level Attack Engine, which directly manipulates sensor data like LiDAR point clouds using techniques such as Point Perturbation, Point Detachment, and Point Attachment to challenge 3D object detectors, and the Communication-Level Attack Engine, which targets V2X data exchange and GNSS signals to compromise shared situational awareness through methods including GNSS spoofing, Forged or replayed Cooperative Awareness Messages (CAMs), and Sybil attacks. This dual-engine capability ensures the rigorous and complete testing of the CAV system's security against both direct sensor and communication-based threats.

5.1.1. Next steps

The CAV Adversarial Attack Simulator establishes a unified and high-fidelity foundation for generating and executing security-relevant CAV scenarios across perception and V2X layers. Building on this capability, the next steps involve the creation of scenario-specific datasets of normal/abnormal operation, emphasizing rare but safety-critical events such as the ones involved in autonomous driving use-cases (e.g., adversarial attacks on perception sensors, GNSS spoofing, attack on communication infrastructure, etc.) that are hard to reproduce in reality. These datasets will enrich

GuardAI's anomaly detection framework and will guide the selection of appropriate threat/defense pairs in task T2.5 (see also Section 6.3).

5.2. AI-enabled Intrusion Detection System

5.2.1. Core Detection Mechanisms of CERTH's IDS tool

Deep Learning-based security incident detection

The Deep Learning-based security incident detection framework represents a sophisticated layer of defense designed to address the unique challenges of modern telecommunications environments. Specifically, the Intrusion Detection System (IDS) employs advanced Artificial Intelligence algorithms to ingest and analyze the high-velocity network data generated by CERTH's 5G testbed. Unlike traditional detection methods, which rely on static signatures, this system utilizes Deep Learning (DL) to autonomously learn complex, non-linear representations of network behavior. This capability is critical for identifying potential dangers and recognizing subtle traffic patterns or statistical irregularities that would otherwise remain obscured or difficult to spot under conventional monitoring circumstances. By continuously adapting to the dynamic nature of 5G traffic, the model effectively uncovers sophisticated threats, such as zero-day attacks or low-footprint anomalies, ensuring a level of visibility and security that standard rule-based systems cannot provide.

Network-based analysis

Network-based detection constitutes a critical component of the IDS framework, leveraging Machine Learning (ML) and DL models to perform robust analysis on network flow data. Unlike resource-intensive full packet inspection, network flow analysis provides a strategic balance of scalability and visibility by capturing high-fidelity metadata, such as source and destination IPs, ports, protocols, and traffic volume. This abstraction enables both real-time monitoring and retrospective analysis of infrastructure communication, making it highly effective for identifying unusual traffic spikes, reconnaissance scanning, data exfiltration, and lateral movement. To operationalize this data, the IDS employs the open-source CICFlowMeter tool as the initial ingestion step, parsing raw packet capture (.pcap) files to generate a granular feature set. This process creates structured CSV datasets containing over 80 statistical metrics, capturing bidirectional flow dynamics such as flow duration, packet counts, inter-arrival times, and byte volume. These features serve as the input vector for the system's pretrained models, enabling the engine to reliably discriminate between legitimate operational traffic and complex patterns indicative of malicious behavior.

A distinguishing feature of this architecture is the implementation of a protocol-specific modeling strategy rather than a monolithic, all-encompassing detection engine. Recognizing that protocols, such as HTTP, DNS, FTP and SMB exhibit unique behavioral norms and distinct security risks, the system utilizes specialized models trained specifically for each communication type. This modular approach allows the detection engine to capture subtle, protocol-centric anomalies that generalized models often miss, while also offering superior interpretability and simplified retraining lifecycles. Following the detection and classification of a malicious flow, the system ensures interoperability with downstream security tools by translating the findings into the STIX 2.1 standard. This process constructs a comprehensive graph of intelligence

objects, including network traffic and IPv4 objects to define the technical parameters, an observed data object to reference the flow artifacts and an attack pattern object that semantically links the incident to the CAPEC framework.

5.2.2. Training & Self-Learning

AI/ML Retraining

To maintain resilience against evolving threats within the experimental 5G environment, the system employs a dynamic retraining strategy driven by the Attack Generation Engine (AGE) module. This simulation function is utilized not only to test system responsiveness but also to trigger a robust model retraining process by injecting simulated logs corresponding to cyberattacks that were previously unknown to the AI/ML detection algorithms. These simulated attack artifacts are processed separately and then combined with existing datasets, effectively enriching the detection knowledge base with new samples. By systematically updating the models with this fresh data, the system ensures that its DL algorithms remain aligned with the latest cyber threat intelligence. This continuous learning cycle results in detection models with enhanced performance, capable of recognizing a broader spectrum of sophisticated cyberattacks and adapting to the specific behavioral norms of the testbed environment.

5.2.3. User Interface & Analytics

Visual Analytics Toolset (Web UI/Dashboard)

The solution features a comprehensive, customizable dashboard that serves as a central hub for Security Operations Center (SOC) analysts to investigate incidents. This Visual Analytics Toolset aggregates and visualizes data from heterogeneous sources, allowing users to inspect raw data, perform historical analysis and identify abnormal patterns through various graphical widgets. The interface supports deep-dive investigations through interactive visualizations, such as network flow graphs composed of node clusters that allow users to visually pinpoint IPs involved in security incidents. Additionally, specific widgets for network packets and protocol distribution, such as bubble charts or pie charts provide granular insights into traffic behavior, enabling analysts to filter by time range or protocol to isolate specific anomalies within the dense 5G traffic.

Real-Time Status

For immediate situational awareness, the upper layer of the dashboard provides a "Live Network Security Status" widget that quantifies the infrastructure's health. This status is characterized by a real-time percentage score, calculated based on the ratio of detected security events to the total number of network IPs, offering an at-a-glance metric of system integrity. Alongside this score, the interface displays the specific count of security events detected within the last hour and explicitly lists the "problematic network nodes" currently affecting the network. This is further supported by a Trust Management page, which assigns a reputation value (0-100) to each asset; assets with low reputation scores due to recent incidents are flagged as insecure, allowing operators to prioritize their response to the most critical nodes in the testbed.

5.2.4. Next steps

The next steps related to the AI-enabled IDS of Use Case 2 (UC2) is to transition the IDS from a detection-oriented component into a fully integrated, closed-loop security framework within the UC2 architecture. This phase focuses on operationalizing the link between the detection engine and the Mitigation component. Specifically, once the IDS translates a detected abnormality, it will automatically trigger the mitigation module. This module is designed to execute a multi-objective optimization process and by evaluating potential countermeasures against conflicting objectives, such as effectiveness, relevance, response time and cost, the system will generate the most optimal mitigation strategies. This integration ensures that the IDS not only identifies protocol-specific anomalies but also drives rapid, data-driven responses, forwarding actionable "Course of Action" objects to the response planner. Concurrently, the roadmap includes strengthening the system's resilience against dynamic, adversarial threats typical of Edge AI and 5G environments. A key step involves fully automating the self-learning module to minimize human intervention in the model update lifecycle. The technical focus will be on tightening the integration with the Attack Simulation Engine (AGE) module to create a continuous feedback loop. By systematically injecting simulated attack logs into the training pipeline, the system will iteratively retrain its DL models. This will validate the IDS's ability to adapt to new attack vectors in real-time, ensuring robust security coverage for the evolving 5G testbed infrastructure.

5.3. Robust Uncertainty Quantification

Modern deep learning models have delivered strong performance across a wide range of decision-making tasks, but their practical use in safety-critical settings is often limited by poorly calibrated confidence estimates. In particular, standard models can remain highly confident even when inputs deviate from the training distribution due to distributional or covariate shift, adversarial perturbations, sensor/data noise, or other real-world artifacts, which can result in unsafe or misleading decisions. Robust uncertainty quantification aims to address this gap by equipping models with reliable measures of predictive uncertainty, so they can signal when a prediction may be unreliable and support more cautious downstream behavior (e.g., fallback policies or human review).

Our goal is therefore to develop a fully post-hoc uncertainty quantification method that can be attached to an already-trained base model (without changing its architecture or parameters), avoids inheriting its overconfidence, and remains robust under distribution shift and adversarial conditions, while preserving in-distribution performance and requiring minimal user intervention. To this extent, we developed the following comprehensive approaches.

Conflict-aware Evidential Deep Learning (C-EDL) is a lightweight, post-hoc uncertainty quantification method designed to address a key failure mode of standard Evidential Deep Learning (EDL): overconfident predictions under adversarial perturbations and distributional shift. While EDL models predictive uncertainty through a Dirichlet distribution in a single deterministic forward pass, it lacks a mechanism to detect inconsistency in its own evidence when the input is perturbed, making it vulnerable to adversarial manipulation.

C-EDL, augments a pretrained EDL classifier by introducing multiple, label-preserving views of each input and explicitly quantifying conflict across the resulting evidence distributions. The method does not modify the base model's parameters and can be applied post hoc to any trained EDL network.

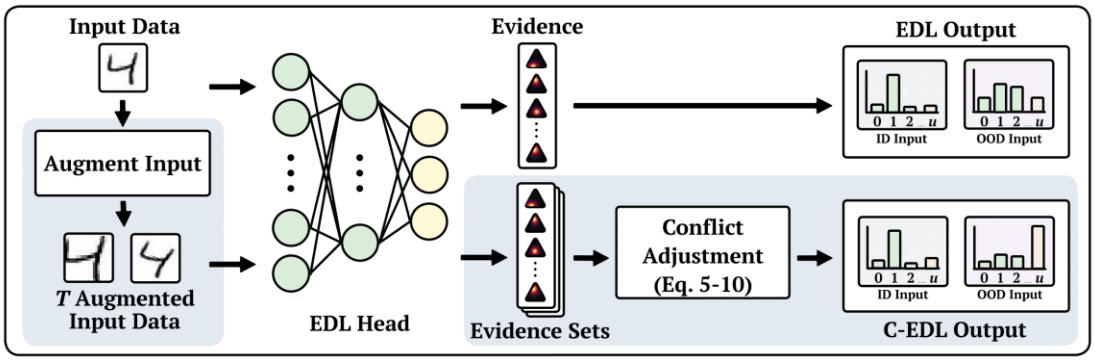


Figure 5.2: Overview of Conflict-aware Evidential Deep Learning (C-EDL) approach, with its key post-hoc steps that advance regular EDL highlighted in blue. For each new input, C-EDL performs metamorphic transformations, yielding a label-preserving evidence set, and then executes conflict adjustment on the accumulated evidence to calibrate the final prediction. When applied to in-distribution inputs, C-EDL closely matches the original EDL output, while given out-of-distribution inputs, C-EDL amplifies uncertainty to better reflect model disagreement.

Given an input, C-EDL generates a set of metamorphic transformations, satisfying a label-preserving constraint under the ground-truth function. Each transformed input is passed independently through the frozen EDL model, producing a corresponding set of Dirichlet parameters. This collection forms an evidence set capturing the variability of the model's beliefs under controlled representational shifts.

C-EDL then computes a conflict score that measures disagreement across this evidence set along two complementary axes. Intra-class conflict captures the variability of evidence assigned to the same class across transformations:

$$C_{intra} = \frac{1}{K}\sum_{k=1}^{K}\frac{\sigma\left(\left\{\alpha_k^{(t)}\right\}_{t=1}^{T}\right)}{\mu\left(\left\{\alpha_k^{(t)}\right\}_{t=1}^{T}\right) + \epsilon} \tag{22}$$

while inter-class conflict measures contradictory support for competing classes within individual transformations:

$$C_{inter} = \frac{1}{T}\sum_{t=1}^{T}\left(1 - \exp\left(-\beta\sum_{k=1}^{K}\sum_{j=k+1}^{K}\left(\frac{\min\left(\alpha_k^{(t)}, \alpha_j^{(t)}\right)}{\max\left(\alpha_k^{(t)}, \alpha_j^{(t)}\right)} \times \frac{\min\left(\alpha_k^{(t)}, \alpha_j^{(t)}\right)}{\sum_{k=1}^{K}\alpha_k^{(t)}} \times 2\right)^2\right)\right) \tag{23}$$

These components are combined into a bounded conflict measure that increases monotonically with disagreement and provably approaches zero only when all transformations yield identical, concentrated evidence:

$$C = C_{inter} + C_{intra} - C_{inter}C_{intra} - \lambda(C_{inter} - C_{intra})^2 \tag{24}$$

Rather than altering class probabilities directly, C-EDL applies conflict-aware calibration by scaling the magnitude of evidence. The aggregated Dirichlet parameters are exponentially decayed as where controls sensitivity to conflict. This preserves the relative shape of the predictive distribution while reducing its total strength. As a result, the predicted class remains unchanged, but the associated uncertainty mass increases in proportion to detected conflict. When conflict is low (as in clean in-distribution data), the adjustment is negligible and C-EDL behaves identically to standard EDL. The mass belief, and probabilities of predictions, seen below:

$$\tilde{S} = \sum_{k=1}^{K} \widetilde{\alpha_k}, \quad \widetilde{b_k} = \frac{\widetilde{\alpha_k} - 1}{\tilde{S}}, \quad \tilde{u} = \frac{K}{\tilde{S}}, \quad \mathrm{E}[\widetilde{p_k}] = \frac{\widetilde{\alpha_k}}{\tilde{S}} \tag{25}$$

Two practical variants are evaluated: C-EDL (Meta), which uses deterministic metamorphic transformations, and C-EDL (MC), which replaces transformations with Monte Carlo Dropout samples. The Meta variant consistently yields stronger uncertainty signals, highlighting the importance of structured, task-preserving perturbations over stochastic sampling.

C-EDL is evaluated across a wide range of datasets, adversarial attacks, and uncertainty thresholds, with performance measured primarily through coverage: the proportion of inputs retained after abstention. Low coverage on adversarial or OOD inputs indicates effective uncertainty estimation, while high coverage on in-distribution data reflects preserved predictive utility.

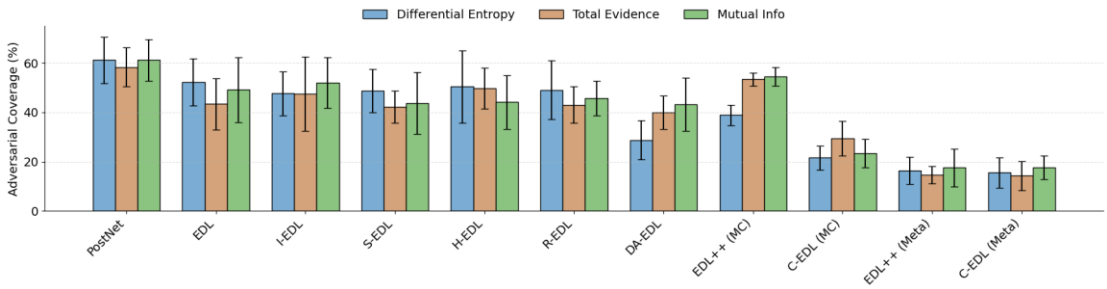


Figure 5.3: Adversarial coverage (%) for different ID-OOD threshold metrics where the ID dataset is MNIST, the OOD dataset is FashionMNIST, and the adversarial attack is L2PGD (=1.0).

The bar plot in Figure 5.3 compares adversarial coverage under three uncertainty measures: differential entropy, total evidence, and mutual information. Across all metrics, C-EDL (both Meta and MC variants) achieves substantially lower adversarial coverage than standard EDL and its extensions (I-EDL, S-EDL, H-EDL, R-EDL, DA-EDL). In many cases, adversarial coverage is reduced by more than half, and by up to an order of magnitude compared to baseline EDL. Importantly, this reduction is consistent across metrics, indicating that C-EDL's gains are not tied to a particular thresholding strategy.

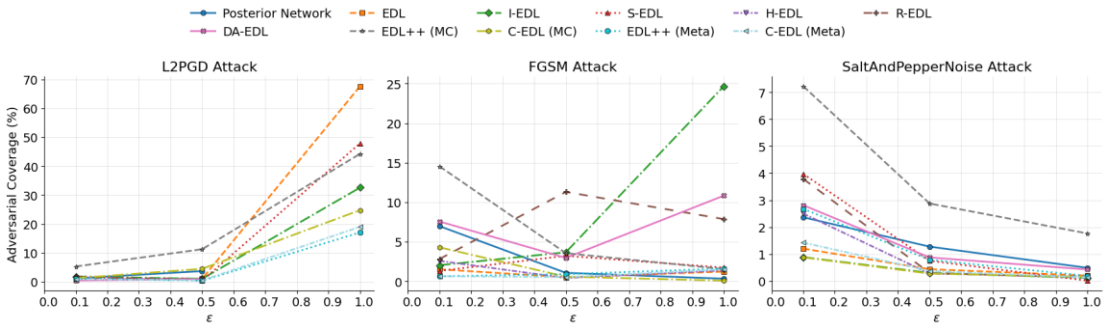


Figure 5.4: Adversarial coverage (%) across varying perturbation strengths () for three attack types (L2PGD, FGSM, and Salt and Pepper noise), where the ID dataset is MNIST, and the OOD dataset is FashionMNIST. Lower coverage indicates better robustness.

The adversarial attack curves in Figure 5.4 further illustrate how C-EDL behaves as perturbation strength increases under L2-PGD, FGSM, and Salt-and-Pepper noise. For strong gradient-based attacks such as L2-PGD, standard EDL exhibits rapidly increasing coverage, exceeding 60–70% at high. In contrast, C-EDL (Meta) maintains low coverage throughout, remaining below 20% even at the highest perturbation levels. Similar trends hold for FGSM, where most methods degrade sharply, while C-EDL variants remain near zero coverage.

Notably, C-EDL also performs strongly under non-gradient-based Salt-and-Pepper noise. While many uncertainty methods struggle in this regime, reflecting reliance on gradient-aligned signals, C-EDL's disagreement-based mechanism generalizes across attack types. This demonstrates that conflict across task-preserving views captures a more fundamental notion of epistemic instability than gradient sensitivity alone.

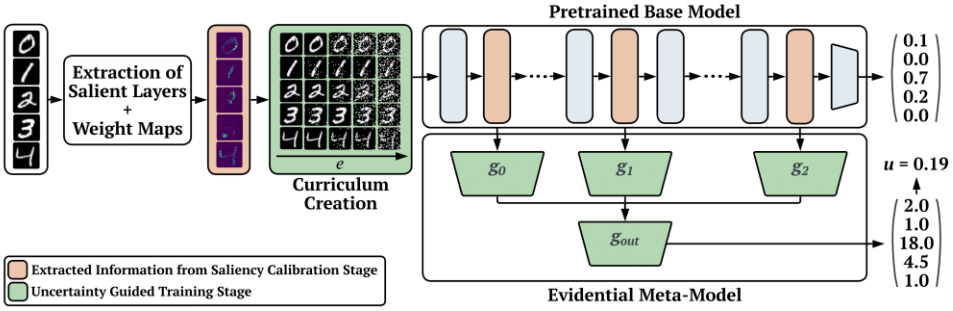


Figure 5.5: Overview of the GUIDE meta-model approach, showing the saliency Calibration and Uncertainty Guided Training stages. GUIDE extracts salient features and weight maps from a pretrained model in a fully post-hoc manner, then generates a noise-driven curriculum to teach the meta-model when to be uncertain. In the figure, g_ℓ are evidential projection branches from salient layers, and u is the predicted uncertainty from the Dirichlet evidence.

Second, the Gradual Uncertainty Refinement via Noise-Driven Curriculum (GUIDE) approach is a fully post-hoc uncertainty quantification approach that explicitly learns when and how much to be uncertain, without modifying or retraining the underlying predictive model. GUIDE operates by attaching a lightweight, evidential meta-model to a frozen, pretrained network and training this auxiliary model to express calibrated epistemic uncertainty under distributional shift and adversarial perturbation.

The approach consists of two stages: saliency calibration and uncertainty-guided training. In the first stage, GUIDE identifies a compact set of informative internal representations from the pretrained model. Using layer-wise relevance propagation, each hidden layer is assigned a relevance score reflecting its contribution to the final prediction:

$$R_{\ell-1}(x) = \left(\frac{\alpha_{\ell-1}W_\ell^\top}{<\alpha_{\ell-1}W_\ell^\top> + \epsilon}\right) \odot R_\ell(x), \quad \ell = L, L-1, \ldots, 1 \tag{26}$$

Layers are ranked by relevance:

$$M_\ell = \frac{1}{N}\sum_{i=1}^{N}\frac{||R_\ell(x_i)||_1}{|R_\ell(x_i)|} \tag{27}$$

and the smallest subset is selected such that their cumulative relevance exceeds a predefined coverage threshold η:

$$\sum_{\ell \in L_{sal}} M_\ell \geq \eta \sum_{\ell} M_\ell \tag{28}$$

This avoids arbitrary layer selection and ensures that the meta-model operates on semantically meaningful features rather than redundant or weakly informative activations.

In parallel, input-level relevance maps are extracted and normalized to form saliency weight maps. These maps identify spatial regions that most strongly influence the base model's prediction and are later used to guide targeted input corruption. Importantly, this calibration stage requires only a single backward relevance pass and introduces no additional training cost for the base model.

In the second stage, GUIDE trains an evidential meta-model on features extracted from the selected layers. For each selected layer ℓ, its activations $\Phi_\ell(x)$ are projected through a small linear head and aggregated to produce Dirichlet parameters $\alpha(x)$, defining a predictive distribution $Dir(\pi|\alpha)$. The total evidence $S = \sum_k a_k$ serves as a measure of confidence: high evidence corresponds to confident predictions, while low evidence reflects epistemic uncertainty.

To teach the meta-model how uncertainty should evolve under degradation, GUIDE constructs a noise-driven curriculum. Inputs are progressively corrupted according to a monotonic schedule $s_t \in [0,1]$, where early stages apply mild perturbations and later stages apply strong corruption. Crucially, corruption is applied in a saliency-aware manner: at low noise levels, perturbations are concentrated in highly relevant regions, while at higher levels the corruption becomes more global. This produces a sequence of inputs that smoothly transition from in-distribution to strongly out-of-distribution.

Rather than using hard labels, GUIDE employs soft uncertainty targets that interpolate between the ground-truth label and the uniform distribution. The interpolation weight depends jointly on the corruption level and the base model's confidence on the corrupted input, encouraging the meta-model to remain confident only when both evidence and input quality justify it. Training is further stabilized with a self-rejecting evidence penalty that suppresses high evidence when predictions disagree with the

target distribution. All components are trained end-to-end while keeping the base model fully frozen, the fully loss is combined below:

$$L = \frac{1}{|B|} \sum_{(\tilde{x},\tilde{y})\in B} \left[-\sum_{k=1}^{K} \widetilde{y_k}\left(\psi(\alpha_k) - \psi(S)\right) + \lambda_{kl} \cdot KL\left(Dir(\alpha)||Dir(\beta)\right) \right] + \frac{S}{K} \cdot (1 \prec \tilde{y}, \hat{p} \succ) \quad (29)$$

Across standard benchmarks and a wide range of evaluation settings, GUIDE consistently produces more reliable uncertainty estimates than both intrusive and post-hoc baselines, while preserving in-distribution predictive performance.

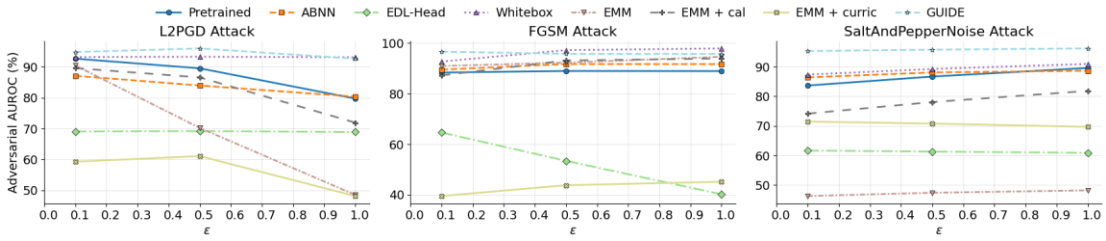


Figure 5.6: Adversarial AUROC (%) across varying perturbation strengths (ϵ) for three attack types (L2PGD, FGSM, and Salt and Pepper noise) where the ID -> OOD dataset is MNIST and FashionMNIST. Higher AUROC indicates better robustness.

Figure 5.6 shows adversarial AUROC as a function of perturbation strength ϵ for L2-PGD, FGSM, and Salt-and-Pepper attacks. GUIDE maintains high AUROC (typically above 90%) across all attack types and perturbation levels. In contrast, the pretrained model, ABNN, and EMM exhibit substantial degradation as ϵ increases, often falling below 70% AUROC under moderate perturbations. Notably, GUIDE remains robust even under non-gradient-based Salt-and-Pepper noise, indicating that its uncertainty estimates are not tied to gradient-specific artefacts but instead reflect broader distributional degradation.

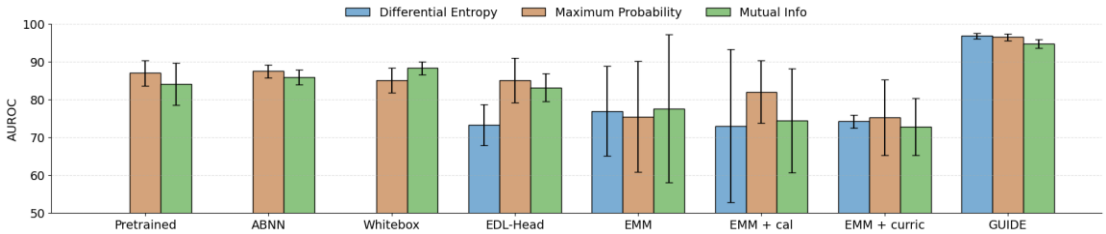


Figure 5.7: AUROC for different ID-OOD threshold metrics (differential entropy, maximum probability, mutual information) where the ID dataset is MNIST, and the OOD dataset is FashionMNIST.

Figure 5.7 evaluates AUROC under different uncertainty measures, including differential entropy, maximum predicted probability, and mutual information. While baseline methods show large variance and sensitivity to the chosen metric, often dropping below 80% AUROC, GUIDE consistently achieves high AUROC (≈95%) with tight confidence intervals across all metrics. This indicates that GUIDE's uncertainty estimates are internally consistent and robust to the choice of decision threshold, an important property for deployment in safety-critical systems.

C-EDL is designed to wrap around GUIDE, forming a post-hoc wrapper over a post-hoc uncertainty model. While GUIDE focuses on learning calibrated evidential uncertainty from salient representations of a frozen base model, C-EDL operates one level above by auditing the consistency of GUIDE's predictions under label-preserving perturbations. By injecting conflict-aware modulation on top of GUIDE's evidential outputs, C-EDL further refines uncertainty estimates without modifying either the original predictor or GUIDE itself. This layered design preserves GUIDE's non-intrusive philosophy while adding an additional safeguard against spurious confidence, particularly under distribution shift and adversarial conditions.

5.3.1. Next steps

Concerning the Robust Uncertainty Quantification modules, our immediate next steps are twofold. First, we will investigate the integration of these modules with the approaches supporting enhanced adversarial resiliency thus providing a unified pipeline that can offer complementary benefits and increase further robustness for edge AI applications. Also, we will investigate the integration of GUIDE and C-EDL into a composite framework that would enable its application to an already trained DL model (acting as the backbone) without requiring any changes to the trained model, ultimately improving the flexibility and generalizability of our approaches. Finally, we will explore mechanisms to the extend both modules with support for other tasks, such as regression, further enhancing the applicability of both GUIDE and C-EDL.

5.4. NWDAF-based Anomaly Detection for 5G Networks

The proposed anomaly detection framework leverages the Network Data Analytics Function (NWDAF) as a central intelligence layer for detecting security incidents in 5G/B5G networks. NWDAF enables the continuous collection and analysis of operational metrics originating from multiple network functions across the control and data planes. Unlike traditional intrusion detection approaches that rely on packet-level inspection or static thresholds, this framework operates on time-series data, allowing it to identify anomalous network behaviors that are symptomatic of security incidents, such as Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) attacks.

Our objectives are to evaluate the effectiveness of different anomaly detection techniques when applied to real 5G operational data, and to demonstrate how such analytics can be integrated into an end-to-end pipeline including data generation, model training, real-time anomaly detection, and visualization.

5.4.1. 5G Testbed and Dataset Generation

The dataset used in the experimental evaluation was generated using an in-house "5G-in-a-box" platform that implements a full-stack 5G system integrating core, edge, and radio access components. The platform is based on the Amarisoft Callbox Classic solution and supports controlled experimentation under realistic operational conditions.

Metrics are collected continuously from both core and radio network functions and aggregated over fixed time windows. The resulting dataset provides fine-grained visibility into network behavior across multiple protocol layers. Control-plane activity is captured through detailed counters of 5G NAS, NGAP, and Diameter signaling exchanged between UEs, gNBs, and core network functions. These metrics include registration and authentication procedures, PDU session establishment, paging

activity, and UE context management, allowing the detection of abnormal signaling patterns such as excessive attach/detach cycles and signaling floods. In parallel, data-plane behavior is represented through PDN/APN traffic statistics and cell-level throughput measurements, including uplink and downlink bitrates, transmitted volumes, retransmissions, and error counters. These indicators enable the identification of traffic surges, sustained abnormal load, and asymmetric UL/DL patterns commonly associated with data-plane DoS attacks targeting services or edge applications.

Beyond signaling and traffic metrics, the dataset incorporates detailed radio access and physical-layer indicators. These include PRB utilization, UE and bearer population statistics, scheduling activity, and RF/PHY processing metrics such as RX/TX processing load, CPU time, RX-to-TX delays, and signal sample statistics. The inclusion of these metrics allows the analysis to capture cross-layer effects and assess how security incidents impact radio resource utilization and low-level processing behavior.

5.4.2. Anomaly Detection Methods

To assess anomaly detection performance over the developed dataset, a set of statistical and unsupervised machine learning models was selected.

The following statistical methods were selected due to their low computational complexity, strong interpretability, and suitability for detecting abrupt deviations in time-series metrics:

- Z-Score: Detects anomalies based on deviations from the mean, expressed in terms of standard deviations.
- Median Absolute Deviation (MAD): A robust alternative to mean-based approaches, MAD is less sensitive to extreme values and non-Gaussian distributions.
- Hampel Filter: A sliding-window technique that identifies outliers based on the local median and dispersion.

In addition, the following unsupervised machine learning techniques were evaluated to capture more complex and non-linear deviations across multiple metrics:

- Isolation Forest: An ensemble-based method that isolates anomalies by recursively partitioning the feature space.
- Local Outlier Factor (LOF): A density-based method that identifies samples that deviate significantly from their local neighborhood.
- One-Class Support Vector Machine (OC-SVM): Learns a boundary around normal data in a high-dimensional space and flags deviations as anomalies.

All models were applied to preprocessed time-series data derived from the collected metrics, with feature selection tailored to the considered attack scenarios.

The comparative evaluation demonstrated clear performance differences between statistical and machine learning techniques, depending on the nature of the anomaly scenario. For control-plane DoS scenarios, statistical methods consistently achieved higher accuracy, recall, and F1 scores, specifically the Hampel Filter and Z-Score detection. Machine learning models such as Isolation Forest and OC-SVM also

performed adequately in these scenarios but did not consistently outperform statistical techniques. For data-plane DoS scenarios, performance varied more significantly across all models. Statistical methods remained effective in detecting pronounced traffic spikes; however, their performance degraded in scenarios where anomalies manifested as sustained but less abrupt deviations.

Overall, the results confirm that statistical methods are well suited for predictable, high-volume anomalies such as signaling floods or traffic surges, offering strong performance, high interpretability, and low computational complexity, while machine learning approaches show potential in detecting more complex or persistent anomalies, with their effectiveness depending on appropriate feature selection and parameter tuning. Based on these findings, the most effective strategy for NWDAF-based security analytics would be a hybrid approach that combines both statistical techniques for fast and reliable detection of high-volume anomalies and ML models capable of capturing more subtle or evolving attack patterns.

5.4.3. End-to-End NWDAF-based Anomaly Detection Pipeline

In addition to the dataset generation and model experimentation activities, the goal of this work is to establish an end-to-end security analytics pipeline operating on a real 5G testbed. The testbed serves both as a controlled environment for generating datasets used to train and benchmark anomaly detection models, and as a live source of operational metrics for online inference.

Here, trained models can be deployed to continuously analyze streaming metrics originating from the testbed, enabling timely detection of anomalous behavior across control-plane and data-plane operations. Once anomalies or attack-related events are detected, the corresponding alerts and contextual information are forwarded to a visualization layer, where a UI-based dashboard presents them in real time.

5.4.4. Next Steps

Ongoing work focuses on extending the experimental evaluation to a broader range of attack scenarios and refining metric selection strategies for different threat types. Beyond statistical and traditional machine learning approaches, further experimentation will explore deep learning architectures, such as autoencoders, to address more complex and subtle attack patterns. These advancements aim to enhance detection capabilities and support the final selection of models to be integrated into the final end-to-end analytics pipeline.

The next steps for the NWDAF-based anomaly detection pipeline focus on its full integration within the UC2 architecture as an end-to-end security analytics component. Specifically, the objective is to integrate the complete pipeline, starting from dataset generation using the Amarisoft 5G testbed, which supports the training and benchmarking of the anomaly detection models. Following the ongoing evaluation of the techniques mentioned in Section 5.4.2 across additional attack scenarios and behavioral patterns, the most suitable models will be finalized for operational deployment. The integrated pipeline will then leverage real-time metric streams from the testbed to enable continuous detection of anomalous behavior, with detected anomalies and alerts made available through the implemented UI dashboard for real-time visualization and monitoring.

6. Holistic Protection with Context and Robustness

This chapter addresses "Task T2.5, *Holistic Protection with Context and Robustness"*. Rather than treating contextual awareness and robustness independently in the frame of other tasks of WP2, T2.5 examines how contextual reasoning and adversarial defense strategies can be jointly leveraged by identifying opportunities where their integration can yield stronger and more adaptive protection mechanisms. In particular, Section 6.1 focuses on concrete opportunities to embed adversarial robustness directly into context-aware CVGL models while Section 6.2 outlines forward-looking research avenues for unifying contextual cues with adversarial defense mechanisms. Finally, T2.5 also focuses on the investigation of which algorithms can be applied and when based on the type of attacks and associated risk levels. To this end, Section 6.3 outlines GuardAI's approach for a scenario-based defense selection strategy tailored for the protection of CAV perception systems.

6.1. Adversarially Resilient CVGL

The Cross-View Geo-Localization (CVGL) scheme estimates accurately the position of agents achieving high situational awareness by reducing the localization error using optimization approaches. Adversarial or spoofing attacks introduce noisy data perturbing and inducing errors to the estimating poses. On this basis, the (CVGL) framework can be positioned as an adversarial-resilient localization scheme by treating cross-view alignment as *a* multimodal consistency check and hardening each input stream before fusion.

CVGL methods that incorporate ground-view images from vehicles or UAV sensor imagery along with the Satellite imagery can be used to mitigate such attacks (see Figure 6.1 and Figure 6.2). Specifically, in the approaches described in Sections 4.1.1 and 4.1.2, can be exposed to adversarial attacks, but dedicated purification (denoising) models are applied to each modality to suppress attack-induced artifacts prior to cross-view matching. The cross-view module then fuses the purified inputs and estimates localization using geometric and contextual agreement between ground/air views and overhead maps, so spurious patterns that fail to remain consistent across viewpoints are less likely to dominate the solution. Evaluating this pipeline under diverse attacks (and selecting the most effective purification/defense per scenario) yields a practical, end-to-end strategy for robust localization in the presence of adversarial or noisy conditions.

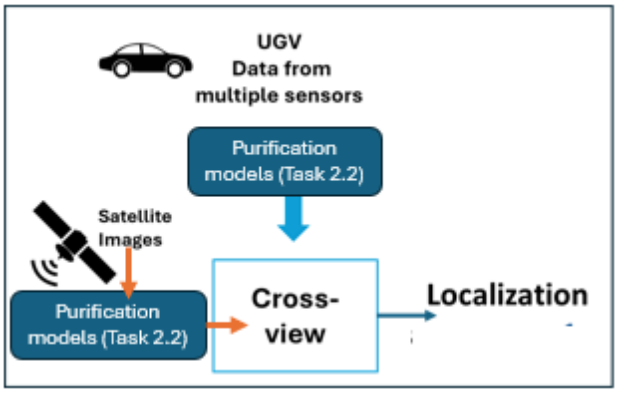


Figure 6.1: Cross-View Geo-Localization with Ground-View and Satellite Images serving as a resilient framework against adversarial attacks.

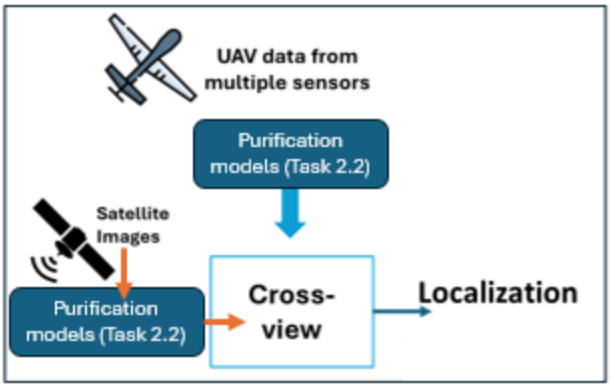


Figure 6.2: Cross-View Geo-Localization with UAV and Satellite Images serving as a resilient framework against adversarial attacks.

6.2. Research Directions for Integrated Defenses

Achieving comprehensive adversarial robustness requires a holistic defense approach (Figure 6.3), moving from isolated components to a coordinated, multi-layered system. The design of such a system involves both sequential and parallel processes, guided by dynamic decision boundaries. The first line of defense involves data purification using diffusion-based methods (Section 3.2), that remove perturbations and "clean" adversarial signals from raw inputs before further processing. After this initial step, multimodal components, such as combining RGB and IR streams (Section 4.3), can operate in parallel to create a more stable and reliable perception, with decision boundaries guiding when to emphasize or switch modalities based on environmental factors or sensor confidence. Additionally, multi-task learning for detection and segmentation runs simultaneously (Section 4.2), providing a more detailed, context-aware scene understanding. Activation of specific components, such as 3D robustness checks (Section 3.5), depends on the availability of spatial data and can be initiated based on confidence thresholds set earlier, ensuring efficient resource use and adaptive defense strategies.

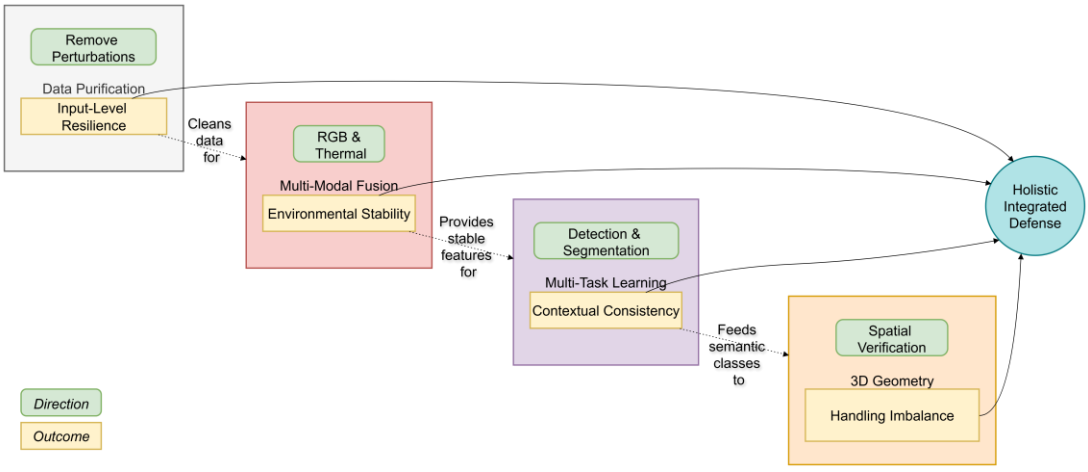


Figure 6.3: Interconnected defense strategies for holistic system robustness.

6.3. Scenario-Driven Defense Selection for Mitigating Adversarial Threats in CAVs

Building on the capabilities of the CAV Adversarial Attack Simulator[97] (introduced in Section 5.1), and also on the activities of tasks T.2.2 – T2.4, in T2.5 we define structured scenarios to construct adversarial attacks targeting object detection and segmentation, and evaluate their impact in multi-modal and multi-agent settings. The objective is to identify how contextual cues and adversarial defense mechanisms can be combined to improve end-to-end resiliency, and to quantify the resulting benefits and computational/operational overheads. In doing so, the overall aim is to assess which (adversarial purification, anomaly detection, and context-aware mitigation) algorithms are most effective when deployed together and, based on the observed risk levels, derive a strategy to guide planning and scheduling decisions on which defenses to activate and when. An example of this matching that is already under investigation is presented in Section 6.1, while an instance from and indicative dataset that has been created concerning adversarially perturbed LiDAR point-clouds, is shown in Figure 6.4.

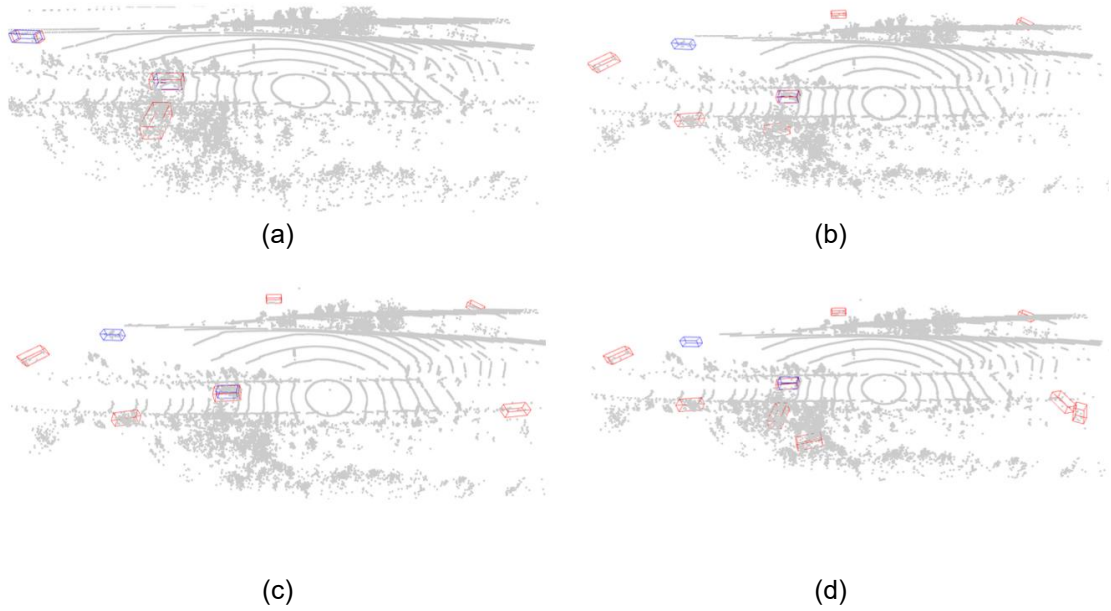


Figure 6.4: Visualization of point clouds under various adversarial attack types. Ground truth 3D bounding boxes are depicted in blue, while predicted bounding boxes are shown in red. (a) Clean input point cloud. (b) Adversarially perturbed input. (c) Point cloud with 300 adversarially attached points. (d) Point cloud after removal of 1% points.

[97] C. Anagnostopoulos et al., "Integrated Simulation Framework for Adversarial Attacks on Autonomous Vehicles", in Proc. of IEEE International Smart Cities Conference, 2025

7. Conclusions

D2.2 consolidates the GuardAI consortium's initial progress on adversarial resilience, Multi-X context awareness, and anomaly detection for robust edge AI in safety-critical settings. A consistent takeaway across the reported components is that dependable operation under attack and operational variability requires layered protection—combining input/model hardening, redundancy-driven consistency checks, and continuous monitoring—rather than relying on any single technique in isolation.

On the adversarial resiliency front, the deliverable demonstrates that substantial robustness gains can be achieved with deployment-feasible designs. Lightweight, optimization-inspired purification (e.g., deep unrolling with domain-specific priors) provides interpretable defenses with low overhead for both RGB and LiDAR pipelines, while efficient diffusion-based purification extends coverage to stronger and more adaptive threat models when compute allows. Complementary analyses and recovery methods address practical physical and patch-based threats and improve generalization across datasets and attack variants.

Multi-X context awareness and anomaly detection complement model-level defenses by using cross-view redundancy and system-level signals to detect inconsistencies and abnormal behavior. The report shows how cross-view geo-localization can both enhance localization robustness and expose GNSS anomalies, how multi-task and multi-modal cues (including IR fallback) can improve runtime trustworthiness, how multi-agent scoring can aggregate evidence into interpretable situational awareness indicators, and how cyber-layer monitoring (IDS and NWDAF analytics) can provide continuous security visibility in 5G-enabled deployments.

Taken together, these outcomes support a holistic protection paradigm in which defenses are selected and composed based on scenario risk, resource constraints, and the type of threat, enabling systems to detect, withstand, and recover from adversarial disruptions while maintaining operational reliability.

Forthcoming steps for the next deliverable and toolkit integration include:

- Broaden evaluation to include adaptive attackers, richer environmental corruptions, and cross-domain generalization, while maintaining comparable baselines and reporting runtime/memory trade-offs.
- Define and implement WP2 toolkit interfaces (inputs/outputs, configuration parameters, latency and compute budgets) to enable plug-and-play integration with perception stacks and project use-case platforms.
- Validate end-to-end pipelines that combine purification, context consistency checks, uncertainty estimation, and anomaly detection, and demonstrate scenario-driven defense selection on representative hardware and operational scenarios.

These steps will mature the techniques documented here into robust, reusable components that improve the security and resilience of edge AI systems in GuardAI's safety-critical application settings.