Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.3, June 2026, pp. 2710~2722
e-ISSN: 2548-6861

Improving YOLO12 Performance Using Efficient Channel Attention For Ship Object Detection

Richard Christoper Subianto [1], Muhammad Naufal [2]*, Farrikh Alzami [3]
[123] Informatics Engineering, Dian Nuswantoro University, Semarang, Indonesia
111202314922@mhs.dinus.ac.id [1], m.naufal@dsn.dinus.ac.id [2], alzami@dsn.dinus.ac.id [3]

Article Info

Article history:

Received 2026-05-22
Revised 2026-05-30
Accepted 2026-06-11

Keyword:

Attention,
Efficient Channel Attention,
Object Detection,
Ship,
YOLO12.

ABSTRACT

Ship object detection in aerial imagery remains a critical challenge due to complex marine backgrounds, varying object scales, and occlusion, which often lead to unstable model performance. This research proposes integrating the Efficient Channel Attention (ECA) module into the YOLO12-L architecture to enhance feature selectivity and prediction robustness. The model was trained for 500 epochs on the Ship Detection from Aerial Images dataset, comprising 621 images and 1,951 annotated ship instances, and performance was evaluated across five distinct random seeds to ensure statistical reliability. Quantitative results demonstrate that the proposed YOLO12-L + ECA model achieved a median Average Precision (mAP@50) of 71.32% and a Precision of 92.5%, outperforming the baseline YOLO12-L model. To evaluate statistical validity, a Paired Bootstrap Median Test with 100 resamples confirmed a statistically significant improvement in median performance (Δ = +1.01%, p = 0.02). Furthermore, the standard deviation of mAP@50 decreased from 1.1% in the baseline to 0.3% in the ECA model, representing a 72.7% reduction in performance variance. Computational efficiency analysis revealed that the ECA module introduced negligible overhead, adding merely 5 parameters (totaling 26,389,880) and keeping FLOPs constant at 89.4, while maintaining a high inference speed of 10.7 FPS (a marginal 2.5% reduction). These findings confirm that ECA effectively suppresses background noise, stabilizes detection outputs, and provides statistically significant improvements without compromising architectural efficiency. The proposed architecture offers a lightweight and reliable solution for automated maritime monitoring systems, particularly in challenging visual environments.

I. INTRODUCTION

The detection of ship objects in remote sensing and aerial imagery is crucial for maritime traffic monitoring, law enforcement in water areas, and coastal disaster mitigation [1], [2]. However, the implementation of automatic detection systems still faces significant obstacles [3]. The dynamic visual characteristics of the marine environment, including wave texture, light reflection, and fog obstruction, create complex backgrounds that burden computational models in distinguishing foreground objects from the background [4]. Additionally, ship objects in aerial imagery exhibit extreme scale variability, from small boats to large cargo ships, with spatial densities that often involve occlusion [5].

Single-stage object detection architectures such as You Only Look Once (YOLO) have been widely adopted due to their balance between inference speed and prediction accuracy [6], [7]. As the latest attention-centric state-of-the-art architecture, YOLO12-L introduces native structural improvements alongside advanced loss-function optimization to maximize gradient stability [8], [9]. Consequently, integrating an ultra-lightweight channel attention mechanism, such as ECA, into this cutting-edge framework represents a highly novel frontier, particularly for isolating complex maritime targets from dynamic environmental noise without disrupting the baseline's optimization path. However, without such a feature filtering mechanism, the model remains vulnerable to irrelevant visual information from backgrounds

[10], limiting its ability to isolate spatial features in small objects or high-density environments [11]. Furthermore, models trained on a single random initialization risk producing unstable performance variance that does not reflect true generalization ability [12], [13].

To address these architectural limitations and training instabilities, previous studies have explored various attention mechanisms and structural optimizations. Xue et al. introduced SSAM into YOLOv5s, improving object placement precision while maintaining a low computational burden [14], while Ahn et al. proposed SAFP-YOLO to reduce model complexity, though at a 5–6% accuracy trade-off [15]. In the ship detection domain specifically, Patel et al. achieved 65% mAP@50 using YOLOv5 [16], and Yildirim et al. achieved 62.30% with YOLOv4 Tiny [17], yet neither addressed feature channel selectivity nor validated performance consistency across multiple training initializations. Recent benchmarks across diverse computer vision domains [18], [19] further confirmed that YOLO architectures consistently outperform traditional methods across varying visual conditions, reinforcing the importance of rigorous multi-seed evaluation in architectural comparisons.

A review of the literature reveals four persistent research gaps. First, prevailing attention mechanisms such as CBAM and SSAM increase architectural complexity by adding convolutional or fully connected operations, thereby directly impacting inference speed [14], [20]. Second, pruning and quantization approaches often sacrifice detection accuracy for small or occluded objects [21]. Third, prior evaluations rarely include computational efficiency analysis or multi-seed consistency tests, leaving robustness to stochastic initialization unassessed [5], [22]. Fourth, generalization of attention mechanisms for ship detection remains limited by the availability of restricted datasets and inconsistent evaluation protocols [23].

To address these gaps, this study proposes integrating an Efficient Channel Attention (ECA) module into YOLO12-L, which adaptively selects feature channels via lightweight one-dimensional convolutions without significantly increasing the number of parameters or computational overhead [24], [25]. This study provides a rigorous evaluation of the ECA-integrated YOLO12-L framework, utilizing the Ship Detection from Aerial Images dataset to ensure diverse scale and background representations. The significance of the architectural improvements is rigorously validated using the Paired Bootstrap Median Test across five random initializations [12], [26].

This study provides distinct contributions across three domains. First, we strategically embed the lightweight ECA module into the YOLO12-L neck to enhance feature selectivity. Second, a rigorous multi-seed training protocol is combined with the Paired Bootstrap Median Test to isolate stochastic effects. Finally, our empirical results demonstrate a 72.7% reduction in performance variance, ensuring architectural stability.

II. METHOD

This research was conducted in several stages to ensure an orderly process and scientifically reliable results. The research flow is shown in Figure 1.

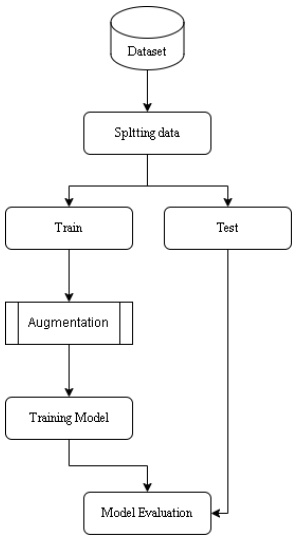


Figure 1. Research Flowchart

Figure 1 illustrates the systematic stages involved in developing an aerial imagery-based ship detection system.

A. Ship Dataset

The initial stage of this research utilizes a ship detection dataset sourced from the public Ship Detection from Aerial Images dataset available on the Kaggle platform [27]. This dataset consists of 621 aerial images at varying resolutions, manually annotated in a bounding-box format with a single main object class: Boat. Although the dataset comprises 621 aerial images, the total number of annotated ship instances is 1,951. This substantial number of annotated instances ensures adequate training data for the model to learn discriminative feature representations, particularly for the channel-wise attention mechanism, which operates on feature maps rather than raw images. The average density of ~3.1 ship objects per image further ensures that the model is exposed to diverse spatial configurations, scales, and contextual backgrounds during training. These images reflect a variety of visual conditions representative of ship detection scenarios in real environments, including significant variations in object scale, high object density, and complex sea backgrounds, including wave texture, sun glint, and clouds. In addition, the spatial distribution of ship objects in the images tends to be uneven, with most objects appearing in the center to the edges of the frame, which requires the model to have robust detection capabilities against positional variations. Consequently, this spatial characteristic justifies integrating the attention mechanism at the strategic feature level to retain global visual context. This dataset was chosen for its relevance to the focus of this research on ship detection from aerial optical imagery, as well as for its open availability for academic research.

Before training, the dataset was split into two sets at an 80:20 ratio, with 497 images for training and 124 for testing. This division ensures that the proportions of visual characteristics remain balanced across both subsets, enabling objective model evaluation on data not used during training.

B. Image Preprocessing

All input images were dynamically resized to a standardized resolution of 640 x 640 pixels and filtered to eliminate visual artifacts surrounding the vessel objects that could interfere with the spatial feature extraction process. Next, the image pixel values were normalized to the range 0-1 to accelerate gradient convergence and improve numerical stability during convolution operations. For data augmentation, this study used the Ultralytics framework's built-in preprocessing pipeline, which runs on-the-fly during training. The basic augmentations applied include horizontal flips with a probability of 0.5, brightness and contrast adjustments through Hue-Saturation-Value (HSV) color space transformations (*hsv_h* = 0.015, *hsv_s* = 0.7, *hsv_v* = 0.4), and geometric variations in the form of minor translation (translate = 0.1) and scaling (scale = 0.5).

C. YOLO12 Architecture

YOLO12-L (Large variant by Ultralytics) is chosen as the baseline model due to its deep feature extraction capabilities, which are essential for handling extreme scale and density variations in maritime aerial imagery. As illustrated in Figure 2, the architecture consists of three core components: backbone, neck, and head. The backbone constructs hierarchical spatial and channel representations by passing the input through initial convolutional layers, followed by C3K2 and Advanced C2f (A2C2f) blocks at varying resolution levels.

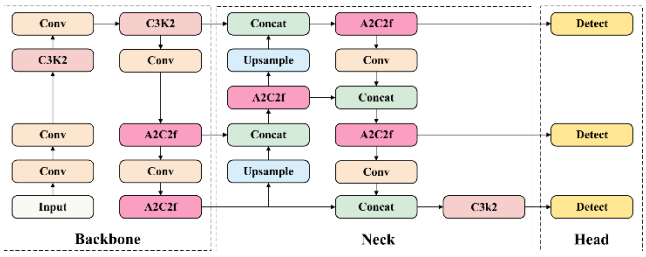


Figure 2. YOLO12 Architecture [9]

The neck component utilizes a Path Aggregation Network (PANet) framework to integrate multi-scale contextual information. This is achieved via upsampling and concatenation operations, structurally reinforced by interleaved A2C2f and C3K2 blocks to align and refine the fused features. For the prediction phase, a decoupled head with three independent detection branches is deployed. Each branch processes a distinct resolution level, enabling robust multi-scale detection that adaptively captures objects ranging from small boats to large cargo ships.

Furthermore, the architecture integrates Automatic Mixed Precision (AMP) to accelerate convergence and a cosine-based learning rate scheduler to stabilize optimization across complex marine environments.

D. Efficient Channel Attention (ECA)

The Efficient Channel Attention (ECA) module is an attention mechanism that adaptively improves feature selectivity across channel dimensions without significantly increasing computational complexity. Unlike conventional attention mechanisms such as Squeeze-and-Excitation (SE), which use fully connected layers to model inter-channel dependencies, ECA uses one-dimensional (1D) convolutional operations with adaptive kernel sizes to capture local interactions between feature channels. This approach enables efficient modeling of channel dependencies without reducing dimensionality or adding too many parameters, making it highly suitable for real-time object detection architectures like YOLO.

The working principle of ECA begins by performing Global Average Pooling (GAP) on the input feature map to generate a channel descriptor vector that represents the global information of each channel. Mathematically, for an input feature map $X \in RC\times H\times W$ with C channels, H height, and W width, spatial compression is performed using the following equation.

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_{c,i,j} \tag{1}$$

where z_c is the global average value for the c^{th} channel, and $x_{c,i,j}$ is the pixel value at position (i, j) in channel c. The result of this operation is a vector $z \in R^{C\times 1\times 1}$ containing global statistical information for each channel.

Next, the 1D convolution kernel size is adaptively determined based on the number of channels using the following equation.

$$k = \max\left(\left\lfloor \frac{\log_2(C)+1}{2} + 0.5 \right\rfloor, 3\right) \tag{2}$$

where k is the smallest odd kernel size corresponding to the number of channels C, this adaptive approach ensures that the scope of inter-channel interactions scales with the feature dimension, allowing the model to capture relevant local dependencies without excessive computation.

The vector $\mathbf{z}$ is then processed with a 1D convolution using a kernel of size k and a sigmoid activation function to generate the channel attention weights. Mathematically, this is shown in the following equation.

$$a_c = \sigma\left(\sum_{i=1}^{k} w_i \cdot z_{c+i-\lfloor k/2 \rfloor} + b\right) \tag{3}$$

where w_i is the 1D convolution weight, b is the bias, and $\sigma(\cdot)$ is the sigmoid activation function defined as the following equation.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (4)$$

The result of equation (4) is an attention weight vector $a \in [0, 1]^C$ which represents the importance level of each feature channel.

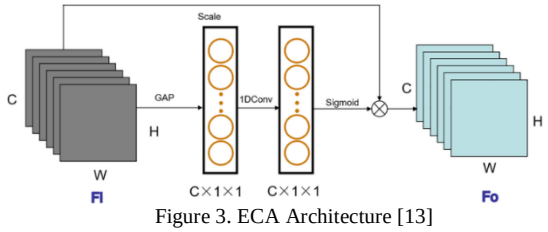


Figure 3. ECA Architecture [13]

Figure 3 shows the workflow of the ECA module, which starts with an input feature map $F_i \in R^{C \times H \times W}$. The feature is first processed through Global Average Pooling (GAP) to produce a C × 1 × 1 channel descriptor vector. This vector is then processed through a 1D convolution (1DConv) with an adaptive kernel size to capture local interactions between channels. The resulting 1D convolution is passed through a sigmoid activation function to produce channel attention weights ranging from 0 to 1. These attention weights are then multiplied element-wise with the original feature map via broadcasting, resulting in an output feature map $F_o \in R^{C \times H \times W}$ that is amplified on relevant channels and suppressed on channels containing noise or background information. This reweighting process can be formulated as follows.

$$y_{c,i,j} = x_{c,i,j} \cdot \sigma\left(\text{Conv1D}_k\left(\frac{1}{H \times W}\sum_{h=1}^{H}\sum_{w=1}^{W} x_{c,h,w}\right)\right) \quad (5)$$

where $x_{c,i,j}$ are the values at channel c and position (i, j) of the input feature map, Conv1D_k is a 1D convolution operation with kernel size k, and $\sigma(\cdot)$ is the sigmoid activation function. The final result $y_{c,i,j}$, is the output feature map whose weights have been adjusted based on channel attention.

E. Model Training

The next stage is Model training, carried out through two systematic experiments designed to measure the contribution of integrating the Efficient Channel Attention (ECA) module to improving ship detection performance. The entire computational process was run in the Kaggle Notebook environment with NVIDIA Tesla T4 × 2 GPU support, using the latest version of the Ultralytics YOLO framework based on PyTorch and CUDA 12.1. The hyperparameter configuration and training protocol for each stage of the experiment are summarized in Table I.

TABLE I
TRAINING HYPERPARAMETER CONFIGURATION

Parameter	Value
Epochs	500
Batch Size	8
Image Size	640 x 640
Optimizer	AdamW
Learning Rate	1 x 10^{-3}
Weight Decay	5 x 10^{-4}
Warmup Epochs	5
LR Scheduler	Cosine Decay
Random Seed	[42, 123, 456, 1234, 3407]

All experimental scenarios used the same YOLO12-L backbone and were conducted with the same hyperparameter configuration presented in Table I. The only variable across experiments was the presence of the Efficient Channel Attention (ECA) module.

Experiment 1 (Baseline) established the fundamental performance by training the YOLO12-L architecture from scratch (pretrained = False). This approach eliminated potential biases introduced by transfer learning from external datasets, ensuring that the model's convergence and generalization were evaluated solely on the basis of the visual characteristics of the maritime dataset. The training duration was extended to 500 epochs to facilitate comprehensive hierarchical feature extraction and optimal convergence. To ensure statistical robustness, each configuration was trained across five distinct random seeds. Advanced training techniques, including a cosine annealing learning rate scheduler, a 5-epoch warmup phase, and Automatic Mixed Precision (AMP), were employed to stabilize gradient updates and accelerate convergence without compromising numerical precision. Additionally, deterministic execution (deterministic = True) and multi-threaded data loading (workers = 4) were configured to guarantee reproducibility across all training sessions.

Experiment 2 (ECA) aimed to test the effectiveness of the Efficient Channel Attention (ECA) module when integrated into the YOLO12-L architecture. Training was still performed from scratch, with hyperparameter configurations fully aligned with those in Experiment 1, to ensure a head-to-head comparison between the baseline model and the attention-integrated variant. Varying the random seed across both experiments verified that the performance improvements obtained were consistent and independent of stochastic initialization effects. A patience parameter of 50 was used to prevent overfitting, automatically stopping training if validation metrics showed no significant improvement over 50 consecutive epochs. The ECA integration was performed modularly at layer 18 (P4 level), which handles medium-resolution feature maps optimal for capturing multi-scale maritime objects without altering the main gradient propagation flow, thereby clearly isolating the contribution of feature channel filtering from other training factors.

By structuring the experiment into two controlled scenarios, this study not only measures overall accuracy improvements but also isolates the specific contribution of the attention architecture. This approach ensures that any claimed performance improvements are methodologically sound, validated through rigorous multi-seed testing, and yield reproducible conclusions in the context of aerial imagery-based maritime object detection.

F. Model Testing on Test Data

The model testing phase was conducted using test data that had been separated from the beginning of the experiment, consisting of 124 images not involved in the training process. Evaluations were conducted five times, depending on the seed variations used in this study. This iterative evaluation approach aimed to test the model's robustness to stochastic initialization variability and ensure that the reported metrics reflected true generalization ability, not the luck of the initial weight initialization.

In the testing phase, model performance was measured using four standard object detection metrics: Precision, Recall, mAP@50, and mAP@50-95. Mathematically, these metrics are calculated using the following formulas.

Precision is a metric that measures the accuracy of a model's predictions, namely, the fraction of detections that are correct out of all detections generated [10].

$$\text{Precision} = \frac{TP}{TP + FP} \tag{6}$$

where TP (True Positive) is the number of correct detections, and FP (False Positive) is the number of incorrect detections.

Recall is a metric used to measure the extent to which a model can find all objects in the test data [10].

$$\text{Recall} = \frac{TP}{TP + FN} \tag{7}$$

where FN (False Negative) is the number of undetected objects.

mAP@50 (mean Average Precision at IoU threshold of 0.5) is a combined metric that takes into account both precision and recall at an Intersection over Union (IoU) threshold of 0.5 [10].

$$\text{mAP@50} = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{8}$$

where AP_i is the Average Precision of each class, and N is the number of classes.

For a more rigorous evaluation of object boundary accuracy, this study also reports mAP@50-95, the average Average Precision across 10 IoU thresholds (0.5 to 0.95 with 0.05 intervals) [10].

$$\text{mAP@50-95} = \frac{1}{10}\sum_{t=0.50}^{0.95} \text{mAP}\ @t \tag{9}$$

where t is the IoU threshold, varying from 0.50 to 0.95 in 0.05 increments.

Concurrently, the qualitative verification is conducted by evaluating the model's inference performance across different object scales, specifically categorized into small-, medium-, and large-scale ship objects. This comprehensive evaluation methodology ensures that the final evaluation captures both statistical macro-performance and granular real-world localization robustness. All metric calculations and visual inferences are performed automatically through the Ultralytics YOLO framework's built-in evaluation pipeline, which is standardized and aligned with international computer vision benchmark protocols.

G. Model Evaluation

In the model evaluation phase, a scientific comparison was conducted to assess the consistency and robustness of performance between YOLO12-L without the attention module and YOLO12-L with the Efficient Channel Attention (ECA) module. Unlike conventional parametric statistical approaches, such as the t-test, which assume normality in evaluation metrics, this study used the more robust Paired Bootstrap Median Test to assess the significance of performance differences [26].

The bootstrap median method was chosen because it can generate confidence interval estimates without requiring specific distributional assumptions and is more resilient to outliers and asymmetric distributions, making it particularly suitable for deep learning metrics whose distributions are often non-normal in small samples [26]. The bootstrap procedure was performed using a resampling technique of 100 iterations of five pairs of matrix values obtained from identical random seed variations in both models.

Mathematically, the median bootstrap procedure for calculating a 95% confidence interval can be formulated as follows. Suppose there are two paired samples $X = \{x_1,x_2,...,x_n\}$ and $Y = \{y_1,y_2,...,y_n\}$ with $n = 5$ seed pairs, where xi is the mAP@50 of the baseline model, and y_i is the mAP@50 of the ECA model at the i^{th} seed. The median of each sample is defined as follows.

$$\tilde{x} = median(X), \tilde{y} = median(Y) \tag{10}$$

The median difference is defined as:

$$\Delta = \tilde{y} - \tilde{x} \tag{11}$$

At each bootstrap iteration b (with $b = 1,2,...,B$ and $B = 100$), random sampling with replacement is performed from X and Y to produce bootstrap samples X^{*b} and Y^{*b}. The median difference is calculated for each pair of bootstrap samples using the following formula.

$$\Delta^{*b} = median(Y^{*b}) - median(X^{*b}) \tag{12}$$

The distributions of $\Delta^{*1},\Delta^{*2},...,\Delta^{*B}$ are then used to construct 95% confidence intervals using the BCa (Bias-Corrected and Accelerated) method, which corrects for bias and skewness in the bootstrap distribution [26]. The lower and upper limits of the confidence intervals are determined by the adjusted percentiles of the bootstrap distribution, as in the following equation.

$$CI_{95\%} = [\Delta^{*(\alpha_1)}, \Delta^{*(\alpha_2)}] \tag{13}$$

where α_1 and α_2 are percentiles corrected for bias and acceleration factors.

The hypotheses tested are:

- H_0: $\theta_2 - \theta_1 = 0$ (no difference in median performance)
- H_1: $\theta_2 - \theta_1 \neq 0$ (there is a difference in median performance)

where θ_1 and θ_2 are the population medians of the baseline and ECA models, the p-values were calculated using the shift

bootstrap method with median-centering under the null hypothesis [12].

If the 95% confidence interval does not include zero and the p-value is <0.05, then the difference in median performance is statistically significant.

In addition to significance testing, the evaluation also includes an analysis of model stability by comparing the standard deviations of evaluation metrics between seeds. A decrease in performance variability in the ECA model relative to the baseline will be interpreted as an increase in the model's robustness to random initialization variations, a critical aspect for deploying object detection systems in dynamic operational environments.

III. Results and Discussion

After going through the stages of dataset collection, data preprocessing, integration of the Efficient Channel Attention module, and implementation of two stages of training experiments with a multi-seed configuration, this section presents the results and discussion of the research that has been conducted. The results obtained from training the YOLO12-L model, both the baseline version and the one that has been integrated with ECA, were then evaluated through key performance metrics and statistically tested to determine the significance of the differences. In addition, qualitative analysis through visualization of detection results was also conducted to provide interpretive context for the quantitative findings.

A. Data Augmentation Results

The data augmentation phase, executed on-the-fly during the training process, successfully produced a variety of training images with higher visual diversity than the original dataset. Each training batch had the probability of transforming, such as horizontal flips, brightness and contrast adjustments through the HSV color space, and minor object position shifts and scaling. With this variation, the model gained exposure to a broader data distribution, thus improving its generalization ability when tested with previously unseen data. A sample of the data augmentation results can be seen in Figure 4.

Based on Figure 4, the visualization of the augmentation results shows that the vessel object can appear in visual conditions different from the original image without changing its geometric structure or aspect ratio. For example, a vessel initially centered in the frame can be shifted laterally, subjected to changes in lighting intensity, or displayed in a mirrored orientation without losing its structural detail. These transformations are intentionally designed to be of limited intensity to preserve the optical characteristics of maritime aerial imagery, while still introducing enough variation to prevent the model from memorizing static patterns (overfitting).

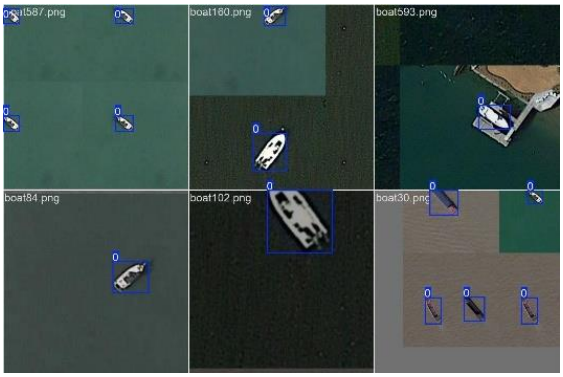


Figure 4. Visualization of Augmentation Results

The impact of this augmentation is clearly visible in the initial training dynamics, where the model exhibits more stable convergence and a consistent reduction in validation loss. The basic augmentation strategy successfully enriched the feature representation without shifting the original domain distribution, allowing the model to not rely solely on fixed visual patterns but also learn to recognize vessels in more realistic variations in lighting, viewpoints, and spatial positions. This approach is particularly relevant given that operational conditions in the field often involve reflections of water surfaces (sun glint), variations in solar intensity, and vessel positions that are not always centered in the image.

B. Model Design Results

At this stage, the YOLO12-L architecture, described in Chapter II, integrates the Efficient Channel Attention (ECA) module to improve the selectivity of channel features in ship object detection. The model architecture design focused on the structured insertion of the ECA module into the YOLO12-L feature processing pipeline, particularly in the medium-resolution backbone layer. This modification is designed to strengthen the representation of channel features relevant to ship objects without disrupting the skip connection mechanism or multi-scale feature distribution in the detection stage.

The designed model architecture consists of three main components with a total of 24 processing layers. In the backbone, the input image is processed through a series of convolutional layers and a C3K2 block for hierarchical feature extraction, followed by an A2C2f (Advanced C2f) block that deepens the spatial representation. Table II presents a summary of the implemented architecture configuration.

TABLE II.
YOLO12 + ECA ARCHITECTURAL CONFIGURATION

Layer	Module	Output Channel
Backbone		
0-1	Conv	64 → 128
2	C3K2	256
3-4	Conv + C3K2	512
5-6	Conv + A2C2f	512
7-8	Conv + A2C2f	1024

Neck		
9-14	Upsample + Concat + A2C2f	256-512
15-17	Conv + Concat + A2C2f	512
18	**ECA**	**512**
19-22	A2C2f + Conv + Concat	512-1024
Head		
23	Detect	nc=1

Based on Table II, the Efficient Channel Attention (ECA) module is integrated at layer 18 of the architecture, directly after the A2C2f block at feature level P4 (Feature Pyramid Level 4), which has 512 channels. The integration of the ECA module at this specific layer is governed by three critical technical factors regarding the Feature Pyramid Network (FPN) hierarchy and the visual characteristics of ship objects in aerial datasets.

First, regarding object scale alignment, in the YOLO architecture, level P3 handles small objects, P4 handles medium-sized objects, and P5 handles large objects. In remote sensing datasets, ship objects generally occupy a medium-sized pixel area (around 10–20% of the image size). They are rarely very small objects (such as pedestrians) or very large objects that fill the screen. Level P4 represents the optimal point where spatial features still retain sufficient resolution to localize the ship's shape, but semantic information is mature enough to distinguish it from the background.

Second, in terms of semantic feature maturity, placing the attention module at the backbone (beginning) risks disrupting basic feature extraction (edges, textures), while placing it at the head (end) often results in delays due to the drastic reduction in spatial information. The P4 level at the neck is strategically located where features have already gone through a fairly deep convolution block (A2C2f), so the feature maps are already rich in semantic meaning but have not undergone the extreme downsampling of P5. The ECA mechanism here functions as a "filter gate" that filters out background ocean noise (such as wave patterns or sun glint) before concatenating the features for the final prediction.

To empirically validate this "filter gate" capability and satisfy the structural verification of the attention mechanism, Figure 5 illustrates the visual transition of the intermediate feature maps before and after the ECA execution.

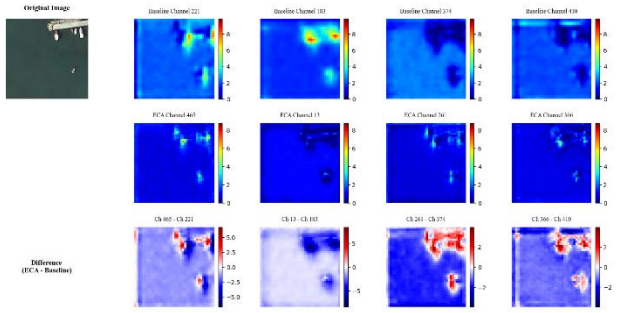


Figure 5. Feature Map Visualization

As demonstrated in Figure 5, the visual evidence confirms that the baseline feature maps (a) contain substantial background clutter, where high activation intensities (bright regions) are broad and unfocused, mistakenly tracking wave Crests and sun glint distortions on the sea surface. Conversely, upon passing through the ECA block (b), the activation maps exhibit clear spatial contraction and channel refinement. The energy paths become highly concentrated on the precise geometric boundaries of the vessel hulls, while the surrounding maritime background noise is visibly suppressed and attenuated. This visual confirmation proves that the ECA module successfully forces the network to prioritize discriminative ship features over complex environmental distractors.

Third, concerning computational efficiency and noise propagation reduction, focusing the attention selection on a single dominant representation path at the P4 level prevents the linear inflation of parameters that occurs when installing attention mechanisms across all blocks (P3, P4, and P5). To substantiate the claim that this architectural modification offers a highly lightweight solution viable for real-time edge deployment, Table III provides a comprehensive comparative breakdown of the network architecture complexities.

TABLE III
COMPUTATIONAL COMPLEXITY AND OVERHEAD ANALYSIS

Model	Parameters	FLOPs	Inference Speed	Overhead
YOLO12	26.389.875	89.4	11.0 FPS	-
YOLO12 + ECA	26.389.880	89.4	10.7 FPS	~2.5%

The empirical measurements in Table III confirm the computational efficiency of the proposed architecture. The integration of the ECA module introduces a negligible increase of 5 parameters to the baseline network (expanding from 26.389.875 to 26.389.880). Furthermore, the overall floating-point operations remain identical at 89.4 GFLOPs, indicating that the 1D convolution operation within the ECA channel-weight generation branch adds virtually no computational strain. In terms of runtime throughput, the inference speed decreases by 0.3 FPS (from 11.0 to 10.7 FPS), corresponding to a very marginal processing overhead of approximately 2.5%. This balance confirms that the model maintains its operational throughput, demonstrating that the

architectural modification introduces profound feature selectivity at nearly zero computational cost.

To provide a visual representation of the integration flow, the resulting architectural diagram is presented in Figure 6 below.

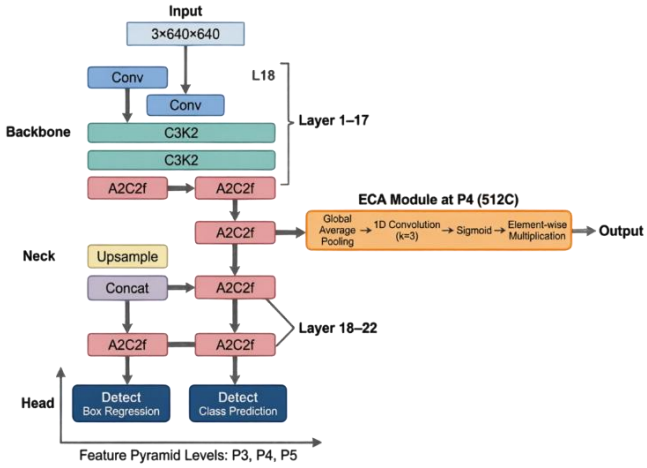


Figure 6. ECA Architecture Implementation on YOLO12

Figure 6 shows the complete feature processing flow, from image input to feature extraction on the backbone, multi-scale fusion on the neck, and final prediction on the head. The integration of the ECA module is explicitly demonstrated at layer 18, placed on the neck branch at level P4 (512 channels).

The diagram shows that the feature maps output from the previous A2C2f block are directed to the ECA side-branch pathway, where spatial information is compressed into channel vectors through Global Average Pooling, then processed by 1D convolution and sigmoid activation to generate attention weights. These weights are then returned to the main pathway through element-wise multiplication, so that the channel-selected features are directly forwarded to the next A2C2f block. This placement visually confirms that the ECA modules operate in parallel without interrupting the gradient flow or disrupting the concatenation mechanism in the Feature Pyramid Network, ensuring that the multi-scale detection structure (P3, P4, P5) remains functional as originally designed.

Overall, the model design results confirm that the integration of the Efficient Channel Attention module into the YOLO12-L architecture can be implemented modularly, efficiently, and without compromising the integrity of the baseline structure. The placement of ECA at layer 18 at the P4 level has proven strategic for selecting dominant mid-scale feature channels in ship objects, while the very marginal parameter addition and computational overhead ensure that the model remains viable for real-time detection applications. With this architecture visually and mathematically verified, the research is ready to move on to the empirical training and testing phase to measure the actual impact of the attention mechanism on improving detection performance.

C. Model Training Results

The training process for the two experimental scenarios was conducted entirely from scratch to ensure a fair and methodologically sound comparison between the baseline architecture and the proposed ECA-integrated variant. By initializing all network weights randomly, this approach isolates the architectural contribution of the attention module without relying on transfer learning priors from external datasets. Both the YOLO12-L Baseline and YOLO12-L + ECA models required extended training durations, reaching stable convergence between the 350th and 400th epochs. This pattern indicates that learning hierarchical features directly from the maritime aerial imagery dataset necessitates a broader exploration of the parameter space before the models can effectively generalize to vessel detection tasks. The dynamics of this optimization process are illustrated in the training and validation loss curves presented in Figure 7 below.

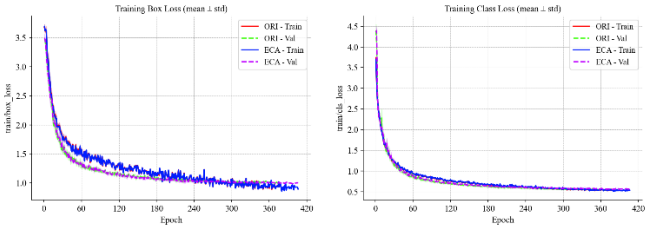


Figure 7. Training and Validation Loss Graphs of Baseline and ECA

Figure 7 shows that the training box loss and classification loss values decreased drastically in the first 60 epochs for both models, indicating an aggressive early learning phase. The graph also shows that the model with ECA integration (purple line) has a slightly more stable validation loss than the baseline (green line) after the 180th epoch, especially for the box loss, which is related to the accuracy of the bounding box localizer. Smaller fluctuations in the ECA validation curve indicate that the feature channel filtering mechanism helps the model avoid overfitting and maintains better generalization. The evolution of the evaluation metrics during training is shown in Figure 8 below.

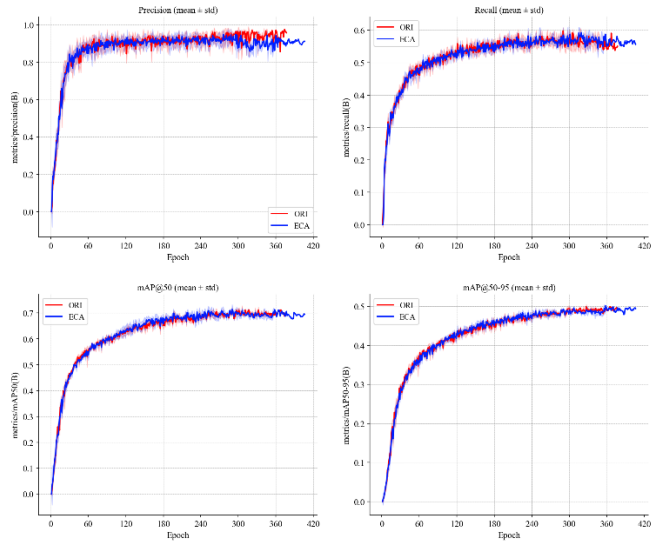


Figure 8. Development of Evaluation Metrics During Training

Figure 8 shows that the precision and recall metrics increase sharply in the first 60 epochs, then reach a relatively stable plateau. The ECA model exhibits a slightly higher recall pattern than the baseline in the 240–360 epoch range, consistent with the attention module's ability to detect more ship objects, especially small ones. However, the absolute difference between the two models is not striking in this aggregate graph, necessitating a more in-depth quantitative analysis using the final summary metrics.

The mAP@50 and mAP@50-95 graphs at the bottom of Figure 8 show parallel convergence patterns between the baseline and ECA models, with both metrics reaching saturation at epochs 300–350. The gradual increase in mAP@50-95 indicates that the model not only learns to detect objects (IoU ≥ 0.5) correctly but also improves the accuracy of its bounding box localization at more stringent IoU thresholds (up to 0.95). The stability of the curve in the final training phase (>350 epochs) was the basis for setting a duration of 500 epochs to ensure optimal convergence of the model. This is also supported by the final average value and standard deviation, as shown in Table IV below.

TABLE IV
SUMMARY OF FINAL EPOCH VALUES (MEAN ± STD)

Metric	YOLO12-L	YOLO12-L + ECA
precision(B)	0.911 ± 0.029	0.925 ± 0.028
recall(B)	0.579 ± 0.011	0.569 ± 0.014
mAP@50(B)	0.705 ± 0.011	0.715 ± 0.003
mAP@50-95(B)	0.491 ± 0.006	0.495 ± 0.003

Table IV shows the performance comparison between YOLO12-L and YOLO12-L + ECA, both trained from scratch without pretrained COCO weights to eliminate transfer learning bias. By turning off pretrained initialization (pretrained = False), the model's convergence ability was measured purely based on the visual characteristics of the ship dataset, ensuring that the architectural contribution of ECA could be assessed independently. In this comparison, the ECA integration resulted in a 1.0 percentage point increase in mAP@50, from 70.5% ± 1.1% to 71.5% ± 0.3%.

This improvement was also accompanied by an increase in precision, from 91.1% ± 2.9% to 92.5% ± 2.8%, indicating that the feature channel filtering mechanism helped the model produce more accurate positive predictions. On the other hand, recall decreased marginally from 57.9% ± 1.1% to 56.9% ± 1.4%, reflecting the natural trade-off between precision and recall: models with ECA tend to be more selective in their detections, thus reducing false positives but slightly increasing the risk of false negatives for very small or partially occluded objects.

The observed variance patterns in the ECA metrics also provide important insights. The lower standard deviation of mAP@50 in the ECA model (±0.3% vs. ±1.1% in the baseline) reflects the attention module's stability under random initialization, with all seeds producing more consistent performance. This phenomenon further reinforces the importance of the multi-seed training protocol implemented in this study, as reporting results from a single seed risks yielding unrepresentative conclusions about the model's true generalization ability.

Overall, the training results confirm three key findings: (1) training from scratch requires a longer optimization duration (~500 epochs) to reach learning saturation, as the model must independently construct hierarchical feature representations without prior domain knowledge; (2) the integration of the ECA module yields consistent improvements in accuracy metrics (mAP@50 and precision) while substantially reducing performance variance across seed variations; and (3) the attention mechanism effectively stabilizes the validation loss trajectory, indicating enhanced generalization and reduced susceptibility to stochastic initialization effects. These observations validate the robustness of the multi-seed scratch training protocol and provide a solid foundation for the subsequent evaluation on unseen test data and statistical significance testing.

D. Test Results on Test Data

After the training process was completed, the two models were evaluated using 124 test images not used during training. The evaluation was performed five times using the predetermined random seed values (42, 123, 456, 1234, and 3407) to ensure the consistency and reproducibility of the results. Tables V and VI show the performance results of each model for each seed used, while the comparative analysis is summarized in Table VII.

TABLE V.
YOLO12-L TEST RESULTS

Seed	mAP@50	mAP@50-95	Precision	Recall
42	0.7025	0.4914	0.9157	0.5872
123	0.7031	0.4953	0.9607	0.5705
456	0.7087	0.488	0.9171	0.5671
1234	0.7237	0.5015	0.8839	0.5973
3407	0.6897	0.4818	0.8781	0.5772
Mean ±Std	0.705 ± 0.011	0.491 ± 0.006	0.911 ± 0.029	0.579 ± 0.011

Based on Table V and Table VI, the model with ECA integration shows better performance than the baseline. The ECA model achieves an average mAP@50 value of 71.5% ± 0.3% with a lower standard deviation than the baseline of ±1.1%, indicating more consistent stability across the five random seed variations. The 1.4% increase in precision (from 91.1% to 92.5%) indicates that the channel attention mechanism successfully helps the model produce more accurate positive predictions.

TABLE VI.
YOLO12-L + ECA TEST RESULTS

Seed	mAP@50	mAP@50-95	Precision	Recall
42	0.7224	0.5006	0.9604	0.57
123	0.7112	0.494	0.9119	0.556
456	0.7157	0.4902	0.8749	0.6101
1234	0.7132	0.4927	0.962	0.5436
3407	0.7125	0.4995	0.9033	0.5671
Mean ±Std	0.715 ± 0.003	0.495 ± 0.003	0.925 ± 0.028	0.569 ± 0.014

To place the contribution of this research in the context of developments in the field of ship detection, Table VII presents a comparison of the results with previous studies.

TABLE VII
COMPARISON OF RESEARCH RESULTS

Author (Year)	**Metode**	**Precision**	**Recall**	**mAP@50**	**mAP@50-95**
Patel et al. (2022) [16]	YOLOv3	0.71	0.44	0.49	-
	YOLOv4	0.67	0.59	0.61	-
	YOLOv5	0.7	0.63	0.65	-
Das & Aravinth (2025) [2]	YOLOv8	0.52	0.42	0.44	0.26
This research	YOLO12	0.911±0.029	0.579±0.011	0.705±0.011	0.491±0.006
	YOLO12 + ECA	0.925± 0.028	0.569±0.014	0.715±0.003	0.495±0.003

Based on Table VII, this study demonstrates superior performance compared to previous studies. The mAP@50 value of 71.5% exceeds the results achieved by Patel et al. (2022), which ranged from 49% to 65% using YOLOv3, YOLOv4, and YOLOv5, and Das & Aravinth (2025), which achieved 44% with YOLOv8. Even the YOLO12-L baseline (70.5% mAP@50) outperforms these prior works, underscoring the inherent advantages of the YOLO12-L architecture for maritime object detection.

To complement these quantitative metrics with qualitative insights across varying operational complexities, Table VIII presents a multi-scale visual evaluation of the prediction outputs. This qualitative testing was conducted using a Python-based inference script on the test dataset to evaluate generalization across small, medium, and large targets under standard confidence thresholds of 0.25 at 640x640 pixels.

TABLE VIII
COMPARISON OF DETECTION RESULTS ON VARIOUS SCALES

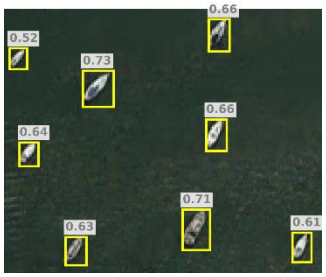
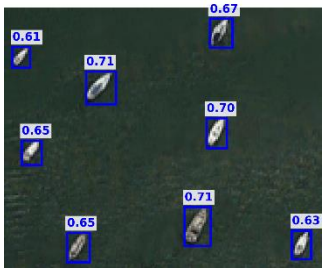
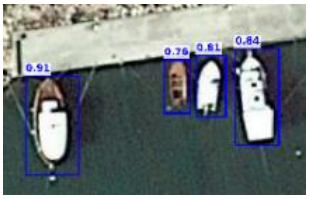
Model	Small	Medium	Large
GT			
YOLO12-L			
YOLO12-L + ECA			

Table VIII employs a standardized color scheme to facilitate clear interpretation: green bounding boxes represent Ground Truth (GT), yellow boxes indicate detections by the YOLO12-L baseline, and blue boxes denote the proposed YOLO12-L + ECA model. As demonstrated in Table VIII, the YOLO12-L + ECA model exhibits markedly superior detection reliability, with consistent improvements in confidence calibration across all object scales.

In the Small scale category, the ECA-integrated model demonstrates a critical improvement in worst-case detection reliability. While the Baseline occasionally produces low-confidence predictions for small vessels (minimum confidence: 0.52), the ECA model elevates this minimum threshold to 0.61—a relative improvement of 17.3% in the lower bound of detection certainty. This shift indicates that the ECA module effectively suppresses background noise while amplifying discriminative structural cues of smaller objects, thereby reducing the risk of missed detections. For the Medium scale, particularly in high-density docking environments, the ECA model achieves substantially higher and more stable confidence scores. In clustered scenarios where the Baseline exhibits moderate detection certainty (~0.62), the ECA model attains significantly elevated confidence levels ranging from 0.71 to 0.83. This represents an improvement of up to 33.9% in detection certainty, confirming that the channel attention mechanism effectively disentangles overlapping features in crowded scenes. In the Large scale category, the ECA model maintains the high-precision localization capability of the baseline while adding an additional layer of feature-level stability, sustaining confidence scores ≥ 0.90 for large cargo ships.

The qualitative evidence gathered across these diverse operational conditions confirms that integrating the ECA module enhances the model's robustness to optical distortions and scale variations without degrading overall spatial alignment.

E. Statistical Analysis (Median Bootstrap Test)

To validate the consistency of the results obtained, this study applies the Paired Bootstrap Median Test method with 100 resampling iterations using the BCa (Bias-Corrected and Accelerated) method. This method was chosen because it is more robust than standard parametric statistical tests, especially in handling Deep Learning evaluation metrics whose distributions are not always normal and have high variability in small samples, as well as utilizing the paired nature of data generated by identical seeds [12], [26]. The results of the estimated performance differences along with

95% confidence intervals (95% CI) are summarized in Table IX.

TABLE IX
MEDIAN BOOTSTRAP TEST RESULTS

Metric	Median		Difference	p-value
	Baseline	ECA		
mAP@50	0.7031	0.7132	+0.0101 (+1.01%)	0.02
mAP@50-95	0.4914	0.4940	+0.0026 (+0.26%)	0.59
Precision	0.9157	0.9119	+0.0038 (+0.38%)	0.69
Recall	0.5772	0.5671	-0.0101 (-1.01%)	0.349

Based on Table IX, the median mAP@50 for the Baseline model was recorded at 0.7031, while the median for the ECA model was 0.7132. The median difference of +0.0101 (+1.01%) indicates that the ECA model has a higher median performance than the baseline model.

The hypothesis test results show a p-value of 0.02, which is less than the $\alpha = 0.05$ significance level. This indicates that the difference in median performance between the Baseline Scratch model and the ECA-integrated model is statistically significant. Therefore, the null hypothesis ($H_0 : \theta_2 - \theta_1 = 0$) is rejected, and it can be concluded that the ECA module integration provides a significant median performance improvement at the 95% confidence level.

The significant p-value (0.02) provides strong evidence that the median improvement of +1.01% is not simply due to random variations in weight initialization, but rather reflects a significant contribution from the Efficient Channel Attention mechanism in improving channel feature selectivity. This finding is in line with the characteristics of the ECA module, which is designed to strengthen the feature representation in channels relevant to the ship object while suppressing the response in channels containing marine background noise.

This finding is reinforced by the model stability analysis, which shows a drastic decrease in the standard deviation of mAP@50 from ±1.1% in the Baseline model to ±0.3% in the ECA model, representing a 72.7% reduction in performance variability. This phenomenon confirms that the ECA mechanism functions effectively as a feature selection filter, significantly improving median performance and stabilizing the model parameter space against random initialization variations.

The Baseline model tends to have performance that is highly dependent on specific weight initializations (e.g., seed 1234 achieves 0.7237, but seed 3407 drops to 0.6897). In contrast, the ECA model narrows this performance range (0.7112 to 0.7224), ensuring consistent, reproducible results regardless of the initial weight initialization.

In the context of real-time ship detection applications, the combination of a statistically significant median performance improvement and significantly improved stability is crucial. A system with high median accuracy and low variance ensures consistent and reliable output in dynamic operational environments, where lighting conditions, wave textures, and object occlusions can change unpredictably. Therefore, the integration of ECA has been shown to improve system reliability by strengthening feature generalization and significantly improving median performance.

IV. CONCLUSION

This study successfully implemented and evaluated the integration of the Efficient Channel Attention (ECA) module into the YOLO12-L architecture to enhance ship object detection in aerial imagery. By employing a multi-seed scratch training protocol and rigorous statistical validation via the Paired Bootstrap Median Test, this research not only quantified detection improvements but also explicitly verified the statistical significance of the performance gains and the model's robustness against random initialization variability.

Experimental results demonstrated that strategically embedding the ECA module at the P4 feature level increased the median mAP@50 by 1.01% (from 0.7031 to 0.7132) and enhanced the mean precision by 1.4% relative to the baseline model. Crucially, the Paired Bootstrap Median Test confirmed that this improvement is statistically significant ($p = 0.02 < 0.05$), indicating that the performance improvement stems directly from the architectural modification rather than stochastic weight initialization effects.

Beyond accuracy extensions, the most notable contribution of this research lies in the substantial enhancement of model stability. Upon integrating the ECA module, the mAP@50 standard deviation dropped sharply from ±1.1% to ±0.3%, representing a 72.7% reduction in performance variance. This empirical outcome confirms that the channel-wise feature filtering mechanism effectively stabilizes the model's parameter space, yielding highly consistent and reproducible predictions across diverse seeds. The synergy between statistically significant precision gains and enhanced operational stability makes the proposed architecture highly viable for real-time maritime monitoring applications.

Qualitative validation through inference testing further supported these quantitative metrics, demonstrating the model's ability to localize vessels while accurately suppressing complex marine background clutter. However, a minor trade-off in recall was observed, attributable to the attention module's highly selective feature-gating mechanism.

Despite these robust findings, the study's limitations include a modest improvement in overall mAP, a slight recall reduction for ultra-small vessels, and the absence of a direct empirical comparison with alternative attention modules such as CBAM, SE, or Coordinate Attention on this specific dataset. Additionally, the evaluation scope remains restricted

to static aerial imagery under standard lighting conditions. Future work should explore hybrid spatial-channel attention mechanisms or multi-scale feature fusion to balance the precision-recall trade-off effectively. Furthermore, expanding dataset diversity to encompass extreme weather conditions and dynamic video streams, and pursuing hardware-aware compression, will be vital steps toward deploying energy-efficient, edge-ready maritime surveillance systems.

REFERENCES

[1] Vitor G. Santos, Diego S. Pereira, Luis B. P. Nascimento, and Pablo J. Alsina, "CNN-based Boat Detection for Environmental Protection Area Monitoring," presented at the XXIV Congresso Brasileiro de Automática, Online, Oct. 2022. doi: 10.20906/CBA2022/3599.

[2] S. Das and Aravinth R, "Navigating the Future: Intelligent Ship Detection through Multisensor Imagery and DeepLearning," *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. X-5/W2-2025, pp. 137–142, Dec. 2025, doi: 10.5194/isprs-annals-X-5-W2-2025-137-2025.

[3] F. Hermens, "Automatic object detection for behavioural research using YOLOv8," *Behav Res*, vol. 56, no. 7, pp. 7307–7330, May 2024, doi: 10.3758/s13428-024-02420-5.

[4] X. Wang *et al.*, "Ship feature recognition methods for deep learning in complex marine environments," *Complex Intell. Syst.*, vol. 8, no. 5, pp. 3881–3897, Oct. 2022, doi: 10.1007/s40747-022-00683-z.

[5] A. Galdelli, G. Narang, R. Pietrini, M. Zazzarini, A. Fiorani, and A. N. Tassetti, "Multimodal AI-enhanced ship detection for mapping fishing vessels and informing on suspicious activities," *Pattern Recognition Letters*, vol. 191, pp. 15–22, May 2025, doi: 10.1016/j.patrec.2025.02.022.

[6] Y. Tan, J. Song, and C. Chu, "Detection of Small Object based on Improved-YOLOv8," *FCIS*, vol. 10, no. 3, pp. 79–85, Dec. 2024, doi: 10.54097/n5rtnt71.

[7] J. He and S. Luo, "An Improved Model Based on YOLOv8 for Small Object Detection and Recognition," *Information*, vol. 17, no. 2, p. 173, Feb. 2026, doi: 10.3390/info17020173.

[8] Ultralytics, "YOLO12: Attention-Centric Object Detection." Accessed: May 10, 2026. [Online]. Available: https://docs.ultralytics.com/models/yolo12/

[9] T. Ge, B. Ning, and Y. Xie, "YOLO-AFR: An Improved YOLOv12-Based Model for Accurate and Real-Time Dangerous Driving Behavior Detection," *Applied Sciences*, vol. 15, no. 11, p. 6090, May 2025, doi: 10.3390/app15116090.

[10] R. F. Puspita, M. Naufal, and F. Al Zami, "Improving YOLO Performance with Advanced Data Augmentation for Soccer Object Detection," *JAIC*, vol. 9, no. 6, pp. 3601–3611, Dec. 2025, doi: 10.30871/jaic.v9i6.11256.

[11] P. Selvam, P. Shanmuga Sundari, M. Tamilselvi, T. Suresh, M. Murugappan, and M. E. H. Chowdhury, "YOLO-SAIL: Attention-Enhanced YOLOv5 With Optimized Bi-FPN for Ship Target Detection in SAR Images," *IEEE Access*, vol. 13, pp. 29523–29540, 2025, doi: 10.1109/ACCESS.2025.3536621.

[12] P. C. Austin, I. Eekhout, and S. Van Buuren, "Evaluating the median *p* -value method for assessing the statistical significance of tests when using multiple imputation," *Journal of Applied Statistics*, vol. 52, no. 6, pp. 1161–1176, Apr. 2025, doi: 10.1080/02664763.2024.2418473.

[13] F. Yuan, X. Gao, and C. Zhang, "Three-Dimensional Model Classification Based on VIT-GE and Voting Mechanism," *CMC*, vol. 85, no. 3, pp. 5037–5055, 2025, doi: 10.32604/cmc.2025.067760.

[14] M. Xue, M. Chen, D. Peng, Y. Guo, and H. Chen, "One Spatio-Temporal Sharpening Attention Mechanism for Light-Weight YOLO Models Based on Sharpening Spatial Attention," *Sensors*, vol. 21, no. 23, p. 7949, Nov. 2021, doi: 10.3390/s21237949.

[15] H. Ahn *et al.*, "SAFP-YOLO: Enhanced Object Detection Speed Using Spatial Attention-Based Filter Pruning," *Applied Sciences*, vol. 13, no. 20, p. 11237, Oct. 2023, doi: 10.3390/app132011237.

[16] K. Patel, C. Bhatt, and P. L. Mazzeo, "Deep Learning-Based Automatic Detection of Ships: An Experimental Study Using Satellite Images," *J. Imaging*, vol. 8, no. 7, p. 182, Jun. 2022, doi: 10.3390/jimaging8070182.

[17] E. Yildirim and T. Kavzoglu, "Ship Detection in Optical Remote Sensing Images Using YOLOv4 and Tiny YOLOv4," in *Innovations in Smart Cities Applications Volume 5*, vol. 393, M. Ben Ahmed, A. A. Boudhir, İ. R. Karaș, V. Jain, and S. Mellouli, Eds., in Lecture Notes in Networks and Systems, vol. 393. , Cham: Springer International Publishing, 2022, pp. 913–924. doi: 10.1007/978-3-030-94191-8_74.

[18] A. A. D. Go *et al.*, "Comprehensive Benchmark of Yolov11n, SSD MobileNet, CenterFace, Yunet, FastMtCnn, HaarCascade, and LBP for Face Detection in Video Based Driver Drowsiness," vol. 7, no. 3, 2025.

[19] M. N. Andrean *et al.*, "Comparing Haar Cascade and YOLOFACE for Region of Interest Classification in Drowsiness Detection," *mib*, vol. 8, no. 1, p. 272, Jan. 2024, doi: 10.30865/mib.v8i1.7167.

[20] R. D. L. Rocha and F. A. P. D. Figueiredo, "Enhancing YOLO-Based SAR Ship Detection with Attention Mechanisms," *Remote Sensing*, vol. 17, no. 18, p. 3170, Sep. 2025, doi: 10.3390/rs17183170.

[21] Y. Li *et al.*, "SOD-YOLO: Small-Object-Detection Algorithm Based on Improved YOLOv8 for UAV Images," *Remote Sensing*, vol. 16, no. 16, p. 3057, Aug. 2024, doi: 10.3390/rs16163057.

[22] Z. Xu, Y. Yang, Y. Wei, and O. L. Magnagna, "Improvement of YOLOv8 for Vehicle Small Object Detection Research," in *International Conference on Artificial Intelligence, Automation and High Performance Computing*, Zhuhai China: ACM, Jul. 2024, pp. 84–89. doi: 10.1145/3690931.3690946.

[23] B. Khalili and A. W. Smyth, "SOD-YOLOv8—Enhancing YOLOv8 for Small Object Detection in Aerial Imagery and Traffic Scenes," *Sensors*, vol. 24, no. 19, p. 6209, Sep. 2024, doi: 10.3390/s24196209.

[24] M. Kim, J. Jeong, and S. Kim, "ECAP-YOLO: Efficient Channel Attention Pyramid YOLO for Small Object Detection in Aerial Image," *Remote Sensing*, vol. 13, no. 23, p. 4851, Nov. 2021, doi: 10.3390/rs13234851.

[25] W. Luo and S. Yuan, "Enhanced YOLOv8 for small-object detection in multiscale UAV imagery: Innovations in detection accuracy and efficiency," *Digital Signal Processing*, vol. 158, p. 104964, Mar. 2025, doi: 10.1016/j.dsp.2024.104964.

[26] F. García Fernández, P. De Palacios, A. García-Iruela, and L. G. Esteban, "Using Bootstrapping to Determine Artificial Neural Network Confidence Intervals—Case Study of Particleboard Internal Bond Determined from Production Data," *Applied Sciences*, vol. 15, no. 8, p. 4554, Apr. 2025, doi: 10.3390/app15084554.

[27] "Ship Detection from Aerial Images." Accessed: May 11, 2026. [Online]. Available: https://www.kaggle.com/datasets/andrewmvd/ship-detection