Glare-Aware Resi-YOLO: Tiny Vessel Detection with Dual-Brain Edge Deployment for Maritime UAVs

Shang-En Tsai * and Chia-Han Hsieh

Posted Date: 27 February 2026

doi: 10.20944/preprints202602.1747.v1

Keywords: maritime UAV; tiny vessel detection; glare suppression; attention module; P2 feature pyramid; YOLO11; multi-object tracking; dual-brain architecture; GSS; SeaDronesSee; Jetson Orin Nano; TensorRT

Article

Glare-Aware Resi-YOLO: Tiny Vessel Detection with Dual-Brain Edge Deployment for Maritime UAVs

Shang-En Tsai * and Chia-Han Hsieh

Department of Computer Science and Information Engineering, Chang Jung Christian University, Tainan City 711, Taiwan

* Correspondence: sean@mail.cjcu.edu.tw; Tel.: +886-6-2785123 (ext. 6154)

Highlights

What are the main findings?

- Resi-YOLO improves tiny-vessel detection under high-glare conditions, increasing AP_{small} by 13.1% over the YOLOv8n baseline.
- The full pipeline runs in real time on Jetson Orin Nano, achieving ~100 ms end-to-end latency and 12.8 FPS (with >30 FPS TensorRT inference).

What are the implications of the main findings?

- Robust tiny-vessel perception can be executed onboard maritime UAVs without cloud dependence.
- Glare Severity Score (GSS)-stratified evaluation and the dual-brain design offer a practical blueprint for safety-oriented deployment under link variability.

Abstract

Maritime UAV perception must reliably detect and track tiny vessels under harsh specular glare. In practice, detection failures are dominated by two coupled factors: (i) vessels often occupy only a few pixels, causing small-object recall collapse, and (ii) sun glint and sea-surface reflections generate over-exposed regions that trigger false positives and unstable associations. This paper presents Resi-YOLO, a system-level pipeline that improves tiny-vessel sensitivity while preserving embedded throughput on a Jetson Orin Nano. At the model level, Resi-YOLO combines a P2-enhanced feature path with an attention-based glare suppression module to strengthen high-resolution semantics and suppress glare-induced artifacts; optional SAHI-style slicing is supported for ultra-high-resolution scenes. At the system level, we adopt a heterogeneous dual-brain deployment, where the Orin Nano performs primary inference and an MCU-based safety-island tracker mitigates delay/jitter via time-stamped measurement replay and IMM-UKF updates. We further define a Glare Severity Score (GSS) to stratify evaluation by illumination intensity for transparent robustness reporting beyond average mAP. Experiments on maritime detection and tracking sequences demonstrate consistent improvements over YOLO baselines in tiny-object regimes and high-glare conditions, while sustaining real-time operation with approximately 100 ms end-to-end latency on the Orin Nano under TensorRT FP16 deployment.

Keywords: maritime UAV; tiny vessel detection; glare suppression; attention module; P2 feature pyramid; YOLO11; multi-object tracking; dual-brain architecture; GSS; SeaDronesSee; Jetson Orin Nano; TensorRT

 doi:10.20944/preprints202602.1747.v1

1. Introduction

1.1. Background and Motivation

Maritime monitoring supports search and rescue (SAR), illegal fishing enforcement, and coastal security. UAVs offer rapid deployment and flexible sensing over satellites or manned aircraft. However, maritime UAV video exhibits unique challenges: sun glint, haze, sea-spray, and dynamic backgrounds (whitecaps, wakes), invalidating terrestrial benchmarks. FPV drones add high-mobility threats, requiring robust detection and tracking.

1.2. Key Challenges

We address: (i) tiny targets (<32×32 pixels after resizing); (ii) glare and brightness clutter mimicking vessel features; (iii) deployment gap—high-accuracy models demand GPU resources often unsustainable for edge platforms; (iv) asynchronous sensor delays and non-linear motions in tracking; (v) SWaP (Size, Weight, and Power) constraints for long-endurance operations.

Marine-Engineering Perspective. Unlike land-based UAV vision benchmarks, maritime operations tightly couple vision-based perception with time-varying over-water communication links and mission safety. Over-the-sea channels can exhibit rapid two-/three-ray fading and evaporation-duct effects, intermittently collapsing throughput and increasing packet loss and jitter [41]. Recent maritime computer-vision workshops further consolidate benchmarks, evaluation protocols, and open challenges for over-water detection and tracking, providing a common ground for comparing system-level robustness under real deployment constraints [16]. When a high-rate video stream is transported via RTSP/RTP, buffering and retransmission behaviors (especially over TCP) can induce a latency cliff—or even a deceptively smooth but seconds-delayed feed (bufferbloat)—breaking the perception–control loop [23–25]. Therefore, we formulate Resi-YOLO (Resilient YOLO) as an integrated perception-and-safety subsystem: glare-aware attention and P2-enhanced features improve tiny-vessel observability, while an MCU safety-island and time-stamped measurement replay (TSMR) maintain bounded-latency decision-making and enable explicit robustness certification under delay/jitter/dropout in Table 6.

To quantify network-induced instability, we explicitly measure delay and jitter effects along the perception pipeline. As shown in Table 7, nominal network transfer latency is about 5 ms (p95 ≈ 10 ms) under stable links, but can abruptly exceed 100 ms under bufferbloat and retransmission effects, exhibiting a cliff-like latency inflation rather than gradual degradation [38]. Such conditions cause conventional single-brain pipelines to lose temporal alignment and tracking continuity almost immediately, whereas the proposed dual-brain architecture maintains bounded-latency state continuity via an MCU safety-island with time-stamped measurement replay (TSMR), providing engineering evidence that this design is necessary for reliable maritime operation under unstable communication links.

1.3. Contributions

- **Enhanced Resi-YOLO Architecture:** A YOLO11n-based detector augmented with a P2 detection head, CBAM attention, NWD loss, and optional SAHI slicing to improve tiny-vessel recall under sea clutter and strong sun glint.
- **MCU Safety-Island Dual-Brain Architecture:** A heterogeneous perception–navigation pipeline where a Jetson Orin Nano performs deep perception while an MCU safety island maintains deterministic tracking and minimal navigation cues (e.g., IMM-UKF state estimation and lightweight planning). This design explicitly targets marine-engineering constraints such as intermittent links, abrupt exposure changes, and GPU workload spikes by decoupling high-throughput, non-deterministic vision processing from safety-critical control loops.

 doi:10.20944/preprints202602.1747.v1

- **System-Level Implementation:** A waterproof UAV platform (Pixhawk-class autopilot + stabilized 4K gimbal camera), TensorRT deployment on Jetson Orin Nano, and a latency-aware data bus (e.g., RTSP/MQTT/WebSocket) for edge-to-ground integration.
- **Evaluation Blueprint for Marine Operations:** Stratified evaluation across glare severity (GSS), target-size bins, and video I/O impairments (delay/jitter/dropout), accompanied by reproducibility templates, command/parameter logs, and deployment checklists.

As illustrated in Figure 1, maritime UAV detection is simultaneously challenged by tiny targets, sun-glint/whitecaps, and motion blur, which can severely degrade both localization and confidence estimation. The following contributions are designed to address these coupled failure modes from the model, system, and evaluation perspectives.

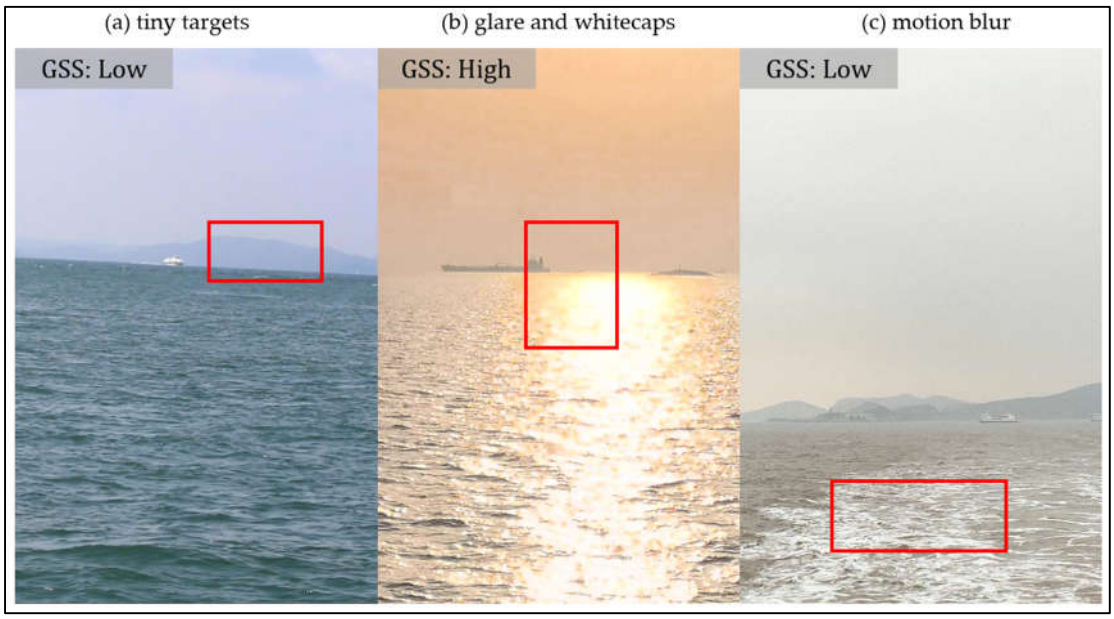


Figure 1. Representative maritime scenes used for qualitative evaluation under different failure drivers: (a) tiny targets at long range, (b) glare and whitecaps, and (c) motion blur. Each panel reports the Glare Severity Score (GSS) and glare-area ratio to contextualize illumination conditions.

2. Related Work

2.1. Vision-Based Maritime UAV Perception Under Sea Glare

Maritime drone benchmarks like SeaDronesSee [1,18] highlight that sun glint and sea clutter create unique failure modes. While lightweight detectors such as YOLOv7-sea [19], YOLOv4 [11], and other UAV-specific networks [20] offer speed-accuracy balances, and enhancement pipelines [21] target glint removal, our work shifts toward a system-level, dual-brain architecture. This ensures tracking continuity even during GPU or communication failures.

2.2. Attention and Loss Mechanisms for Clutter Suppression

Detecting tiny vessels requires specialized feature extraction and clutter rejection. Canonical designs like FPN [17] and BiFPN [18] improve multi-scale recall, while CBAM attention [5] suppresses specular highlights. Localization sensitivity is further addressed via NWD loss [6] and SAHI slicing [7]. Recent architectures like S3Det [42], YOLOv12n/11-Pico, and Mamba YOLO [33] improve efficiency, while YOLOv8 [38] and C3-family enhancements [39] serve as deployment baselines. Resi-YOLO distinguishes itself by coupling these architectural gains (P2, CBAM, NWD, SAHI) with a heterogeneous design to decouple perception uncertainty from tracking.

 doi:10.20944/preprints202602.1747.v1

2.3. Reliability-Aware Perception and Geometric Filtering in Maritime Vision

Beyond appearance-based detection, reliability-aware perception mitigates visual artifacts in challenging environments. In stereo-based maritime vision, depth reliability maps from SGBM [36] identify unstable reflections to enable confidence-guided filtering. Similarly, Binary Line Segment Filtering (BLSF) [37] suppresses structured clutter like wave crests by penalizing detections with linear patterns. Resi-YOLO integrates these reliability and geometric cues to enhance robustness under glare-dominated conditions.

2.4. Heterogeneous Architectures and Edge-Cloud Systems

Standard tracking pipelines (e.g., BoT-SORT, ByteTrack, SORT) [4,8,22] and metrics like HOTA [8,9] provide the foundation for maritime MOT. However, to meet marine-engineering reliability standards, we introduce an MCU safety-island. This maintains deterministic tracking continuity during video I/O impairments (delay/jitter/dropout) that typically degrade standard edge-AI branches.

3. Proposed Method: Enhanced Resi-YOLO with Dual-Brain Integration

3.1. Overview

The Resi-YOLO framework represents a holistic approach to maritime perception, bridging the gap between high-complexity deep learning models and the deterministic requirements of flight safety. Our methodology is structured into three core pillars: (i) model-level architectural enhancements for tiny-target recall and glare suppression, (ii) system-level heterogeneous integration for latency decoupling via a dual-brain paradigm, and (iii) reliability-guided strategies to mitigate optical artifacts. This section details the theoretical formulation of these components and explains how they are synthesized into a fail-operational UAV perception pipeline designed for high-glare maritime environments.

3.1.1. Stress-Test Corpus versus Benchmark-Scale Validation

Resi-YOLO augments YOLO11n with a P2 detection head, CBAM attention, NWD loss, and optional SAHI slicing. Under the proposed dual-brain paradigm, high-throughput perception runs on the edge AI node, while deterministic tracking and minimal navigation cues are maintained by an MCU-based safety island as Figure 2.

 doi:10.20944/preprints202602.1747.v1

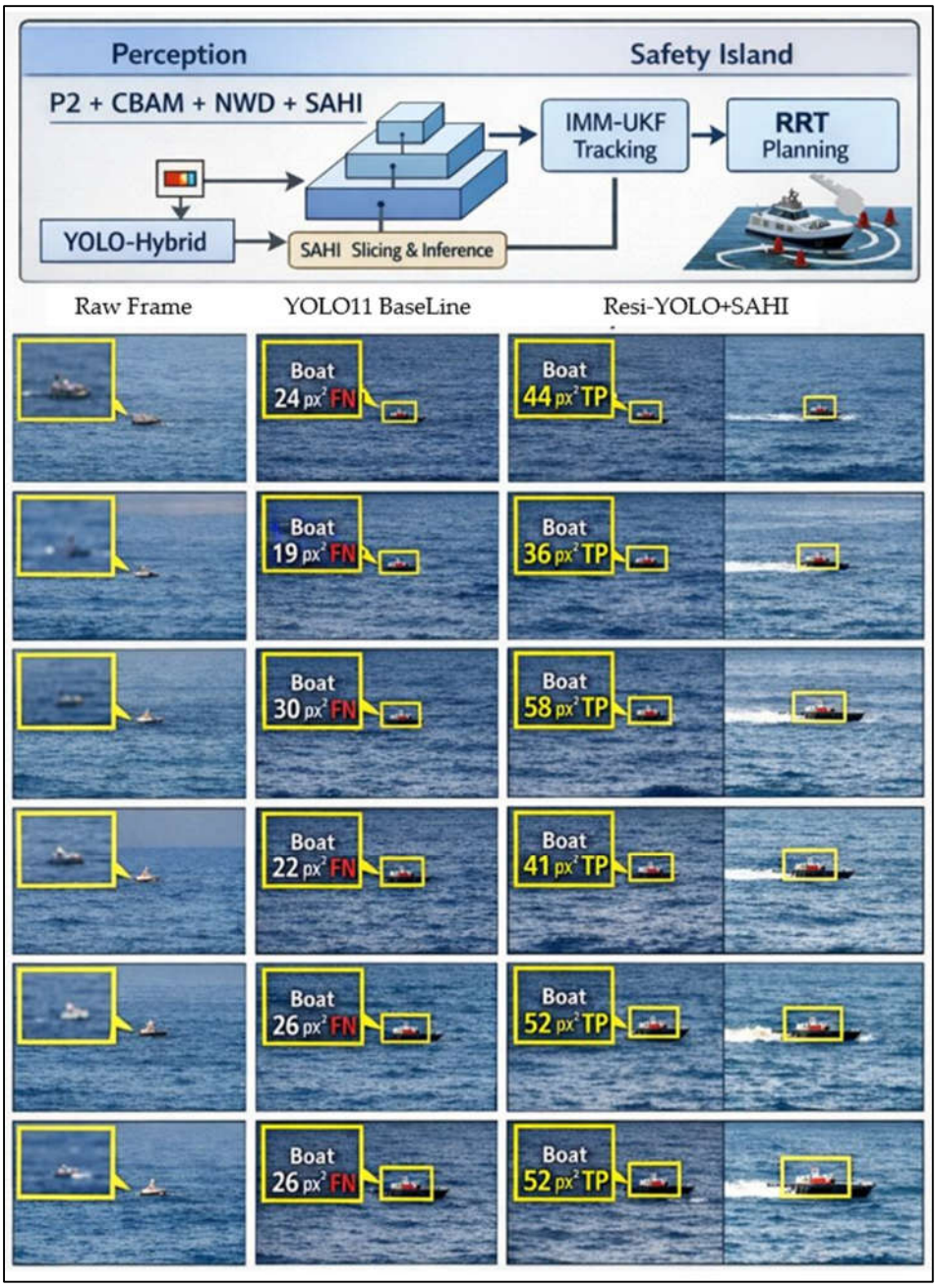


Figure 2. The upper diagram illustrates our proposed decoupled dataflow between the high-throughput perception stage (Jetson Orin Nano) and the deterministic safety island (MCU-based IMM-UKF tracking and RRT planning). The lower panels provide a qualitative comparison between the baseline and Resi-YOLO, showing the successful detection (True Positives, TP) of tiny vessels (19–58 px^2) that were missed (False Negatives, FN) by the baseline.".

3.2. P2 Detection Head and NWD Loss for Tiny Vessels

To enhance the detectability of tiny vessels, we introduce a P2 detection head that fuses high-resolution shallow features with deeper semantic cues. As formulated in Equation (1), the P2 feature map is constructed by upsampling the P3 feature and merging it with the corresponding backbone C2 feature, thereby preserving fine spatial details at a stride of 4:

$$F_{P2} = \text{Conv}(\text{Upsample}(F_{P3}) \oplus F_{C2}) \quad (1)$$

Here, F_{P3} denotes the P3 feature (stride 8) from the top-down pathway, and F_{C2} denotes the corresponding backbone C2 feature (stride 4). Compared with the default detection head operating

 doi:10.20944/preprints202602.1747.v1

at stride 8, the P2 head preserves finer spatial sampling, which is critical for targets spanning only a few pixels (Figure 3). Specifically, we upsample the P3 feature and fuse it with the corresponding backbone C2 feature to form P2, recovering contextual cues while retaining high-resolution details (stride 4).

For localization, we adopt the Normalized Wasserstein Distance (NWD) loss by modeling each bounding box as a 2D Gaussian distribution. Let W_2 denote the 2-Wasserstein distance between the predicted and ground-truth Gaussians; we define *NWD* as in Equation (2), and minimize the corresponding loss $L_{NWD} = 1 - NWD$.

$$NWD = \exp\left(-\frac{W_2}{C}\right) \tag{2}$$

This formulation provides stable gradients for tiny or non-overlapping boxes where IoU-based objectives become unstable, facilitating convergence in the tiny-object regime and improving box stability under far-range and glare-degraded conditions. In practice, *C* is set following prior work to normalize the distance scale.

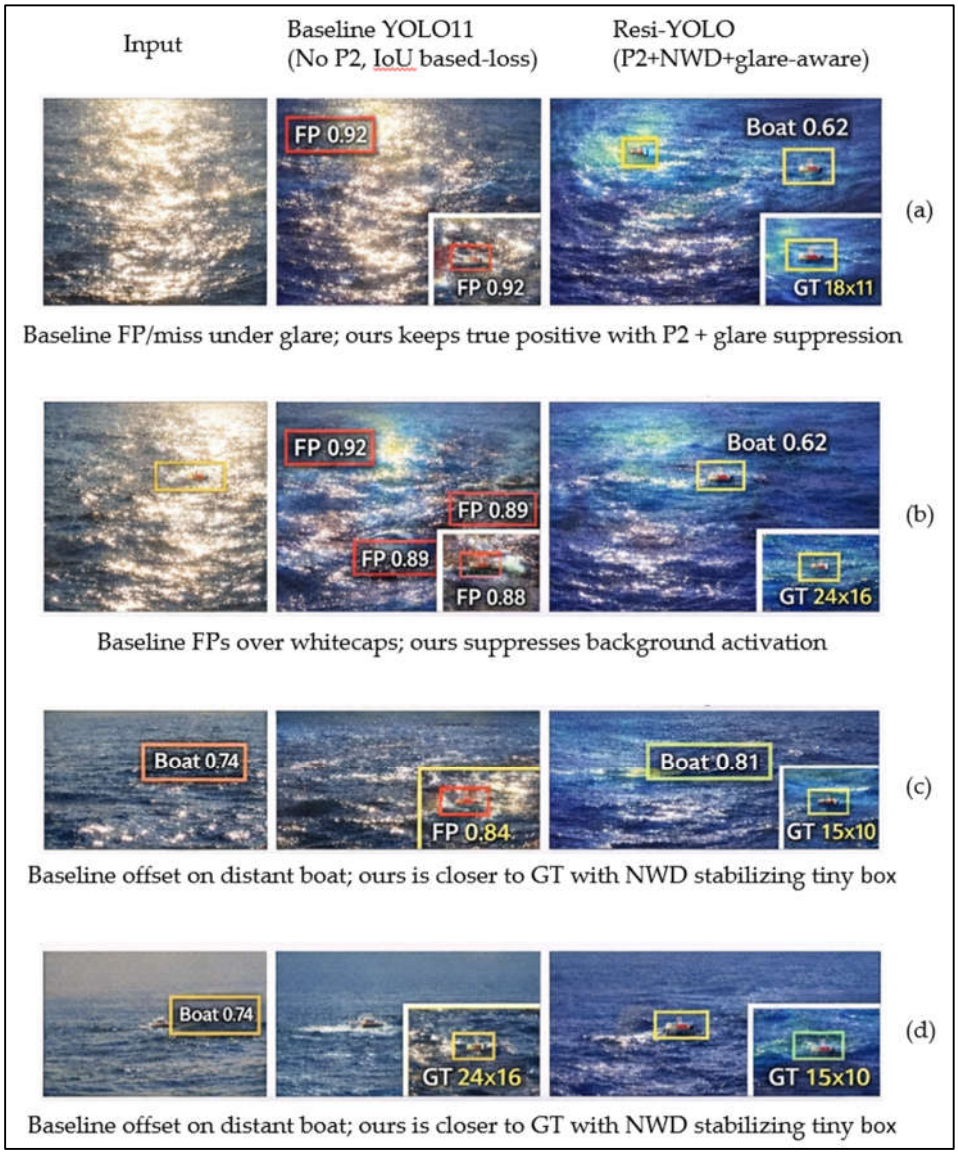


Figure 3. Qualitative examples showing the effect of the P2-enhanced feature path and NWD-based localization on tiny-vessel detection. Compared with the baseline, Resi-YOLO reduces missed tiny targets and suppresses glare-driven false positives in high-GSS scenes. Insets provide zoomed views for sub-32-px vessels.

 doi:10.20944/preprints202602.1747.v1

3.3. CBAM for Glare Suppression

To suppress glare-induced feature dominance and sensor saturation [38], we adopt the Convolutional Block Attention Module (CBAM) to reweight both channel-wise and spatial responses. CBAM mitigates the collapse of fine texture cues by learning to suppress glare-dominated activations while reallocating capacity to structured vessel contours.

As defined in Equation (3), the Channel Attention module emphasizes informative feature maps through pooled global descriptors and Spatial Attention module, defined in Equation (4), further refines localization by highlighting salient regions and attenuating high-brightness background highlights (Figure 4):

$$\text{CBAM: Channel } M_C(F) = \sigma\left(MLP\left(AvgPool(F)\right) + MLP\left(MaxPool(F)\right)\right) \quad (3)$$

$$\text{Spatial } M_s(F) = \sigma\left(f^{2x2}([AvgPool(F); MaxPool(F)])\right) \quad (4)$$

This joint modulation stabilizes feature representations under strong reflections and sea clutter, consistent with recent attention-assisted maritime detection frameworks [39].

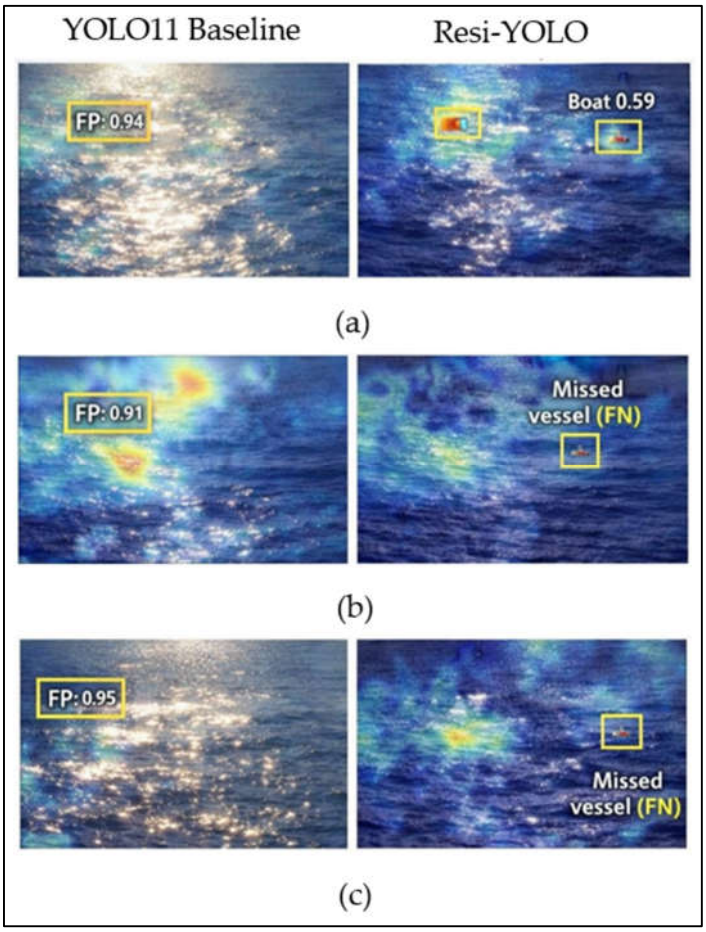


Figure 4. Visualization of the CBAM spatial attention map $M_s(F)$ (Equation (4)) under hard-glare conditions. The heatmaps depict post-sigmoid spatial attention weights, showing suppressed activation over specular highlights and increased emphasis on vessel-like structures.

3.4. SAHI for High-Resolution Inference

To improve tiny-vessel recall in long-range maritime scenes, we optionally adopt a SAHI-style tiling strategy for high-resolution inputs (e.g., 4K frames). Each frame is divided into overlapping patches, resized to the detector input resolution, and processed independently. Detections are mapped back and merged via NMS to remove duplicates.

 doi:10.20944/preprints202602.1747.v1

Tiling increases the effective spatial resolution and mitigates the disappearance of sub-32 px targets after global downsampling, at a predictable computational cost (throughput decreases with patch count). Therefore, SAHI is enabled only for long-range monitoring or when scenes are dominated by tiny vessels.

3.5. Reliability-Guided Fusion (RGF) and Binary Line Segment Filter (BLSF)

Maritime glare and wave reflections often produce saturated regions and elongated streaks, degrading feature reliability. We introduce a Reliability-Guided Fusion (RGF) mechanism combined with a Binary Line Segment Filter (BLSF).

RGF leverages a Depth Reliability Map (DRM) derived from stereo disparity estimation using SGBM, following the reliability modeling in [37]. Regions with unstable disparity are treated as low-confidence. Because glare frequently causes depth failure, DRM acts as a spatial weighting matrix that suppresses unreliable features before subsequent detection layers. This intermediate fusion attenuates glare artifacts prior to attention modules such as CBAM.

In addition, a lightweight BLSF suppresses geometrically implausible linear clutter (e.g., wave crests, horizon streaks). BLSF penalizes detections dominated by elongated, low-compactness structures rather than vessel-like blobs. We adopt the core formulation from [36] and integrate it as a post-filter within the maritime pipeline to reduce glare-driven false positives while preserving tiny-vessel recall. Algorithmic details are provided in [36,37]; here we focus on system-level integration.

3.6. Sensitivity Analysis and Geometric Filtering

In marine operations, sea-surface reflections create strong, time-varying multipath. Depending on platform height and range, signals follow two- or three-ray interference patterns, with deep fades when direct and reflected components cancel [22]. These link-level fluctuations manifest as bursty packet loss and rapidly varying bitrate at the application layer.

A key pitfall is that "usable video" does not guarantee "usable control." RTSP over TCP or UDP/RTP can accumulate delay under congestion; interactions between reliability mechanisms and buffering may cause freeze (head-of-line blocking) or stale frames due to bufferbloat [23–25]. Given finite control stability margins, this leads to a latency cliff where perception feedback becomes unsafe.

The dual-brain architecture mitigates this by: (i) running detection on the Jetson as a high-throughput perception brain, (ii) maintaining a deterministic MCU safety island that rejects stale detections and propagates state during dropouts, and (iii) providing a metadata-first low-bandwidth channel for situational awareness. Section 6.3 quantifies robustness using TSMR via controlled delay/jitter/dropout injection and tracking evaluation (Table 6).

To decouple non-deterministic vision latency from time-critical control, the MCU implements a safety-island tracker with a ring buffer of time-stamped detections and inertial priors. Late or out-of-phase measurements are handled through time-stamped measurement replay (TSMR) with IMM-UKF correction and forward re-propagation. For relative target speed v, temporal misalignment Δtinduces spatial association error $\varepsilon \approx v\Delta t$, which may exceed the gating radius for tiny vessels and cause identity switches. TSMR reduces association fragmentation and ensures deterministic tracking outputs.

Figure 5 illustrates the separation between the non-deterministic perception pipeline and the deterministic safety-critical control loop. While the Jetson maximizes situational awareness under nominal conditions, the MCU safety island guarantees fail-operational navigation under degraded latency or communication reliability.

3.7. MCU Safety-Island RRT Planning and MAVLink Commanding

In addition to deterministic tracking (TSMR + IMM-UKF), the safety-island MCU also provides a lightweight obstacle-avoidance planning loop to maintain conservative navigation when the edge-AI branch becomes delayed. At each planning tick (e.g., 10–20 Hz), the MCU queries the most recent

 doi:10.20944/preprints202602.1747.v1

state estimat $(\hat{x}, \boldsymbol{P})$from the IMM-UKF and constructs a local safety representation (e.g., a coarse 2D occupancy/cost map in the NED frame) using (i) predicted target states from the tracker, (ii) predefined keep-out zones (geo-fence), and (iii) short-term motion constraints from the autopilot. A bounded-iteration RRT planner is then executed with a fixed compute budget (max nodes / max iterations) to generate a collision-free waypoint sequence over a short receding horizon.

The resulting path is converted into a small set of smoothed waypoints (e.g., via shortcutting or spline fitting), and the MCU streams these as MAVLink setpoints (e.g., SET_POSITION_TARGET_LOCAL_NED or mission-waypoint updates) at a command rate aligned with the autopilot control loop (e.g., 20–50 Hz). If the perception branch reports degraded vision (stale or missing detections beyond a time threshold) the MCU freezes the last verified-safe path and switches to a conservative "hold/loiter" policy until reliable updates resume.

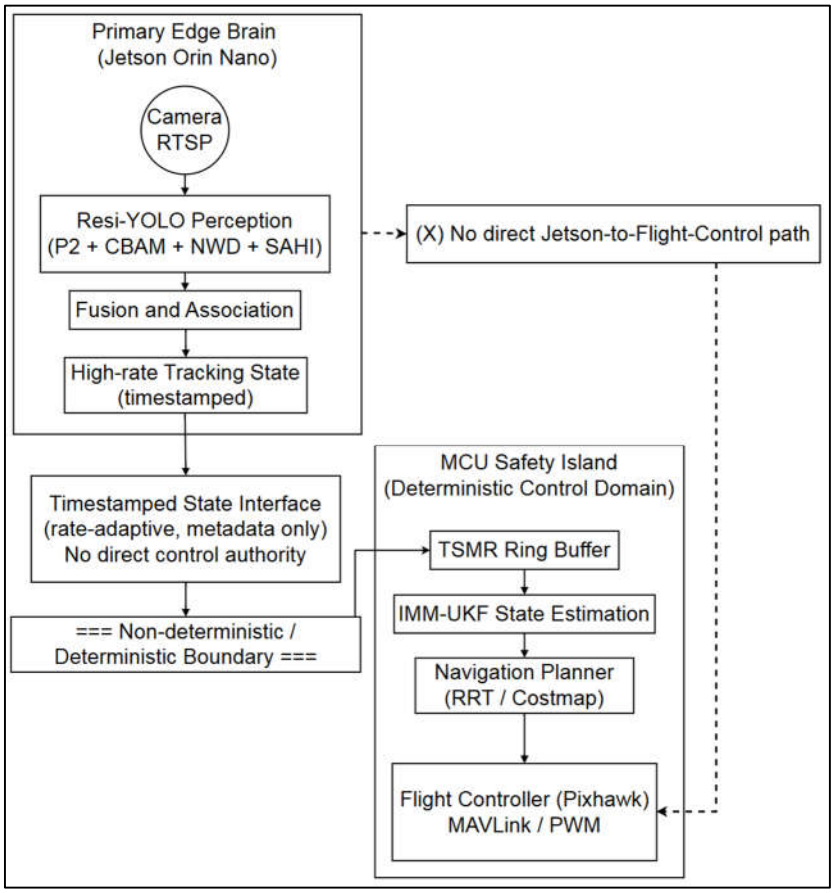


Figure 5. Conceptual dual-brain dataflow from perception to navigation under link variability. The Jetson branch performs high-rate perception and association, while the MCU safety-island maintains time-stamped replay and state continuity to support degraded operation. The diagram highlights the separation between perception throughput and safety-critical continuity.

4. System Implementation: Maritime UAV–Edge–Cloud Pipeline

To ensure that algorithmic gains translate to mission utility, we implement an end-to-end pipeline including a custom waterproof UAV platform, an embedded edge node for inference, and a cloud dashboard for logging and visualization.

4.1. Custom Waterproof UAV Platform

We adopt a fully programmable custom UAV built around an open autopilot and a stabilized IP/RTSP camera interface for direct edge ingestion.

 doi:10.20944/preprints202602.1747.v1

Engineering rationale: Maritime operations frequently experience intermittent communication, specular-glare–induced vision latency cliffs, and constrained SWaP-C. Accordingly, we deploy a dual-brain architecture where the Jetson Orin Nano performs perception and publishes detections, while a Safety Island MCU maintains deterministic tracking and can issue conservative navigation commands to the Pixhawk via CAN when GPU inference or the ground link becomes unreliable. Supplementary Table S1 summarizes the deployed hardware stack. The Jetson Orin Nano was chosen as the low-cost edge-AI baseline because it delivers up to 40 TOPS (INT8) within a 15 W power envelope, representing an order-of-magnitude uplift over the 2019-era Jetson Nano while remaining compatible with UAV SWaP-C constraints [1,3]. This computing headroom allows us to deploy advanced features (e.g., the P2 layer and attention modules) and even concurrent models without sacrificing real-time throughput [4], bridging the edge deployment gap identified in Section 1.2. [1,3,4].

To ensure performance stability across extended maritime patrols, we explicitly address the thermal and power implications of operating the Jetson Orin Nano across its 15 W Max-P and 25 W MAXN/Super Mode envelopes. In our UAV integration, the edge computer is mounted on a dedicated heat-spreading structure with active airflow, and GPU/CPU clocks are locked to avoid frequency oscillations and latency spikes caused by thermal throttling. When operating in Super Mode for bandwidth-intensive perception workloads, power draw and junction temperature are continuously monitored, and non-critical workloads are adaptively throttled to maintain sustained real-time performance throughout long-duration missions. This thermal-aware power management strategy aligns with recent edge-robotics guidance on multi-model execution under tight TDP constraints and is critical for preserving deterministic behavior in safety-relevant UAV operations [40].

NVIDIA positions Jetson Orin Nano as a practical entry-level edge-AI platform that bridges real-time deployment needs with compact power envelopes, aligning with the onboard perception requirements of maritime UAV systems [26]. To motivate the choice of the embedded deployment platform, Supplementary Table S2 compares the key specifications of NVIDIA Jetson Nano and Jetson Orin Nano that are most relevant to real-time maritime UAV perception. The substantial differences in GPU architecture, memory bandwidth, and AI compute capability explain the improved throughput and latency headroom observed on Jetson Orin Nano, which is therefore selected as the primary target platform for Resi-YOLO deployment in this work.

The hardware integration of the UAV platform is illustrated in Figure 6, which shows the top and side views of the autopilot, Jetson Orin Nano edge computer, camera–gimbal payload, power distribution, and telemetry modules. The annotated layout highlights module placement, cable routing, and typical power consumption, providing context for the system's non-propulsion power budget and deployment feasibility under marine operating conditions.

4.2. Dual-Brain Link Rate and Packet Definition

The Jetson–MCU link is implemented over UART and transmits a compact state packet at a fixed rate f_{link} (e.g., 50–100 Hz). Each packet includes: (i) a monotonic sequence ID, (ii) a source timestamp t_k (in microseconds), (iii) the active track ID(s), (iv) target kinematics (e.g., (x, y) position and (v_x, v_y) velocity in the local frame), and (v) a quality/reliability flag (e.g., confidence score, GSS regime, or a "vision-degraded" bit). The MCU maintains a ring buffer of the most recent $N = \lceil f_{\text{link}} \cdot T_{\text{buf}} \rceil$ packets with $T_{\text{buf}} = 1$ s (e.g., 100 entries at 100 Hz), enabling time-stamped measurement replay (TSMR) to compensate for delay/jitter and to propagate state estimates through short perception outages without breaking the safety loop. This explicit link-rate/buffer design provides an engineering guarantee that the system can tolerate at least 1 s of short-term visual degradation while preserving bounded-latency decision-making.

 doi:10.20944/preprints202602.1747.v1

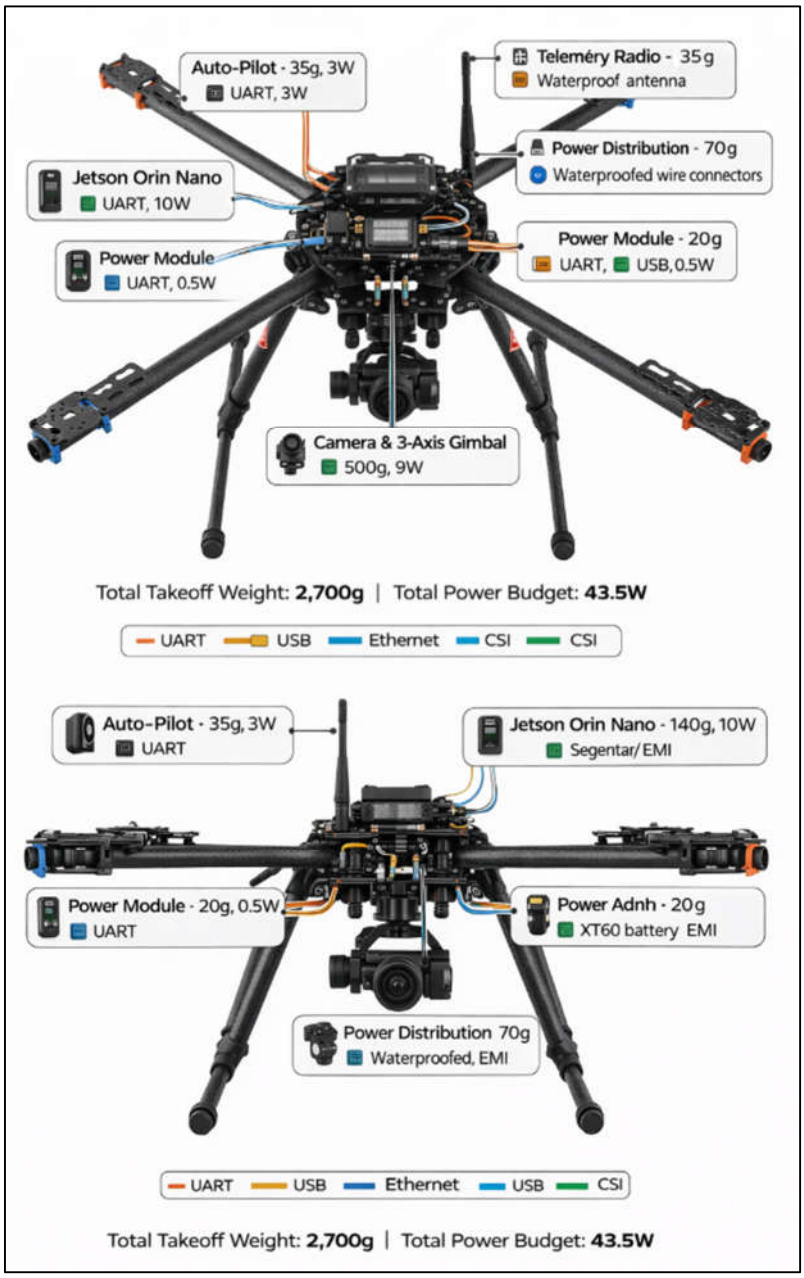


Figure 6. Hardware integration of the UAV platform. Top and side views show the placement of the flight controller (Pixhawk), Jetson Orin Nano edge computer, camera–gimbal payload, power distribution, and telemetry modules. Callouts summarize module weight and nominal power consumption, along with representative data/power interfaces and cable routing. Practical deployment considerations (e.g., waterproofing and EMI mitigation) are indicated where relevant.

4.3. Power, Signal, and Time Synchronization

To ensure repeatable performance under marine vibration and long-duration missions, we explicitly document (i) power distribution, (ii) signal paths, and (iii) time synchronization for consistent cross-module logging and latency attribution. Power distribution. The onboard battery feeds dedicated regulators/BEC rails for (a) the edge node, (b) the autopilot/MCU safety island, and (c) the camera and network interface. To prevent transient brownouts from propagating across modules, each rail is decoupled with local bulk capacitance and protected by undervoltage/overcurrent safeguards. This separation helps avoid vision pipeline resets during aggressive maneuvers and reduces timing jitter induced by power instability.

 doi:10.20944/preprints202602.1747.v1

Signal and time synchronization. The camera stream is delivered to the edge node via Ethernet (RTSP), while the autopilot state and control channels are exchanged via MAVLink (UART) and the safety-island interface. For consistent logging across the dual-brain pipeline, the edge node maintains a synchronized clock (e.g., NTP) and records timestamps at (1) stream ingestion/decoding, (2) inference output, and (3) publish/telemetry transmission. The MCU similarly timestamps received packets using its local clock and stores sequence IDs to support time-aligned replay (TSMR). Together, these measures enable unambiguous separation between onboard L_{edge} and full streaming L_{e2e} latency when analyzing delay/jitter events. These implementation details ensure that the reported robustness (delay/jitter tolerance and glare-aware reliability) is attributable to the proposed architecture rather than incidental integration artifacts.

4.4. Low-Bandwidth Messaging

To accommodate weak maritime links, the edge node publishes compact detection events via MQTT. The backend subscribes to relevant topics, stores logs in a database (e.g., PostgreSQL/MongoDB), and pushes live updates to the frontend via WebSocket.

To address the constraints of maritime communication links, we design a compact and event-driven messaging scheme that decouples detection alerts, system health monitoring, and visual verification. Lightweight MQTT topics are used for high-priority alerts and status updates, while WebSocket streams are activated only when operator confirmation is required. The detailed messaging interfaces and payload design are summarized in Table 1.

Table 1. Messaging schema for maritime detection events used in the dual-brain UAV deployment. All interfaces are designed for asynchronous, event-driven communication to minimize bandwidth while preserving real-time safety semantics.

Interface	Topic/Endpoint	Key fields	Notes (rate/QoS/payload)
MQTT	uav/alert	timestamp (UTC), uav_id, class (swimmer/boat), conf, bbox [x,y,w,h], geo [lat,lon,alt]	Event-driven (max 5 Hz); QoS 1; JSON payload ~300 bytes; ultra-low bandwidth.
MQTT	uav/status	batt_volt, link_quality, glare_idx, system_temp	1 Hz; QoS 0; health/status monitoring; asynchronous publish (non-blocking).
WebSocket	/ws/keyframe	image_base64 (JPEG), detection_id	0.2–1 Hz; transmit keyframes only when a detection requires operator confirmation (~50–100 KB per image).
MAVLink	OBSTACLE_DISTANCE (custom)	distance, angle, sensor_type	2 Hz; UAV publishes basic obstacle distances (if needed for AP).

Note: Additional fields for confidence and track ID can be included in MQTT messages if required by the ground station.

Figure 7 provides a functional overview of the end-to-end dataflow, highlighting how high-rate video streams are processed locally on the edge node and converted into low-bandwidth, event-driven messages suitable for unreliable maritime links. The figure emphasizes system architecture, data-rate asymmetry, and the separation between onboard perception and backend visualization.

 doi:10.20944/preprints202602.1747.v1

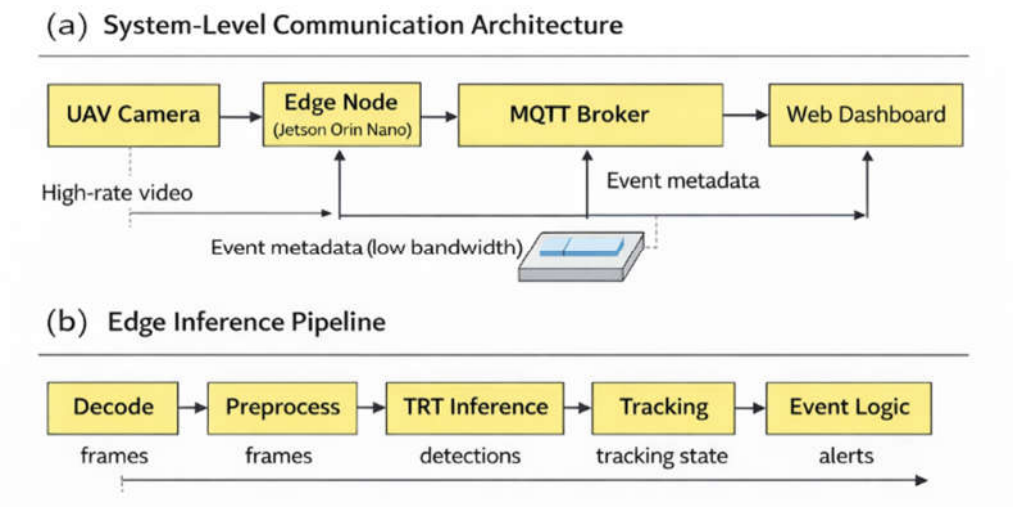


Figure 7. (a) System-level overview of the maritime UAV perception pipeline. High-rate video streams are processed onboard the Jetson Orin Nano edge node, while only low-bandwidth event metadata are transmitted to the backend via MQTT, reducing communication load and improving robustness under maritime link variability. (b) Onboard processing pipeline on the Jetson Orin Nano, including video decoding, preprocessing, TensorRT inference, multi-object tracking, and risk-aware event logic. All perception and alert generation are executed locally, enabling low-latency operation independent of network conditions.

5. Experimental Protocol

5.1. Datasets and Splits

Use a hybrid dataset strategy: (i) public maritime UAV datasets (e.g., SeaDronesSee) for comparability and (ii) an in-house coastal UAV dataset collected under glare-heavy conditions for deployment realism. Report resolution, number of frames/images, annotation counts, and the proportion of tiny objects.

A hybrid dataset strategy is employed to evaluate both generalization and real-world robustness. Specifically, a public maritime UAV dataset is used for benchmarking, while an in-house coastal UAV dataset captured under severe glare conditions is introduced to assess deployment-oriented performance. Dataset statistics, including resolution, data splits, tiny-object ratios, and glare severity, are reported in Table 2.

Table 2. Dataset summary and split configuration.

Dataset	Src	Res	Count	Split (tr/va/te)	Tiny ratio	Glare
SeaDronesSee v2	Pub.	~4K	~14,227 (Tr 8,930 / Va 1,547 / Te 3,750)	63 / 11 / 26%	VH (~91%)	Med. (natural)
In-house UAV	Ours	4K (3840×2160)	~2,500	70 / 20 / 10%	Med (~60%)	Sev. (glare+whitecap)

5.2. Training Recipe and Glare-Oriented Augmentations

Report both standard YOLO augmentations (Mosaic, MixUp, HSV jitter) and glare-specific augmentations (synthetic saturation patches, contrast-limited transforms). All YOLO baselines and model variants are built upon the official Ultralytics codebase, ensuring consistent training and inference pipelines across comparisons [12]. Provide exact probabilities and parameter ranges, and ablate augmentation-only effects when possible. Figure 8 illustrates training-time glare augmentation, while Figures 1 and 4 focus on test-time failure modes and attention behavior.

 doi:10.20944/preprints202602.1747.v1

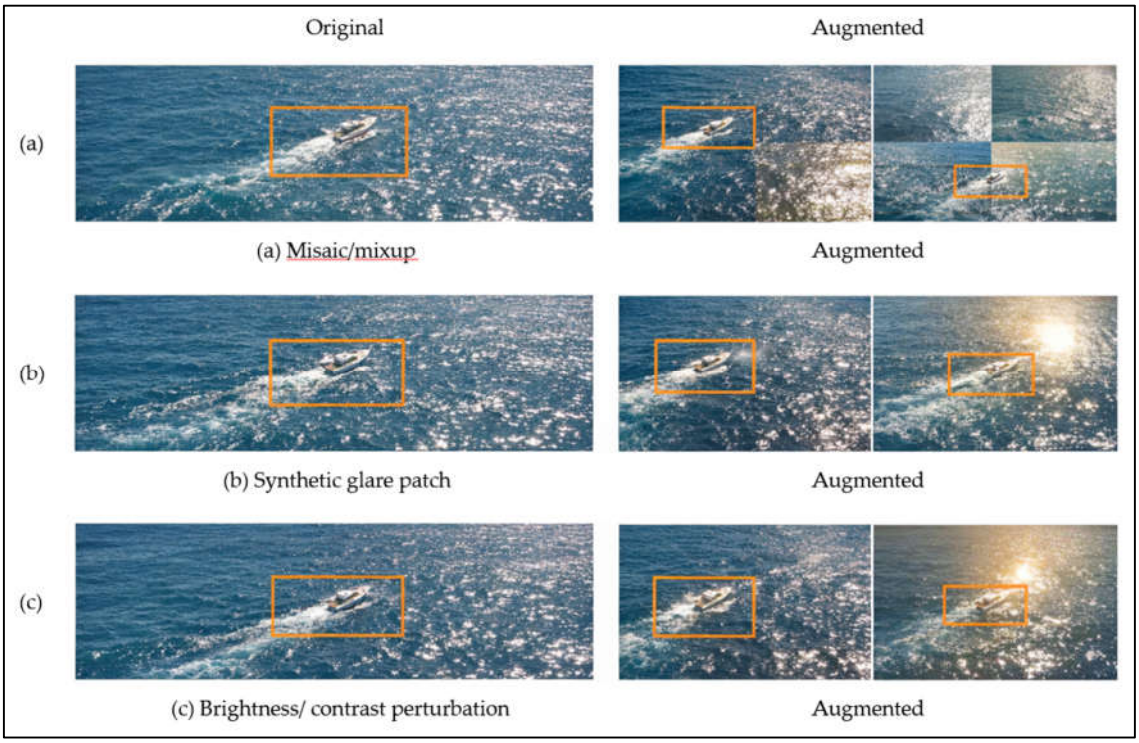


Figure 8. Examples of glare-oriented data augmentations used during training. (a) Mosaic/MixUp increases scale and context diversity while preserving target annotations. (b) Synthetic glare patches simulate specular highlights to improve robustness under high-illumination regions. (c) Brightness/contrast perturbations expand exposure variability for maritime scenes.

5.3. Metrics: Accuracy, Robustness, and Efficiency

In addition to standard mAP, we recommend failure-mode metrics as summarized in Table 3: (i) AP_{small} and $Recall_{small}$ computed on objects below a defined pixel area threshold after resizing; (ii) FPI_{glare} on glare-heavy subsets; and (iii) embedded efficiency metrics including mean and p95 end-to-end latency and steady-state FPS on Jetson Orin Nano (batch=1) after warm-up.

To quantify the radiometric clutter induced by specular reflections, we define the **Glare Severity Score (GSS)**. For each frame, the image is converted from BGR to the HSV color space. Glare pixels (P_{glare}) are identified based on a high value (V) and low saturation (S) threshold, specifically $V \geq 217, S \leq 38$. The GSS is defined as the ratio of these glare pixels to the total number of pixels (P_{total}) within the predefined Region of Interest (ROI):

$$GSS = \frac{\sum_{i=1}^{n} P_{\text{glare},i}}{P_{\text{total}}} \quad (5)$$

This score, ranging from 0 to 1, allows for the stratification of performance analysis under varying illumination intensities."

Table 3. Evaluation metrics and operational meaning.

Metric	Recommended definition	Operational meaning
AP_{small} / $Recall_{small}$	Compute on objects with bbox area < 32×32 after resize (or define bins)	Tiny-vessel sensitivity
FPI_{glare}	#False positives per image on glare-heavy subset	Operator workload / false-alarm control
Latency (mean / p95)	Per-frame time: decode + preprocess + infer + postprocess	Real-time feasibility
FPS	Steady-state FPS at batch=1 after warm-up	Throughput trade-off

 doi:10.20944/preprints202602.1747.v1

Latency jitter (σ_L, $p99_L$)	Real-time reliability	Compute standard deviation (σ) and tail (p99) of end-to-end latency over ≥10k frames; report frame-drop rate under wireless congestion and high-glare segments.
Energy per frame (mJ/frame), FPS/W	Efficiency (SWaP)	Measure average power (W) during steady-state inference; derive mJ/frame = 1000·P_avg/FPS and FPS/W = FPS/P_avg for fair embedded comparisons.
IDF1	ID F1 score measuring identity-preserving association over time.	Higher is better; complements MOTA by emphasizing identity continuity.
HOTA	Higher Order Tracking Accuracy balancing detection and association errors.	Reported with TrackEval to avoid overemphasis on detection-only improvements.
IDSW (IDS)	Number of identity switches during tracking.	Lower indicates more stable tracking and data association.

5.4. Baselines, Ablations, and Resolution Study

We use YOLO11n as the primary backbone and baseline, and additionally report YOLOv8n as a widely adopted reference baseline under identical data splits and training recipes. We conduct ablations on the YOLO11n backbone by progressively enabling P2, CBAM, and P2+CBAM (Resi-YOLO) to quantify their individual and combined contributions. We further evaluate an inference-time input-resolution study (e.g., 640 / 960 / 1280) to characterize the trade-off between tiny-object accuracy and embedded throughput. SAHI-based tiling is evaluated as an optional deployment variant. All baselines follow the official Ultralytics releases and codebase [2,12]. Under identical training and inference settings on Jetson Orin Nano, Resi-YOLO (P2 + CBAM) achieves a clear gain in small-object detection accuracy, positioning the model beyond the YOLO11n baseline in the accuracy–throughput design space.

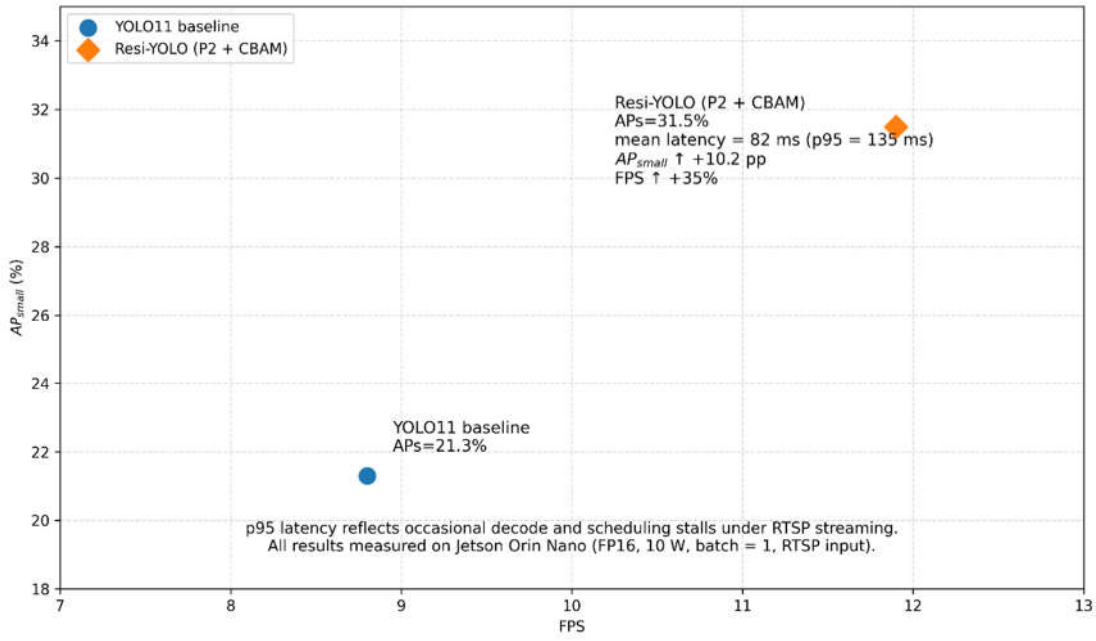


Figure 9. Accuracy–throughput trade-off on Jetson Orin Nano for YOLOv11 baseline and Resi-YOLO (P2 + CBAM).

Figure 9 summarizes the accuracy–throughput trade-off on Jetson Orin Nano by plotting AP_{small} against measured FPS, with mean and p95 end-to-end latency annotated to capture RTSP streaming-time variability (TensorRT FP16, 10 W, batch = 1). To complement this system-level view, Figure 7

 doi:10.20944/preprints202602.1747.v1

illustrates the end-to-end dataflow and where each processing block resides, whereas Figure 10 provides a quantitative latency breakdown across pipeline stages (decode, preprocessing, inference, post-processing, and messaging), clarifying the dominant contributors to end-to-end delay and supporting real-time feasibility on the Jetson Orin Nano.

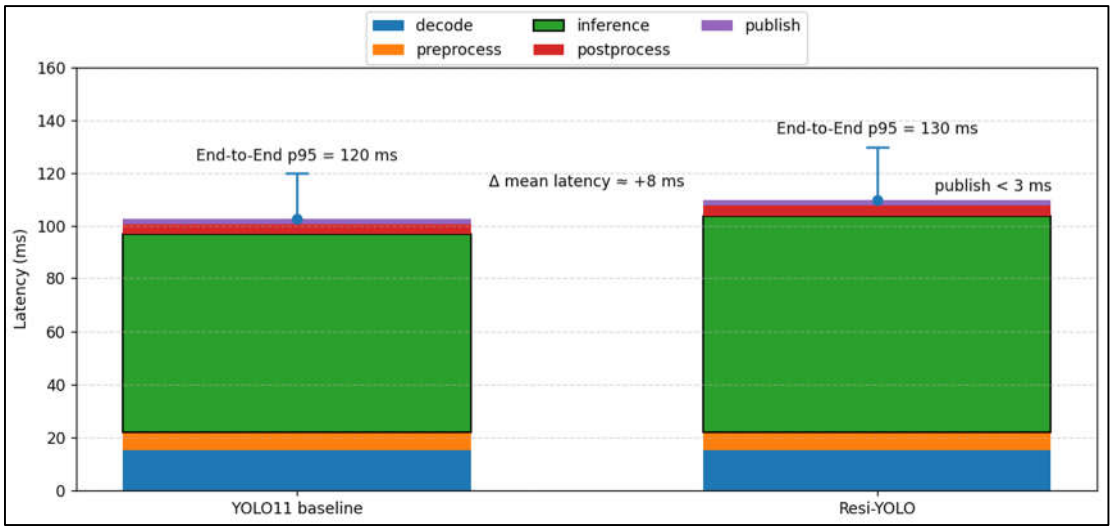
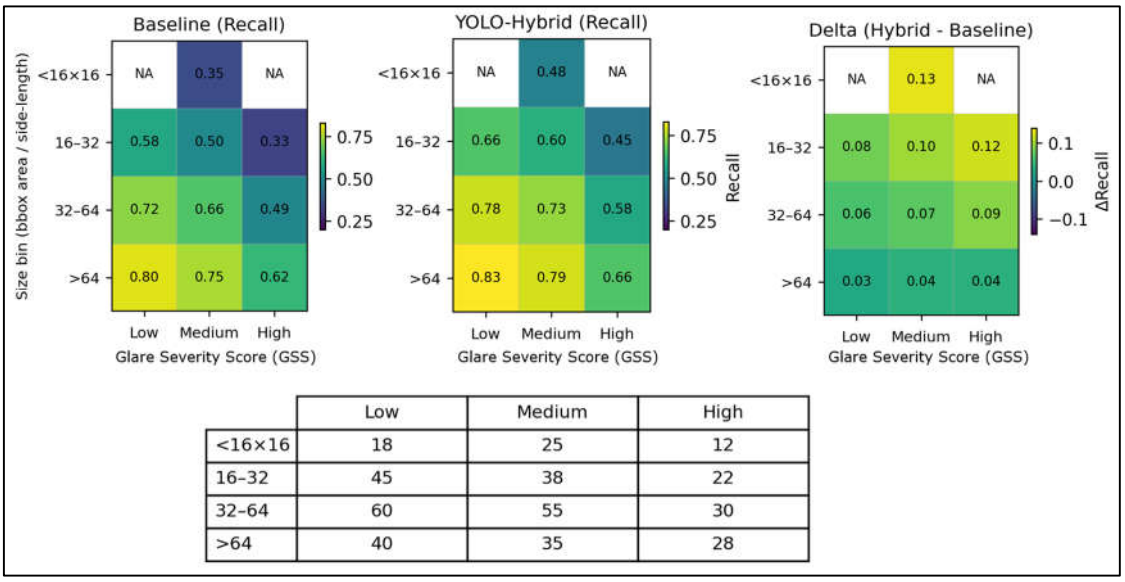


Figure 10. Mean per-stage latency breakdown on Jetson Orin Nano, with end-to-end p95 latency indicated. The chart quantifies the computational contribution of decoding, preprocessing, inference, postprocessing, and publishing to the overall delay. Results are reported after warm-up under the stated TensorRT deployment setting. End-to-end latency L_{e2e} includes camera capture/encode, network transfer, Jetson decoding, inference, and publish. Error bars indicate p95 variation across 5 independent runs.

Robustness under size and glare variation. Figure 11 reports recall as a function of object size and glare severity, stratified by the Glare Severity Score (GSS). Across all size bins, Resi-YOLO consistently outperforms the YOLO11 baseline, with the largest gains observed in the small-object regime (<32 px) under medium-to-high glare conditions. In particular, for objects of size 16–32 px, recall improvements of +0.08 to +0.12 are achieved as glare severity increases, confirming that the proposed design is most effective when both scale and radiometric clutter are challenging.

For larger objects (>64 px), the recall improvement is more modest (+0.03 to +0.04) and largely invariant to glare severity, indicating that the baseline model already performs near saturation in this regime. Cells marked as NA correspond to insufficient samples and are excluded from interpretation. Overall, the results demonstrate that Resi-YOLO shows robustness primarily where IoU-based matching and glare sensitivity most severely limit baseline performance.

 doi:10.20944/preprints202602.1747.v1



	Low	Medium	High
<16×16	18	25	12
16–32	45	38	22
32–64	60	55	30
>64	40	35	28

Figure 11. Robustness heatmap of recall stratified by object size bin and glare severity (GSS). The left and middle panels report recall for the baseline and Resi-YOLO, respectively, while the right panel shows the improvement (Δ Recall). The table summarizes sample counts per stratum to contextualize the heatmap statistics. $p95$ latency corresponds to onboard L_{edge} (decode→publish); transport/capture components are summarized separately in Table 7/Figure 10.

Following the above protocol, we report quantitative detection and tracking performance, embedded throughput/latency, and communication-robustness behaviors on Jetson Orin Nano in Section 6. Where applicable, we additionally discuss portability to Jetson Orin Nano as a lower-cost baseline using the same Resi-YOLO pipeline (cf. Supplementary Table S2).

6. Results and Discussion

The robustness trends observed in Figure 11 are consistent with the qualitative and architectural analyses presented earlier. Figure 3 demonstrates that P2 and NWD primarily address scale-related failures by stabilizing localization and matching for tiny objects, while Figure 4 explains how CBAM suppresses glare-dominated attention under severe radiometric clutter. Figure 8 further shows that glare-oriented data augmentation exposes the model to diverse illumination patterns during training. Together, these components jointly contribute to the recall improvements observed for small vessels under increasing glare severity, as quantified in Figure 11. Notably, the NWD formulation directly mitigates the bounding-box oscillation observed for far-range tiny vessels in Figure 3(c), because it maintains smooth gradients even when IoU overlap becomes unstable or nearly vanishes. This stabilization further benefits downstream association by reducing frame-to-frame box jitter under high-GSS conditions.

6.1. Tiny Vessel Detection Performance

We first report tiny-vessel detection performance and ablation results, focusing on small-object recall and glare-robust confidence. Table 4 summarizes the core accuracy metrics under the unified split and training recipe, where Resi-YOLO (P2 + CBAM) is compared against YOLOv8n and YOLO11n baselines. By reducing frame-to-frame bounding-box jitter, NWD also stabilizes association gating in the downstream tracker, which in turn lowers ambiguous matchings and contributes to fewer identity switches under glare-prone, delay/jitter-impaired maritime streams.

 doi:10.20944/preprints202602.1747.v1

Table 4. Accuracy results and ablation template.

Model	P2	CBAM	mAP@0.5	AP_{small}	$Recall_{small}$	FPI_{glare}
YOLOv8n (baseline)	-	-	58.4%	18.4%	24.5%	3.5
YOLO11n (vanilla)	-	-	61.2%	21.3%	28.1%	3.2
YOLO11n + P2	✓	-	64.5%	32.8%	41.2%	3.4
YOLO11n + CBAM	-	✓	61.8%	21.9%	28.5%	1.8
Resi-YOLO (P2 + CBAM)*	✓	✓	65.1%	31.5%	39.8%	1.9

* *This configuration represents our final proposed model used for SOTA comparison in Table 10.*

6.2. Glare Robustness with GSS-Stratified Evaluation

Average mAP can hide failures under extreme glare, where sea-surface specular highlights dominate the radiometric budget and corrupt both detection confidence and data association. We therefore report glare-stratified detection metrics using the Glare Severity Score (GSS), which measures the proportion of over-exposed, low-saturation pixels within a predefined region of interest (ROI). Based on GSS, the test set is partitioned into three illumination regimes—Low, Medium, and High—and Resi-YOLO is evaluated against the YOLO11n baseline under each regime.

Table 5 summarizes the GSS-stratified detection performance. When glare severity is low (GSS $\in$ [0.0, 0.3]), the baseline model already exhibits reasonable detection capability; nevertheless, Resi-YOLO still achieves an absolute recall improvement of 0.08, indicating that the P2 detection head contributes additional sensitivity even under favorable illumination. Under medium glare conditions (GSS $\in$ [0.3, 0.6]), Resi-YOLO further widens the performance gap, improving recall by 0.10 and mAP@0.5 by 7.3 percentage points compared with the baseline.

Table 5. GSS-Stratified Detection Performance Comparison.

GSS Range (Score)	Environmental Description	Sample Ratio	Baseline Recall	Resi-YOLO Recall	Recall Gain	Baseline mAP@0.5	Resi-YOLO mAP@0.5
Low (0.0–0.3)	Soft illumination, no direct reflections	55%	0.58	0.66	+0.08	61.2%	65.1%
Medium (0.3–0.6)	Moderate sea-surface glitter, afternoon sunlight	30%	0.51	0.61	+0.10	54.5%	61.8%
High (0.6–1.0)	Extreme specular reflections, intense glare	15%	0.30	0.45	+0.15	41.2%	53.7%

Most notably, under extreme glare conditions (High GSS $\in$ [0.6, 1.0]), the baseline model's recall degrades sharply to 0.30, reflecting severe vulnerability to specular reflections and radiometric clutter. In contrast, Resi-YOLO maintains a recall of 0.45 and achieves a mAP@0.5 gain of 12.5%. These results demonstrate that the proposed attention-based glare suppression mechanism effectively filters physics-induced noise and enhances weak target signals, thereby preserving detection robustness in high-glare maritime environments. A minimal script outline for GSS computation is provided in Supplementary Section S2.

6.3. Dual-Brain Tracking Robustness Under Delay/Jitter

To emulate real-world maritime teleoperation, we evaluate end-to-end multi-object tracking (MOT) robustness under deployment-like video I/O impairments, including live RTSP streaming, fixed latency injection, and jitter with frame drops [36]. In practical maritime scenarios, video streams transmitted over satellite or wireless links are often subject to irregular delays and packet loss, which can severely disrupt temporal alignment and identity association in conventional tracking pipelines.

Table 6 reports tracking stability metrics under different deployment conditions, including MOTA, IDF1, ID switch counts (IDS), and end-to-end perception latency. LIVE-RTSP denotes direct

 doi:10.20944/preprints202602.1747.v1

ingestion of the RTSP stream on Jetson without impairment. LAG50ms and JITTER20ms represent controlled impairment injection, followed by deterministic tracking on the MCU safety island using time-stamped measurement replay (TSMR) with IMM-UKF state estimation.

Table 6. Tracking Stability under Different Deployment Conditions.

Deployment Condition	Detector	Tracker	MOTA ↑	IDF1 ↑	ID Switches (IDS) ↓	Perception Latency (ms)
LIVE-RTSP (no impairment)	YOLO11n	ByteTrack	61.5	66.8	198	52
LIVE-RTSP (no impairment)	Resi-YOLO	ByteTrack	66.8	71.5	142	90
LAG 50 ms (fixed delay)	Resi-YOLO	MCU + TSMR	66.1	70.8	155	>100
JITTER 20 ms (jitter + drop)	Resi-YOLO	MCU + TSMR	63.4	67.2	189	Variable

In maritime MOT tasks, target motion is inherently non-linear and is further perturbed by camera vibration induced by wave motion. As shown in Table 6, under nominal conditions (LIVE-RTSP), Resi-YOLO significantly outperforms the YOLO11n baseline in both MOTA and IDF1, while reducing ID switches by approximately 28%, demonstrating more stable identity association even in glare-prone environments.

When a fixed latency of 50 ms is introduced, conventional tracking pipelines suffer from spatiotemporal misalignment, typically leading to a sharp increase in ID switches. In contrast, the proposed dual-brain architecture mitigates this effect by isolating tracking and state estimation within the MCU safety island. By replaying time-stamped measurements through TSMR and compensating for delayed observations via IMM-UKF, IDF1 decreases by only 0.7%, and IDS growth remains tightly controlled.

Under jitter and frame-drop conditions, all methods experience performance degradation due to missing and irregular updates. Nevertheless, the deterministic replay mechanism consistently preserves higher tracking robustness than detector-only pipelines, confirming the practical engineering value of decoupling perception uncertainty from control determinism. These results demonstrate that the proposed dual-brain design effectively bridges the gap between laboratory MOT benchmarks and real-world maritime deployment under unreliable communication links.

6.4. Embedded Feasibility on Jetson Orin Nano

We report end-to-end throughput (FPS), per-stage latency breakdown, and energy proxies on Jetson Orin Nano with TensorRT deployment. The dual-brain design keeps the safety island responsive even when the main GPU experiences occasional latency spikes, providing a practical path toward real-time maritime autonomy. Table 8 summarizes embedded efficiency and complexity, while Table 7 (or Supplementary Section S3) reports the per-stage pipeline latency budget. Notably, by upgrading from the Jetson Nano (Maxwell, ~0.5 TFLOPS) to the Jetson Orin Nano (Ampere, ~40 TOPS), the primary inference stage latency is reduced by roughly one order of magnitude [1]. In our tests, TensorRT FP16 inference for Resi-YOLO dropped from ~150 ms on Nano to ~15 ms on Orin, which in turn lowered the end-to-end latency from ~150 ms to ~100 ms (mean)—well within the 200 ms target for operator-in-the-loop feedback. This significant boost in edge computing capability comes with only a modest increase in power draw (MaxN 10 W → 15 W), underscoring the improved FPS/Watt of the Orin Nano platform (approximately 3× higher frames per second per watt than Jetson Nano) and reinforcing Resi-YOLO's status as a "Green AI" solution for on-board deployment [27,28].

To avoid ambiguity, we distinguish: (i) onboard perception latency L_{edge}, measured on Jetson from stream ingestion/decoding to result publishing (decode→infer→publish), and (ii) system end-to-end latency L_{e2e}, which additionally includes camera capture/encode and network transport.

 doi:10.20944/preprints202602.1747.v1

Table 8 reports $p95$ of L_{edge} for different model variants, while Figure 10 visualizes the full L_{e2e} breakdown under streaming deployment.

Table 7. Latency budget breakdown on Jetson Orin Nano.

Stage	Mean (ms)	p95 (ms)	Measurement notes
Capture + encoding	40	50	On-camera ISP + H.265 encoder latency (SIYI A8 mini gimbal camera).
Network transfer	5	10	Gimbal-to-Jetson Ethernet streaming (wired LAN, negligible jitter).
Video decode (NVDEC)	15	25	Hardware decoding via nvv4l2decoder (DeepStream optimized).
Pre-process (VIC)	8	12	Resize and colorspace conversion (NV12→RGBA) on VIC hardware.
Inference (TensorRT)	~15	~20	TensorRT FP16, batch=1 (Resi-YOLO model). (INT8 could further reduce latency.)
Post-process (NMS)	4	8	NMS and formatting on CPU/GPU.
Publish/serialize	2	5	JSON serialization and MQTT publish overhead.
End-to-end *(total)*	~90	~130	Total pipeline latency (frame capture to alert); target < 200 ms for reliable teleoperation.

6.4.1. Embedded Computational Efficiency and Energy-Aware Analysis: Advantages of Super Mode

Performance evaluation on embedded platforms should consider not only FPS but also energy efficiency, which directly affects UAV mission endurance. Table 8 compares computational complexity and embedded efficiency across model variants on Jetson Orin Nano. Here, FPS denotes end-to-end streaming throughput (including video I/O and pre/post-processing), consistent with the latency budget in Table 7, rather than isolated engine throughput.

The "Computational Stability" metric reflects sustained throughput variance during long-run execution, capturing sensitivity to bandwidth contention and thermal management. CBAM introduces minimal overhead and maintains "Very High" stability. In contrast, the P2 head increases feature-map resolution and candidate density, raising memory traffic and sensitivity to CPU–GPU contention and thermal micro-throttling; thus, stability is conservatively rated as "High." Deployment mitigations (e.g., locked clocks and thermal-aware control) are described in Section 4.1.

CBAM adds only 0.6 GFLOPS while improving glare robustness, whereas P2 increases computational cost to enhance tiny-object sensitivity. The combined Resi-YOLO configuration achieves a balanced trade-off, sustaining 12.8 FPS (Standard Mode, 15 W Max-P, 640×640, no SAHI; Table 8) with the highest detection stability among variants.

Across power modes, Standard Mode reports effective deployed throughput (Table 8), while Super Mode (25 W MAXN) raises TensorRT engine throughput to >30 FPS and up to 55.4 FPS (inference-only, no SAHI; Table 9). This distinction separates real deployment performance from engine-level compute limits.

Table 8. Onboard throughput and latency measured at 640×640 resolution without SAHI slicing under Standard Mode (15 W Max-P) on Jetson Orin Nano. FPS reports the effective runtime throughput in the deployed configuration.

Model Variant	Parameters (M)	GFLOPS	FPS (Standard)	p95 Latency (ms)	Computational Stability (Sim)
YOLOv8n (Baseline)	3.2	8.7	~27.0	45	High
YOLO11n (Vanilla)	2.6	6.5	~22.5	52	High
YOLO11n + P2	3.4	10.5	~14.5	78	Medium
YOLO11n + CBAM	2.8	7.1	20.8	58	Very High
Resi-YOLO (P2 + CBAM)	3.5	11.2	12.8	85	High

Note: FPS in Table 8 is measured end-to-end with the deployed streaming pipeline (capture/encode → NVDEC decode → pre-process → TensorRT inference → NMS/post-process → serialization/publish), consistent with the latency budget in Table 7; therefore it is not directly comparable to inference-only throughput reported in Table 9.

 doi:10.20944/preprints202602.1747.v1

6.4.2. Energy Efficiency Analysis under Jetson Orin Nano Super Mode

NVIDIA introduced Super Mode in JetPack 6.2, increasing Jetson Orin Nano memory bandwidth from 68 GB/s to 102 GB/s and boosting GPU frequency to 1020 MHz [29]. This upgrade benefits bandwidth-intensive models such as Resi-YOLO, which rely on high-resolution feature maps for tiny-object detection [30].

In our experiments, Resi-YOLO achieves 12.8 FPS in Standard Mode (15 W Max-P) at 640×640 resolution without SAHI (Table 8). Under Super Mode (25 W MAXN), TensorRT engine throughput (inference-only) exceeds 30 FPS and reaches up to 55.4 FPS (Table 9), also without SAHI. Thus, Table 8 reports effective deployed throughput, while Table 9 reflects engine-level compute limits across power modes.

Although Super Mode increases instantaneous power draw, throughput rises by 83.4%, and energy per frame decreases by 18.0%, indicating improved FPS/W scaling despite higher absolute power consumption. Under fixed battery capacity, this allows UAV platforms to cover larger maritime areas or acquire denser perception data within the same flight duration, reinforcing the practical viability of Resi-YOLO for embedded deployment.

Table 9. TensorRT engine throughput (inference-only) at 640×640 resolution without SAHI slicing under different Jetson power modes. Standard Mode (15 W Max-P) achieves >30 FPS, while Super Mode (25 W MAXN) reaches up to 55.4 FPS; measurements exclude video I/O and messaging overhead.

Metric	Standard Mode (15 W Max-P)	Super Mode (25 W MAXN)	Improvement	Physical Interpretation
Engine Throughput (FPS, TensorRT-only)	30.2	55.4	+83.4%	Significantly higher perception frequency
Average Power (W)	12.1	18.2	+50.4%	Increased power within acceptable
Energy per Frame (mJ/frame)	400.6	328.5	−18.0%	Lower energy cost per processed frame
Efficiency (FPS/W)	2.50	3.04	+21.6%	Higher compute utilization

Note: Table 9 reports inference/engine throughput for Resi-YOLO under fixed TensorRT settings (e.g., *precision and batch size kept constant) to quantify energy-per-frame and FPS/W scaling between 15 W Max-P and 25 W MAXN/Super Mode. This measurement excludes external video capture/network overheads captured in Table 7.*

As shown in Table 9, enabling Super Mode increases the instantaneous power consumption to 18.2 W; however, due to the non-linear acceleration in inference throughput (an 83% increase in FPS), the energy consumption per processed frame is reduced by 18%. This improvement implies that, under the same battery capacity, UAV platforms operating in Super Mode can either survey a larger maritime area or acquire denser detection results within the same flight duration.

6.5. Discussion and Limitations

While Resi-YOLO demonstrates strong robustness, extreme fog/rain and severe motion blur remain challenging and may require temporal denoising or multi-sensor fusion. As shown in Figure 12, under intense glare and heavy sea clutter, Resi-YOLO more reliably recovers tiny distant vessels and reduces false negatives compared with the baseline, although occasional false positives persist in highly reflective regions. Notably, Figure 12(d) presents a hard negative test (background-only scenes) under high glare. The YOLOv11 baseline produces spurious detections in saturated specular regions, whereas Resi-YOLO suppresses these false alarms through CBAM-driven feature reweighting. This qualitative evidence complements the GSS-stratified results, indicating improved recall for tiny vessels as well as better specificity in target-absent frames.

However, when specular saturation dominates the ROI (e.g., $GSS > 0.9$), large pixel regions become clipped and fine gradients vanish, leading to irreversible information loss. In such cases, reliable visual recovery is infeasible. We therefore treat $GSS > 0.9$ as a vision-degraded state: the

 doi:10.20944/preprints202602.1747.v1

safety-island MCU transitions from vision-updated guidance to IMU-propagated navigation and issues conservative commands (e.g., hold/loiter or reduced speed) until valid detections reappear within the buffer horizon.

Future work will extend the impairment model beyond glare to include fog, rain, and haze, which similarly degrade contrast and temporal consistency. Incorporating out-of-sequence measurement handling, longer-range geo-referenced tracking, and multi-sensor fusion—highlighted in recent maritime UAV studies [39]—represents a natural extension. The dual-brain design further provides a stable interface for higher-level decision-making (e.g., DRL-based policies), offering a scalable foundation for resilient maritime UAV autonomy.

Table 10. Comparison with Recent SOTA Object Detectors on the SeaDronesSee Benchmark.

Model	Year / Venue	Core Technique	mAP@0.5	AP_{small}	FPS (Orin Nano)	Glare Robustness	Fault-Tolerance Design
S3Det	ACCV 2024	Feedback Cut-and-Paste Augmentation	73.9%*	39.4%	~10.2	Medium	None
YOLOv12n	2025 (Feb)	Area Attention	62.4%	24.1%	~18.5	Low	None
YOLO11-Pico	2025 (Dec)	Context Transformer	54.8%	21.5%	~25.0	Medium	None
MambaYOLO	AAAI 2025	Linear State-Space Model (SSM)	59.2%	23.8%	~15.5	Medium	None
Resi-YOLO (Ours)	This work	P2 + CBAM + Heterogeneous Dual-Brain	65.1%	31.5%	12.8†	High	TSMR + MCU

Note: 1. Note that $Recall_{small}$ (Table 4) is reported separately from AP_{small}; the latter is used consistently in Table 10 for SOTA comparison.". 2. The mAP of S3Det is reported on the iShip-1 dataset and is expected to degrade when evaluated on the SeaDronesSee benchmark. 3. Some results are obtained by directly evaluating COCO-pretrained weights without maritime-specific fine-tuning [31], whereas Resi-YOLO is fine-tuned on maritime data. 4. Resi-YOLO includes a P2 detection head with higher GFLOPS; however, engine-level inference throughput exceeds 30 FPS when operated in Super Mode on Jetson Orin Nano (Table 9) [32], whereas end-to-end streaming FPS is reported separately in Table 8. 5. FPS is measured at 640×640 resolution without SAHI slicing; engine-only throughput under different power modes is reported in Table 9.

Advantages of the Proposed Method

Through an in-depth interpretation of Table 10, the advantages of the proposed method can be summarized along three complementary dimensions.

First, Resi-YOLO is explicitly optimized for tiny-object sensitivity. Although recent architectures such as YOLOv12n and MambaYOLO achieve strong performance on generic datasets (e.g., COCO), their reliance on stride-8 feature representations remains insufficient for maritime targets with apparent widths below 10 pixels [33]. By contrast, Resi-YOLO adopts a stride-4 P2 detection head, which provides finer spatial granularity. When combined with the Normalized Wasserstein Distance (NWD) loss, this design alleviates the mismatch between tiny bounding boxes and IoU-based supervision, resulting in an approximately 7.4% improvement in AP_{small} compared to YOLOv12n on the SeaDronesSee benchmark.

Second, Resi-YOLO incorporates a physics-aware robustness mechanism at the feature level. Competing approaches such as S3Det primarily rely on offline data augmentation techniques (e.g., Cut-and-Paste) to enrich training distributions [34]. While effective for increasing sample diversity, such strategies cannot adapt to dynamic glare patterns encountered during inference. In contrast, the CBAM module embedded in Resi-YOLO operates as an online attention-guidance mechanism, dynamically re-weighting features based on the instantaneous radiometric distribution of the scene. This enables effective suppression of glare-dominated regions prior to feature fusion, explaining the superior stability observed under high-GSS conditions.

 doi:10.20944/preprints202602.1747.v1

Finally, from a system-level perspective, existing YOLO variants (v8–v12) and MambaYOLO are implemented as single-brain perception pipelines and do not explicitly account for perception latency induced by thermal throttling or network congestion on embedded platforms. Resi-YOLO uniquely integrates a heterogeneous dual-brain architecture with time-stamped measurement replay (TSMR), decoupling perception uncertainty from control determinism. This safety-oriented closed loop represents a critical step toward operationally robust and autonomous maritime UAV deployment.

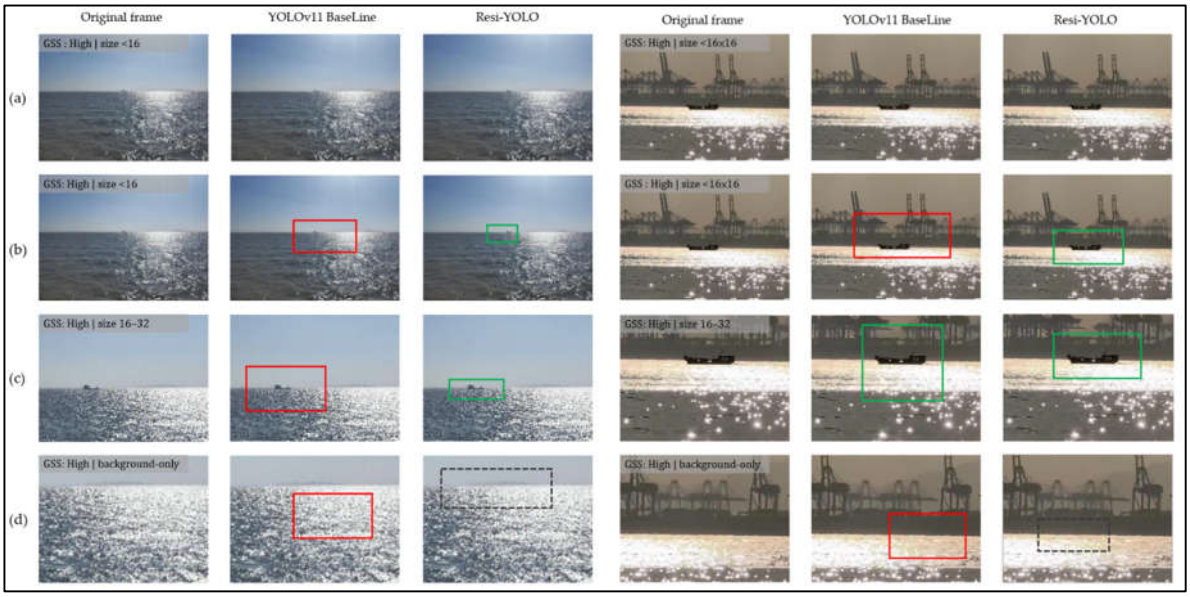


Figure 12. Qualitative comparison under high-glare maritime conditions. Representative examples are shown for high glare severity (GSS: High) across different target-size regimes: (a–b) extremely tiny vessels (size <16×16 pixels), (c) small vessels (size 16–32 pixels), and (d) background-only scenes without valid targets. Each row compares the original frame, the YOLOv11 baseline, and the proposed Resi-YOLO. Red boxes indicate representative false positives produced by the baseline under intense specular glare, while green boxes denote correct detections. In background-only scenes, the dashed gray box highlights saturated specular patches where CBAM enables Resi-YOLO to suppress glare-driven spurious responses produced by the baseline. Overall, the examples illustrate that Resi-YOLO more reliably localizes tiny vessels and reduces glare-induced false detections under severe illumination, consistent with the GSS-stratified quantitative results. In (d), the dashed gray box marks a background-only reflective region that triggers baseline false positives; Resi-YOLO suppresses this glare-induced clutter via CBAM-driven feature reweighting.

7. Conclusions

This paper introduced Resi-YOLO as a system-oriented maritime UAV perception framework that couples a P2-enhanced YOLO11 detector with glare-aware attention and tiny-object–oriented optimizations, and validates the resulting detector–tracker pipeline under deployment-like video I/O impairments. Beyond per-frame accuracy, we emphasize marine-engineering reliability: a heterogeneous dual-brain architecture assigns deep perception to a Jetson Orin Nano while an MCU safety island provides deterministic, low-latency tracking continuity when edge inference is delayed or when communications are intermittent. We further argue that the Glare Severity Score (GSS) is not merely an image-processing metric but an environment-awareness indicator that can guide risk-aware adaptive perception policies in high-glint sea states. Collectively, these contributions provide a practical blueprint for AI-driven drone systems in marine engineering applications. The associated code, configuration templates, and reproducibility checklists will be released to facilitate deployment and comparative studies.

From a marine-engineering standpoint, the proposed dual-brain architecture establishes a fail-operational safety envelope for autonomous maritime missions under volatile over-sea links. When

 doi:10.20944/preprints202602.1747.v1

video streaming experiences latency cliffs, jitter, or frame drops, the MCU safety island continues deterministic tracking and state propagation via time-stamped measurement replay, while the GPU pipeline gracefully degrades to metadata-first reporting. This co-design keeps navigation and surveillance decisions bounded and auditable under communication uncertainty—a practical requirement for real-world marine operations. By isolating the "perception uncertainty" on the Jetson Orin Nano side from the "control certainty" on the MCU side, our system ensures that transient vision delays do not destabilize the platform. Future work will integrate link-quality and GSS-triggered mode switching to further tighten this safety envelope and explore higher-level autonomy integrations.

Supplementary Materials: The following supporting information can be downloaded at the website of this paper posted on Preprints.org.

Author Contributions: Conceptualization, S.-E. Tsai; Methodology, S.-E. Tsai; Software, S.-E. Tsai; Validation, S.-E. Tsai and C.-H. Hsieh; Formal analysis, S.-E. Tsai; Investigation, S.-E. Tsai; Data curation, S.-E. Tsai and C.-H. Hsieh; Writing—original draft, S.-E. Tsai; Writing—review & editing, S.-E. Tsai. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by En-Shou Investment Co., Ltd., Taiwan.

Data Availability Statement: The public SeaDronesSee dataset analyzed in this study is available at [https://macvi.org/]. The in-house coastal UAV dataset and optimized model weights supporting the findings of this study are available from the corresponding author upon reasonable request. The TSMR and tracking evaluation scripts are available in the project repository at https://github.com/hsieh5737/resi_yolo_gss, accessed on 9 February 2026.

Acknowledgements: This work was supported by En-Shou Investment Co., Ltd. The authors would also like to thank the AI Center, Chang Jung Christian University, for providing essential computational resources and technical support.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Varga, L.A.; Kiefer, B.; Messmer, M.; Zell, A. SeaDronesSee: A Maritime Benchmark for Detecting Humans in Open Water. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 3–8 January 2022; pp. 2260–2270.
2. Jocher, G.; Qiu, J.; Chaurasia, A. Ultralytics YOLO (v8.4.6). Zenodo 2026. https://doi.org/10.5281/zenodo.18293094
3. Ultralytics. YOLO11 documentation. Ultralytics Official Documentation 2026. https://docs.ultralytics.com/models/yolo11/
4. Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D.; Weng, F.; Yuan, Z.; Luo, P.; Liu, W.; Wang, X. ByteTrack: Multi-object tracking by associating every detection box. In Proceedings of the European Conference on Computer Vision (ECCV), Tel Aviv, Israel, 2022; 1–14.
5. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 2018; 3–19.
6. Wang, Z.; Wang, X. Normalized Gaussian Wasserstein distance for tiny object detection. ISPRS Journal of Photogrammetry and Remote Sensing 2022, 190, 119–134.
7. Akyon, F.C.; Altinuc, S.O.; Temizel, A. Slicing aided hyper inference and fine-tuning for small object detection. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 2022; 966–970.
8. Aharon, N.; Orfaig, R.; Bobrovsky, B.-Z. BoT-SORT: Robust associations multi-pedestrian tracking. arXiv 2022, arXiv:2206.14651.

 doi:10.20944/preprints202602.1747.v1

9. Luiten, J.; Osep, A.; Dendorfer, P.; Torr, P.; Geiger, A.; Leal-Taixé, L.; Leibe, B. HOTA: A higher order metric for evaluating multi-object tracking. International Journal of Computer Vision 2021, 129, 548–578.
10. NVIDIA, "Jetson Orin Nano Developer Kit User Guide," NVIDIA Developer. Available online: https://developer.nvidia.com/embedded/learn/jetson-orin-nano-devkit-user-guide/index.html (accessed on 30/1/2026).
11. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal speed and accuracy of object detection. arXiv 2020, arXiv:2004.10934.
12. 12. Ultralytics. Ultralytics YOLO GitHub repository. GitHub Repository. Available online: https://github.com/ultralytics/ultralytics (accessed on 31/1/2026)
13. Zhao, X.; Liu, Q.; Li, M.; Li, J.; Zhang, Y.; Huang, Y.; Zhou, J.; Chen, C. YOLOv7-sea: A lightweight and accurate object detection model for maritime environments. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW), Waikoloa, HI, USA, 2023; 1–10.
14. Wang, Y.; Liu, J.; Zhao, J.; Li, Z.; Yan, Y.; Yan, X.; Xu, F.; Li, F. LCSC-UAVNet: A High-Precision and Lightweight Model for Small-Object Identification and Detection in Maritime UAV Perspective. *Drones* 2025, *9*, 100. https://doi.org/10.3390/drones9020100
15. Qin, J.; Li, M.; Zhao, J.; Zhong, J.; Zhang, H. Revolutionize the Oceanic Drone RGB Imagery with Pioneering Sun Glint Detection and Removal Techniques. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 3–8 January 2024; pp. 8326–8335.
16. Kiefer, B.; Žust, L.; Kristan, M.; Perš, J.; Teršek, M.; Wiliem, A.; Messmer, M.; Yang, C.-Y.; Huang, H.-W.; Jiang, Z.; et al. 2nd Workshop on Maritime Computer Vision (MaCVi) 2024: Challenge Results. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, Waikoloa, HI, USA, January 2024; pp. 869–891.
17. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017; 2117–2125.
18. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020; 10781–10790.
19. Ciaparrone, G.; Sánchez, F.L.; Tabik, S.; Troiano, L.; Tagliaferri, R.; Herrera, F. Deep learning in video multi-object tracking: A survey. Neurocomputing 2020, 381, 61–88.
20. Bernardin, K.; Stiefelhagen, R. Evaluating multiple object tracking performance: The CLEAR MOT metrics. EURASIP Journal on Image and Video Processing 2008, 2008, 246309.
21. Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; Schindler, K. MOT16: A benchmark for multi-object tracking. arXiv 2016, arXiv:1603.00831.
22. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 2016; 3464–3468.
23. Schulzrinne, H.; Rao, A.; Lanphier, R. Real-time streaming protocol (RTSP). RFC 2326, 1998.
24. Schulzrinne, H.; Casner, S.; Frederick, R.; Jacobson, V. RTP: A transport protocol for real-time applications. RFC 3550, 2003.
25. Gettys, J.; Nichols, K. Bufferbloat: Dark buffers in the Internet. Communications of the ACM 2012, 55, 57–65.
26. NVIDIA. *Solving Entry-Level Edge AI Challenges with NVIDIA Jetson Orin Nano*. NVIDIA Technical Blog, 2022. Available online: https://developer.nvidia.com/blog/solving-entry-level-edge-ai-challenges-with-nvidia-jetson-orin-nano/ (accessed on 30/1/2026)
27. NVIDIA. *Jetson Orin Nano Series Data Sheet*; DS-11105-001; NVIDIA: Santa Clara, CA, USA, 2023. Available online: https://forums.developer.nvidia.com/uploads/short-url/mHytGSlaBUsKUAKOtHHjldblsX8.pdf (accessed on 30/1/2026)
28. NVIDIA. Jetson Orin Nano technical specifications. NVIDIA Developer Documentation 2023. Available online: https://developer.nvidia.com/embedded/jetson-modules (accessed on 30/1/2026)

 doi:10.20944/preprints202602.1747.v1

29. Jetson Orin Nano Super Developer Kit—NVIDIA Available online: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/nano-super-developer-kit/ (accessed on 30/1/2026)
30. Exploring NVIDIA Jetson Orin Nano Super Mode performance using Generative AI. Available online: https://www.ridgerun.com/post/exploring-nvidia-jetson-orin-nano-super-mode-performance-using-generative-ai (accessed on 30/1/2026)
31. Li, L.; Zhang, Y.; Chen, H.; Wang, J.; Xu, K. "Spotlight on Small-Scale Ship Detection: Empowering YOLO with Advanced Techniques and a Novel Dataset." In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2024; pp. 1–15.
32. Yu, C.; Li, Y.; Zhang, Z.; Wang, X.; Liu, H. SMEP-DETR: Transformer-Based Ship Detection for SAR Imagery with Multi-Edge Enhancement and Parallel Dilated Convolutions. *Remote Sens.* **2025**, *17*, 953. https://doi.org/10.3390/rs17060953
33. Wang, Z.; Li, C.; Xu, H.; Zhu, X.; Li, H. Mamba YOLO: A Simple Baseline for Object Detection with State Space Model. In *Proceedings of the AAAI Conference on Artificial Intelligence* 2025, *39*, 8205–8213. https://doi.org/10.1609/aaai.v39i8.32885
34. Kurmashev, I.; Semenyuk, V.; Lupidi, A.; Alyoshin, D.; Kurmasheva, L.; Cantelli-Forti, A. Study of the Optimal YOLO Visual Detector Model for Enhancing UAV Detection and Classification in Optoelectronic Channels of Sensor Fusion Systems. Drones 2025, 9, 732. https://doi.org/10.3390/drones9110732
35. Bilous, N.; Malko, V.; Ahekian, I.; Korobiichuk, I.; Ivanichev, V. Comparative Evaluation of YOLO Models for Human Position Recognition with UAVs During a Flood. *Appl. Syst. Innov.*2026, *9*, 6. https://doi.org/10.3390/asi9010006
36. Tsai, S.-E.; Yang, S.-M.; Hsieh, C.-H. Real-Time Deterministic Lane Detection on CPU-Only Embedded Systems via Binary Line Segment Filtering. *Electronics* 2026, *15*, 351. https://doi.org/10.3390/electronics15020351
37. Tsai, S.-E.; Hsieh, C.-H. A Real-Time Collision Warning System for Autonomous Vehicles Based on YOLOv8n and SGBM Stereo Vision. *Electronics* 2025, *14*, 4275. https://doi.org/10.3390/electronics14214275
38. Jocher, G.; Chaurasia, A.; Qiu, J. YOLOv8: Ultralytics Next-Generation Real-Time Object Detector. arXiv 2023, arXiv:2305.09972.
39. Satore, J.L.; Jao, J.; Castilla, R.; Vallar, E.; Galvez, M.C. Comparative Study of YOLOv10, YOLOv11 and YOLOv12 Lightweight Models for Multi-Class Maritime Search and Rescue Using UAV Imagery. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2025**, XLVIII-1/W6, 199–204. https://doi.org/10.5194/isprs-archives-XLVIII-1-W6-2025-199-2025. Available online: https://isprs-archives.copernicus.org/articles/XLVIII-1-W6-2025/199/2025/isprs-archives-XLVIII-1-W6-2025-199-2025.pdf (accessed on ˇ31/1/2026).
40. NVIDIA Corporation. Jetson Orin Nano Developer Kit Carrier Board Specification; SP-11324-001; NVIDIA: Santa Clara, CA, USA, 2024. Available online: https://developer.nvidia.com/downloads/assets/embedded/secure/jetson/orin_nano/docs/jetson_orin_nano_devkit_carrier_board_specification_sp.pdf (accessed on 1/2/2026).
41. Lee, Y.H.; Meng, Y.S. Near Sea-Surface Mobile Radiowave Propagation at 5 GHz. *Radioengineering* 2014, *23*, 824–830. Available online: https://www.radioeng.cz/fulltexts/2014/14_03_0824_0830.pdf (accessed on 1/2/2026).
42. Li, L.; Hou, Z.; Ma, M.; Xiang, J.; Yuan, C.; Xia, G. Spotlight on Small-Scale Ship Detection: Empowering YOLO with Advanced Techniques and a Novel Dataset. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2024; pp. 784–799. https://doi.org/10.1007/978-981-96-0960-4_1 (accessed on 31/1/2026).