

## Article

# MCSwin-YOLOv8: Multi-Scale Feature Learning for Maritime Ship Detection

Yuqing Ren <sup>1,†</sup>, Guohao Wen <sup>1,†</sup>  and Yingbang Huang <sup>2,3,\*</sup>

<sup>1</sup> School of Navigation and Naval Architecture, Dalian Ocean University, Dalian 116023, China; renyuqing@dlou.edu.cn (Y.R.); wgh\_dlou@163.com (G.W.)

<sup>2</sup> South China Sea Fisheries Research Institute, Chinese Academy of Fishery Sciences, Guangzhou 510300, China

<sup>3</sup> College of Ocean and Meteorology, Guangdong Ocean University, Zhanjiang 524088, China

\* Correspondence: hkbone99@163.com

† These authors contributed equally to this work.

## Abstract

Maritime ship detection remains challenging because of large scale variations, high inter-class visual similarity, weak target boundaries, and complex maritime backgrounds. This study proposes MCSwin-YOLOv8, an enhanced YOLOv8-based detector that combines three complementary architectural designs. First, a re-parameterizable multi-scale convolutional backbone, named RepMCSwin, is introduced to extract scale-aware semantic information and fine-grained boundary cues. Unlike the standard Swin Transformer, the MCSwin block does not use window-based self-attention but adopts cascaded multi-scale convolutions and residual feature transformation. Second, a Multi-Feature Parallel Convolutional Block Attention Module (MFPCBAM) is developed to compute channel and spatial attention in parallel, thereby preserving weak ship features while suppressing irrelevant background responses. Third, a Modified Generalized Feature Pyramid Network (MGFPN) is constructed to improve cross-level feature interaction and retain high-resolution spatial information through an additional  $160 \times 160$  prediction branch. Experiments were conducted on the public SeaShips dataset and a private infrared maritime ship dataset. MCSwin-YOLOv8 achieved an F1-score of 94.4%, mAP@0.5 of 97.4%, and mAP@0.5:0.95 of 75.1% on SeaShips. On the infrared dataset, the corresponding results were 91.2%, 94.1%, and 66.9%, respectively. Compared with the YOLOv8 baseline, mAP@0.5 increased by 1.4 and 3.0 percentage points on the two datasets. These accuracy gains were accompanied by an increase in model complexity from 11.12 M to 19.29 M parameters and from 28.5 G to 56.7 G FLOPs, indicating an accuracy–complexity trade-off that requires further runtime evaluation.

**Keywords:** YOLOv8; RepMCSwin backbone; MFPCBAM; MGFPN; ship target detection



Academic Editor: José A. Orosa García

Received: 16 July 2026

Revised: 20 August 2026

Accepted: 22 August 2026

Published: 24 August 2026

**Copyright:** © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

## 1. Introduction

In recent years, the rapid growth of the shipping industry has substantially increased maritime trade activities. At the same time, ship-related accidents have become more frequent in complex traffic waters, such as coastal ports and fishing harbor areas, posing increasing challenges to maritime traffic safety [1]. To address this issue, extensive research has been conducted on deep learning-based ship target detection. Advances in image recognition have significantly improved the capability of automatic ship detection, providing technical support for shipping operations and maritime surveillance. However, two

key challenges remain. First, ship detection in multi-class scenarios is strongly affected by the high visual similarity among ship categories, dense target distributions, and large variations in target scale. Second, existing detection algorithms still struggle to achieve a satisfactory balance between detection accuracy and computational efficiency.

Traditional ship target detection mainly relies on radar, AIS, and other sensing devices. However, such methods suffer from limitations, including the inability to observe ship types directly and the failure to identify vessels with AIS turned off or with signal loss. Vision-based detectors identify vessels directly from maritime imagery, providing category and appearance information that is generally unavailable from conventional sensing systems. Ke et al. [2] incorporated self-attention into a neural detector to refine feature responses, leading to improved detection accuracy. Li et al. [3] proposed an enhanced Swin Transformer network, which improved the fusion efficiency of adjacent features in the feature pyramid, thereby enabling more efficient transmission and utilization of information. Zha et al. [4] used multi-feature transformation and fusion to reduce background-induced false positives and recover small ships in noisy SAR imagery. Their method successfully improved detection accuracy, robustness, and feature extraction capability. Huang et al. [5] integrated convolutional features with Swin-based representations for fine-grained SAR ship recognition, reporting better efficiency and generalization. Xie et al. [6] augmented YOLOv4 with weighted multi-scale fusion and K-means-derived anchors, improving the localization and recall of small ships in complex nearshore scenes. Cao et al. [7] proposed a YOLOv8-based target detection algorithm for small inland-river vessels, which enhanced the feature extraction capability for small ship targets and improved both the accuracy and efficiency of ship target detection. Chen et al. [8] combined generative adversarial networks (GANs) with convolutional neural networks (CNNs), which effectively alleviated the detection difficulty caused by the limited number of small-ship samples. Zhou et al. [9] improved YOLOv5s by optimizing the loss function and enlarging the receptive field of the spatial pyramid pooling (SPP) layer, while Han et al. [10] proposed ShipYOLO to further improve the speed and accuracy of ship detection. In addition, Liu et al. [11] introduced a reverse depthwise separable convolution (RDSC) structure into YOLOv4 to improve detection efficiency. Jin et al. [12] incorporated the GhostBlock structure and a coordinated attention mechanism into YOLOv4, thereby reducing model complexity and improving the extraction of spatial location information. Zhao et al. [13] introduced a hybrid attention mechanism into the backbone network and proposed the YOLO-Marine model, which enhanced the feature extraction capability for ship targets. Guo et al. [14] combined the SimAM attention mechanism with YOLOv3 to strengthen ship target feature representation, thereby improving ship detection accuracy. Li et al. [15] proposed a lightweight ship target detection algorithm, named YOLOShip, based on an improved YOLOv5 framework. Luo et al. [16] proposed an adaptive feature fusion-based multi-scale ship detection method, which improved detection accuracy by suppressing background interference in complex scenarios. Recently, Cui et al. [17] proposed a parallel convolutional neural network–transformer model based on multimodal feature learning for underwater target recognition, which combines the strengths of convolutional neural networks (CNNs) and transformers, significantly improving the accuracy and stability of small target detection. Yu et al. [18] incorporated a multi-head self-attention mechanism into the YOLO framework, which significantly improved ship target detection performance in complex backgrounds.

Recent studies have further investigated ship detection under modality-specific and challenging maritime conditions. Lu et al. [19] developed multiscale feature-based and dual-channel feature-fusion methods for infrared ship detection, aiming to enhance weak target representation in low-contrast infrared imagery. Zhang et al. [20] proposed a lightweight YOLOv5-based detector for multi-scale ship targets, while Gao et al. [21] improved YOLOv8

for small-target detection in SAR images under specific marine environments. More recently, Zeng et al. [22] combined multi-scale feature extraction with attention fusion for ship detection in foggy scenes. These studies demonstrate the importance of multi-scale representation, attention enhancement, and feature fusion in complex maritime environments. Building on these findings, the present study jointly improves the backbone, attention mechanism, and feature-pyramid structure and evaluates the resulting detector on both visible-light and infrared ship datasets.

Existing maritime ship detectors generally improve one of three aspects: multi-scale feature extraction, attention-based feature enhancement, or feature-pyramid fusion. However, several limitations remain. First, conventional multi-scale modules may enhance receptive-field diversity but do not explicitly preserve weak boundary cues during repeated downsampling. Second, sequential channel-spatial attention, as used in conventional CBAM, may propagate an inaccurate channel response to the subsequent spatial attention stage, which is potentially unfavorable for weak and low-contrast ship targets. Third, conventional FPN and PAFPN structures mainly exchange information between adjacent feature levels, whereas small maritime targets require the preservation of higher-resolution spatial information. Based on these observations, this study investigates a unified detector in which the backbone, attention mechanism, and feature-fusion neck address different but complementary aspects of maritime ship representation.

Infrared ship detection is particularly challenging because ship targets often exhibit weak texture, blurred boundaries, large scale variations, and high inter-class visual similarity. The original YOLOv8 framework still has limitations in extracting fine-grained infrared ship features, selecting discriminative target regions under complex maritime backgrounds, and fusing multi-scale semantic information. The specific advances introduced in this study are outlined below:

(1) A re-parameterizable convolutional backbone, termed RepMCSwin, is developed for multi-scale maritime ship representation. The proposed MCSwin block replaces the window-based attention operations of the standard Swin Transformer with cascaded multi-scale convolutions and residual feature transformation, while the RepVGG-style branches can be structurally re-parameterized for inference.

(2) A Multi-Feature Parallel Convolutional Block Attention Module (MFPCBAM) is proposed for weak-ship-feature enhancement. Unlike conventional CBAM, which sequentially applies channel attention followed by spatial attention, MFPCBAM computes the two attention maps in parallel from the same input feature map. The channel branch employs ConvMLP-based channel interaction, while the spatial branch combines partial convolution with multi-scale atrous convolutions. The two branches jointly recalibrate the input features, thereby preserving weak ship responses and suppressing maritime background interference.

(3) An MGFPN neck is designed to strengthen cross-level feature interaction and introduce an additional  $160 \times 160$  prediction branch. In the complete architecture, MFPCBAM is inserted into selected MGFPN fusion nodes, while the two components remain independently defined.

(4) Extensive ablation and comparative experiments are conducted on both the public SeaShips dataset and a private infrared maritime ship dataset. The results demonstrate that MCSwin-YOLOv8 improves detection accuracy over the YOLOv8 baseline and shows competitive performance against several mainstream object detection models.

## 2. Materials and Methods

### 2.1. YOLOv8 Network Model

Released by Ultralytics, the YOLOv8 object detection model adopts a newly designed network architecture, as illustrated in Figure 1. In comparison with YOLOv6 and YOLOv7, YOLOv8 introduces a series of significant improvements by integrating multiple advanced techniques. The YOLOv8 family includes the n, s, m, l, and x configurations, offering progressively larger model capacities and computational requirements. This design allows users to flexibly choose an appropriate model version according to specific task requirements, thereby achieving optimal experimental results.

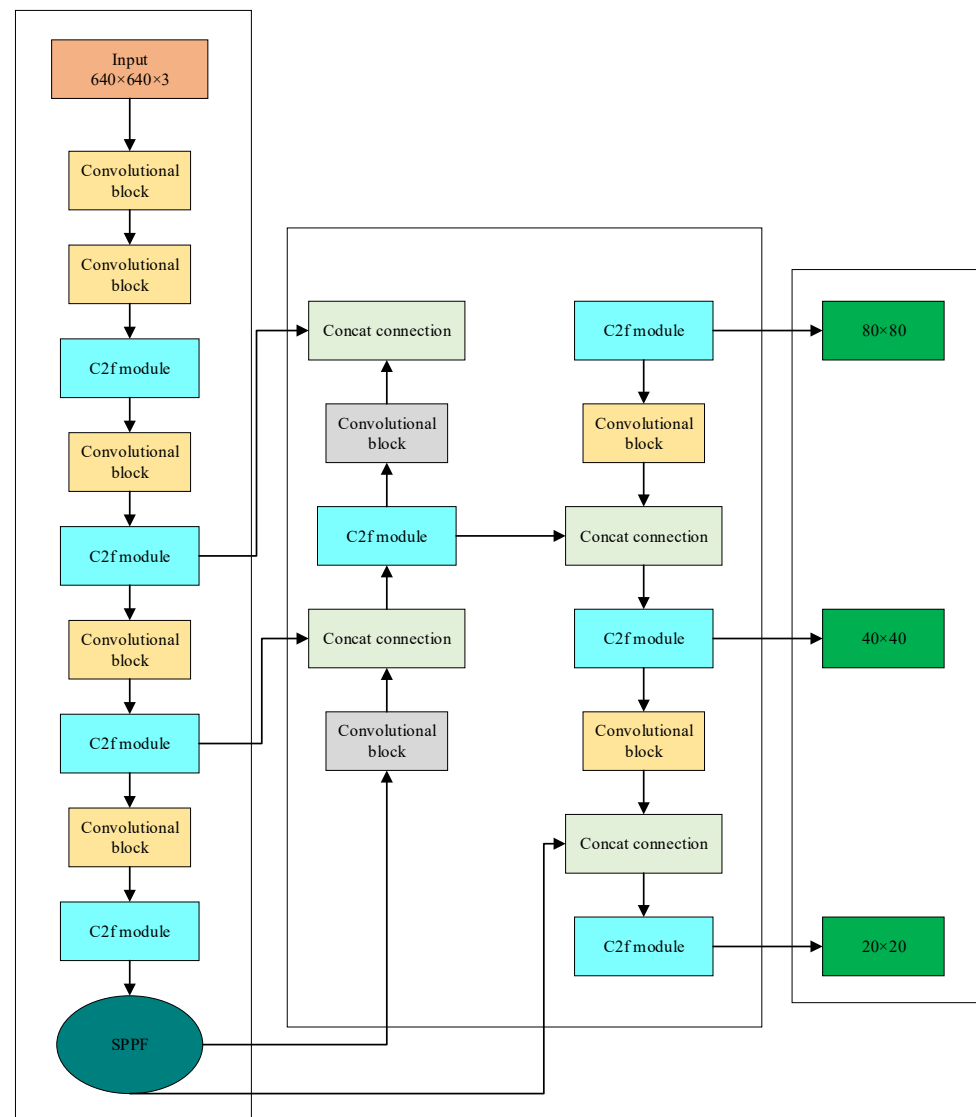


**Figure 1.** YOLOv8 model structure. Different colors indicate different network modules, and arrows indicate the direction of feature transmission.

### 2.2. Architecture of the MCSwin-YOLOv8 Detector

The proposed MCSwin-YOLOv8 follows the one-stage YOLOv8 detection framework and consists of a RepMCSwin backbone, an MGFPN neck, and a decoupled detection head, as shown in Figure 2. In the backbone, RepMCSwin replaces part of the original CBS and C2f modules to enhance multi-scale feature extraction and weak-boundary representation for infrared ship targets, while the SPPF module is retained for stable context aggregation. In the neck, MFPCBAM is embedded into the GFPN-based feature fusion structure to

enhance discriminative ship features and suppress background interference. To improve the detection of ships with large scale variations, the detection head adopts four prediction scales, i.e.,  $160 \times 160$ ,  $80 \times 80$ ,  $40 \times 40$ , and  $20 \times 20$ . Compared with the original YOLOv8 head, the additional high-resolution prediction branch provides finer spatial information and improves the sensitivity of the detector to small infrared ship targets.

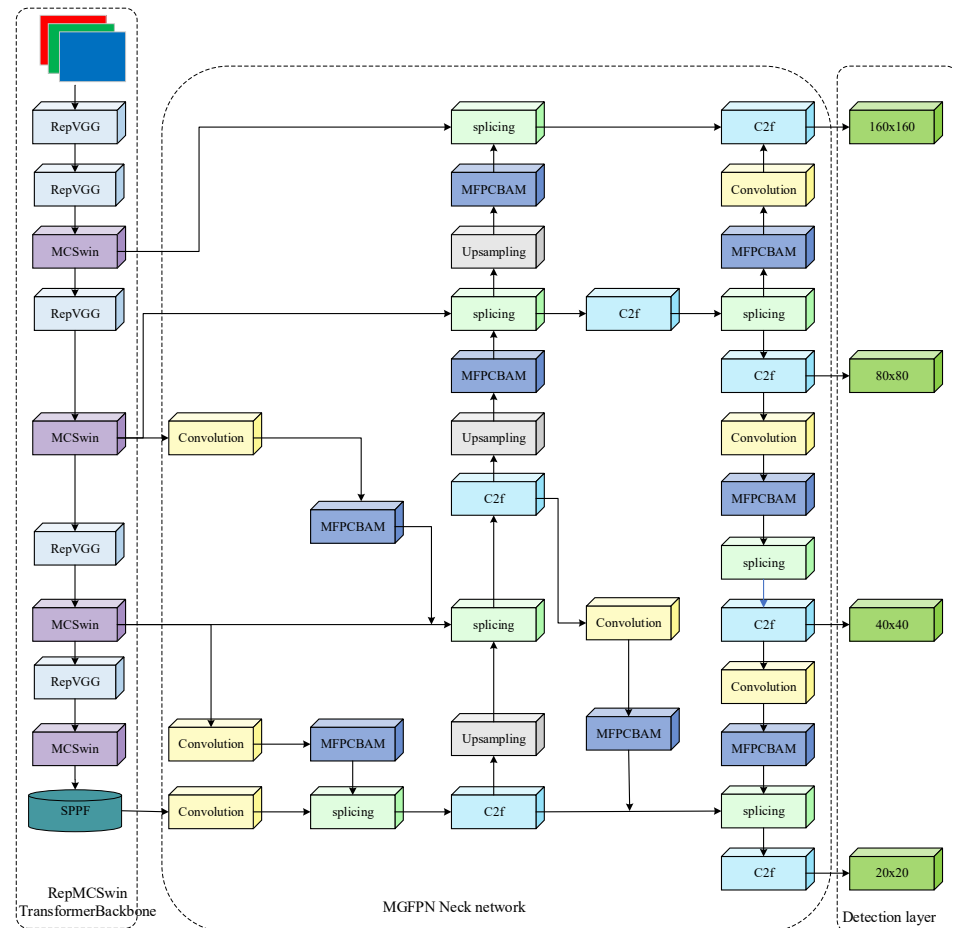


**Figure 2.** Structure of the MCSwin-YOLOv8 model. Different colors indicate different network modules, and arrows indicate the direction of feature transmission.

To avoid ambiguity, MGFPN and MFPCBAM are treated as two independently defined components throughout this manuscript. MGFPN refers to the modified GFPN topology with enhanced cross-level feature interaction and an additional  $160 \times 160$  high-resolution prediction branch. MFPCBAM denotes the proposed parallel channel–spatial attention module. In the complete MCSwin-YOLOv8 architecture, MFPCBAM is inserted into selected feature-fusion nodes of MGFPN.

### 2.3. RepMCSwin

The traditional Transformer model [23] captures global information and internal features by deeply encoding target images, thereby establishing contextual dependencies that enhance the model’s semantic understanding and improve detection accuracy. In particular, the Swin Transformer [24] adopts a shifted window attention mechanism, which enables effective modeling of features at different scales while simplifying the network structure and improving computational efficiency. Compared with the traditional Transformer, the Swin Transformer can model the spatial relationships of targets more accurately and preserve more spatial positional information, thereby demonstrating superior per-

formance. However, a standard Swin backbone does not explicitly accommodate the pronounced changes in ship scale, shape, weather, and maritime background, which can reduce detection stability in practical scenes.

Although the name “MCSwin” reflects the hierarchical multi-stage design motivation of Swin Transformer, the proposed block does not employ window-based multi-head self-attention or shifted-window attention. Instead, it adopts a convolution-oriented implementation consisting of cascaded multi-scale convolutions, feature concatenation, channel projection, normalization, and residual feature transformation. Therefore, RepMCSwin should be regarded as a re-parameterizable convolutional backbone rather than a Transformer backbone.

To address the above issues, this study proposes a RepMCSwin backbone structure, which combines the RepVGG module with the MCSwin module. Specifically, the RepVGG module is employed to enhance the representation capability of the model, while the MCSwin module is used to further extract multi-scale semantic features of ship targets. The architecture of the RepMCSwin backbone is illustrated in Figure 3.

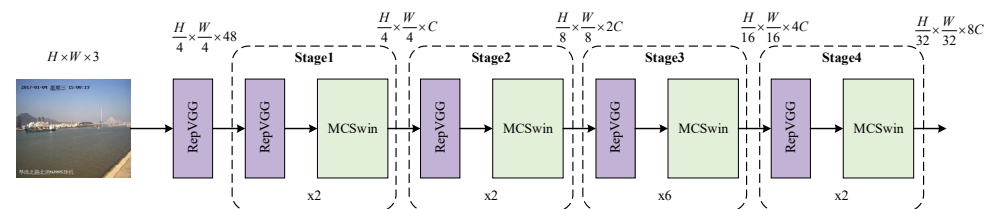


**Figure 3.** Structure of the RepMCSwin backbone. The non-English label in the figure denotes the original module or data-source label and is not an additional network component.

In the RepMCSwin backbone architecture, the RepVGG module performs dimensionality reduction on ship target features while enhancing the model’s representation capability, thereby ensuring scale consistency between adjacent stages. Through four stages, the network conducts deeper and more fine-grained feature extraction. Specifically, Stage 1 performs RepVGG-based feature processing to reduce the feature-map size and mine feature information more thoroughly. Stages 2 and 3 further capture and integrate features, providing richer feature representations. At Stage 4, RepVGG performs the final spatial downsampling before the resulting representation is refined by MCSwin. Figure 4 depicts the three RepVGG paths: a  $3 \times 3$  convolution, a  $1 \times 1$  convolution, and an identity or projection connection.

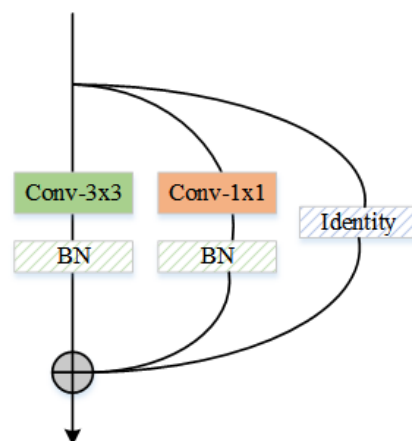


**Figure 4.** Structure of the RepVGG module.

In the RepVGG module, the stride of the  $3 \times 3$  convolution layer is set to 2, and down-sampling is adopted as the core operation, enabling the sampling of ship target features while reducing the height and width of the feature map. A stride-2  $1 \times 1$  convolution simultaneously aligns the channel dimension and supplies a second downsampling route alongside the  $3 \times 3$  branch. The features from the two downsampling branches are fused with the projection branch when downsampling is required.

Different from the original Swin Transformer, which relies on window-based self-attention, the proposed MCSwin block adopts a convolution-oriented multi-scale modeling strategy. This design is motivated by the characteristics of infrared ship images, where target boundaries are often weak and local edge cues are critical for classification and localization. Specifically, MCSwin employs cascaded multi-scale convolutional operations and an MLP-based feature transformation structure to extract local details and scale-aware semantic information. This makes the module more suitable for infrared ship detection than directly applying the standard Swin Transformer structure. First, multi-branch convolution operations are employed to capture the edge information of ship targets more thoroughly while preserving the key information of the original image, thereby providing high-quality input features. Second, a residual connection mechanism is introduced to enhance the stability and reliability of the MCSwin-YOLOv8 network, while also alleviating the gradient vanishing problem caused by increasing network depth. Finally, layer normalization is applied before the MCBot and MLP modules to stabilize optimization and maintain consistent feature statistics during training. The structure of the MCSwin module is shown in Figure 5.

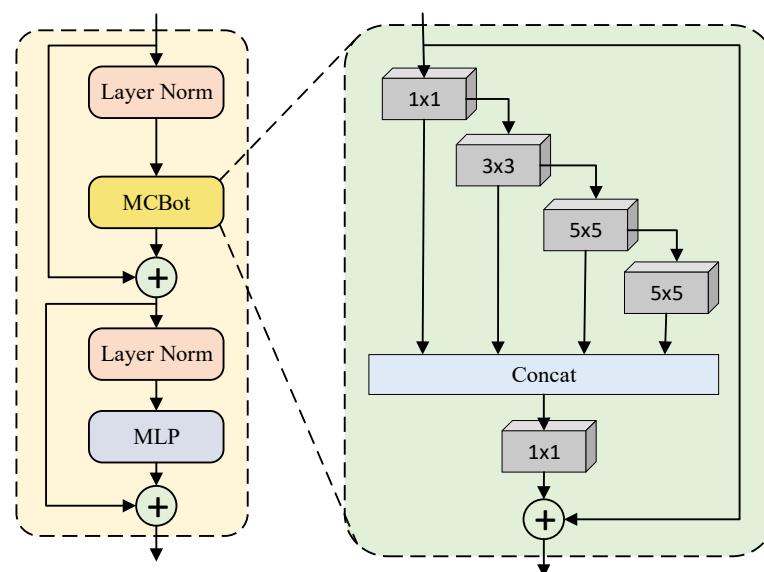


**Figure 5.** Structure of the MCSwin module. Different colors indicate different network modules, and arrows indicate the direction of feature transmission.

It should be noted that the proposed MCSwin block does not directly follow the W-MSA and SW-MSA operations used in the standard Swin Transformer. Instead, it introduces a convolution-oriented MCBot module to improve local edge modeling and scale-aware feature extraction for infrared ship targets. For an  $H \times W$  input, MCBot derives complementary representations through convolutional kernels with sizes of  $1 \times 1$ ,  $3 \times 3$ , and  $5 \times 5$ . The resulting multi-scale features are concatenated along the channel dimension and then compressed by a  $1 \times 1$  convolution to restore the channel dimension. The original representation is subsequently added through a shortcut connection, which retains input information and facilitates gradient flow.

The overall RepMCSwin backbone is organized into four stages. In each stage, the RepVGG module is used for feature downsampling and channel adjustment, while the MCSwin module further refines the extracted features. During training, the RepVGG block contains a  $3 \times 3$  convolution branch, a  $1 \times 1$  convolution branch, and an identity or projection branch followed by batch normalization. These branches enhance representation diversity during training and can be re-parameterized for efficient inference.

Before inference, the convolution and batch-normalization parameters of each branch are fused. The  $1 \times 1$  kernel and identity kernel are zero-padded to  $3 \times 3$ , and the equivalent kernels and biases of all branches are added to obtain a single  $3 \times 3$  convolution. This transformation preserves the mathematical equivalence of the block while removing the multi-branch topology during inference.

In the MCSwin module, the ship features are first normalized by an LN layer, after which the processed data are fed into the MCBot module. Within the MCBot module, feature extraction is sequentially performed using  $1 \times 1$ ,  $3 \times 3$ ,  $5 \times 5$ , and  $5 \times 5$  convolution kernels, resulting in a rich multi-scale feature representation. The extracted features are then further processed by a convolution, followed by a residual addition operation. Subsequently, the features output from the MCBot module are passed through another LN layer and an MLP layer, after which a second residual addition operation is performed to preserve valuable information from the original input. The detailed formulations are given in Equations (1) and (2).

$$\hat{Z}^k = \text{MCBot}\left(\text{LN}\left(z^{k-1}\right)\right) + z^{k-1} \quad (1)$$

$$\overset{k}{Z} = \text{MLP}\left(\text{LN}\left(\hat{Z}^k\right)\right) + \hat{Z}^k \quad (2)$$

where  $k = 1 \cdots L$ ,  $\hat{Z}^k$  denotes the vector obtained from the residual connection at the  $k$ -th layer of the MCBot module;  $\overset{k}{Z}$  denotes the vector obtained from the residual connection at the  $k$ -th layer of the MLP module; and  $\text{LN}$  denotes the normalization operation.

#### 2.4. MFPCBAM Attention Mechanism

Conventional CBAM applies channel attention followed by spatial attention in a sequential manner. Consequently, the input to the spatial-attention branch has already been recalibrated by channel attention, and inaccurate channel weighting may suppress useful spatial information for weak or low-contrast ship targets. In contrast, MFPCBAM generates the channel-attention and spatial-attention maps in parallel from the same input feature map. The channel branch uses global average pooling, global max pooling, and two  $1 \times 1$  convolutions in a ConvMLP structure. The spatial branch processes pooled spatial descriptors using partial convolution and parallel  $3 \times 3$  atrous convolutions with dilation rates of 1, 2, and 3. The two attention responses are then jointly applied to recalibrate the input features. Therefore, the novelty of MFPCBAM lies not in simply stacking two conventional attention modules, but in its parallel branch organization and enhanced multi-scale spatial-context modeling.

In the channel branch, global average pooling and global max pooling generate complementary channel descriptors. Instead of the fully connected multilayer perceptron used in conventional CBAM, MFPCBAM employs two  $1 \times 1$  convolutional layers for channel expansion and reduction. This ConvMLP implementation enhances channel interaction while maintaining a convolution-compatible representation.

In the spatial branch, channel-wise average and maximum pooling are first applied to obtain spatial descriptors. The concatenated descriptors are processed by a multi-branch

convolutional path consisting of partial convolution and  $3 \times 3$  atrous convolutions with dilation rates of 1, 2, and 3. The resulting features are concatenated and projected by a  $1 \times 1$  convolution to generate the spatial attention map.

To address these limitations, MFPCBAM computes channel attention and spatial attention in parallel. The channel pathway summarizes the input with average and maximum global pooling, producing two complementary channel descriptors. The descriptors are then processed by ConvMLP, where  $1 \times 1$  convolutions are used for channel expansion and reduction, preserving spatial consistency while enhancing channel-wise interactions. In the spatial branch, the pooled spatial descriptors are processed by a multi-scale convolutional path consisting of a  $1 \times 1$  convolution, a PConv branch, and three parallel  $3 \times 3$  atrous convolution branches with dilation rates of 1, 2, and 3. After concatenation, a  $1 \times 1$  projection compresses the branch outputs, and sigmoid normalization converts them into spatial weights.

$$F_{out} = F_C + F_S + F_{in} \quad (3)$$

Here, C denotes the number of feature channels, H and W denote the spatial dimensions,  $F_C$  and  $F_S$  denote the outputs of the channel-attention and spatial-attention branches, respectively, and  $F_{in}$  denotes the original input feature. The overall structure of the MFPCBAM is illustrated in Figure 6.

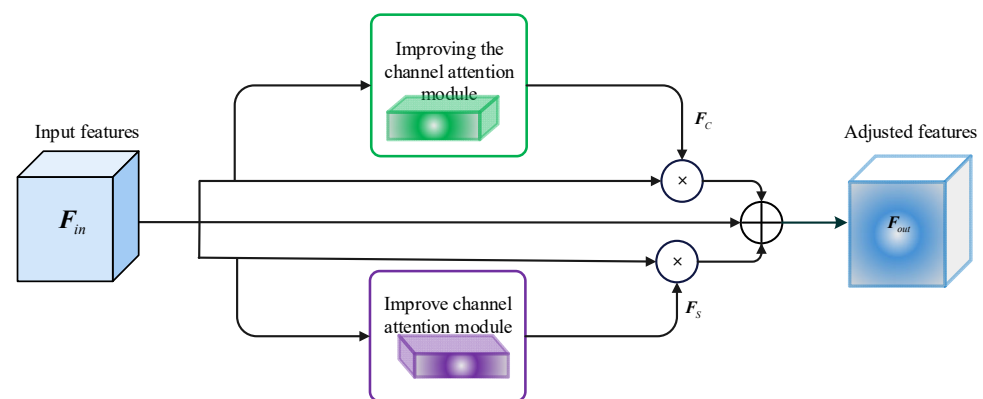


**Figure 6.** MFPCBAM attention module.

In the improved channel attention module of MFPCBAM, the input feature  $F_{in}$  is first processed by global average pooling and global max pooling; these two pooling operations complement each other. The resulting complementary features are then fed into the ConvMLP module, in which convolution operations are used to extract and transform the input features, thereby enhancing the feature representation capability. Unlike a conventional MLP, the ConvMLP module adopts a different processing strategy: in MFPCBAM, two  $1 \times 1$  convolution layers are employed to perform channel expansion and channel reduction. Subsequently, the outputs generated by the ConvMLP module are added and fused to obtain  $F_{add}$ . A sigmoid operation then maps  $F_{add}$  to the final set of channel-attention coefficients. The corresponding formulations are given in Equations (4)–(7). The structure of the improved channel attention module is shown in Figure 7.

$$F_{max} = \text{ConvMLP}(\text{MaxPool}(F_{in})) \quad (4)$$

$$F_{avg} = \text{ConvMLP}(\text{AvgPool}(F_{in})) \quad (5)$$

$$F_{add} = F_{avg} + F_{max} \quad (6)$$

$$F_C = \text{Sigmoid}(\text{Conv}_{1 \times 1}(F_{add}) \times F_{in}) \quad (7)$$

where denotes  $Conv_{1 \times 1}$  a convolution operation with a kernel size of  $1 \times 1$ ; Sigmoid denotes the nonlinear activation function, whose output ranges from 0 to 1.

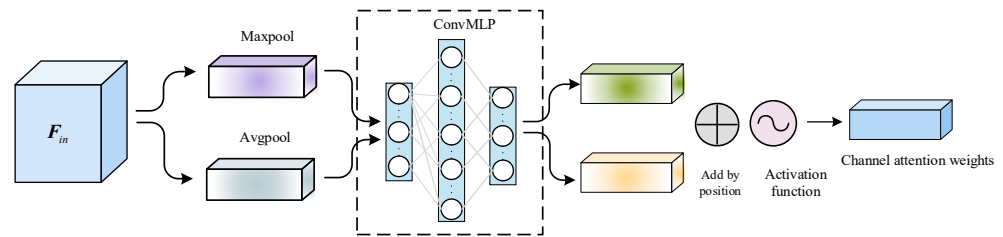


**Figure 7.** Architecture of the enhanced channel-attention branch.

In the spatial attention module, multi-scale atrous convolution [25] and PConv [26] are introduced. Multi-scale atrous convolution enlarges the receptive field of the convolution kernel through different dilation rates, thereby enhancing the ability to capture spatial contextual information and improving the model's capacity for understanding comprehensive images. In MFPCBAM, convolutions at different scales are combined in parallel with PConv to emphasize both the global structure and local details of the image, and the underlying architecture is illustrated in Figure 8. As shown in Figure 8, the  $1 \times 1$  convolution layer is used for linear transformation and channel adjustment; PConv is used to extract finer-grained features; and the atrous convolution layers are used to capture spatial context at different scales. This multi-scale design enables the model to attend to both global and local structures simultaneously, thereby improving its ability to capture and recognize critical features as well as its overall perception of image features.

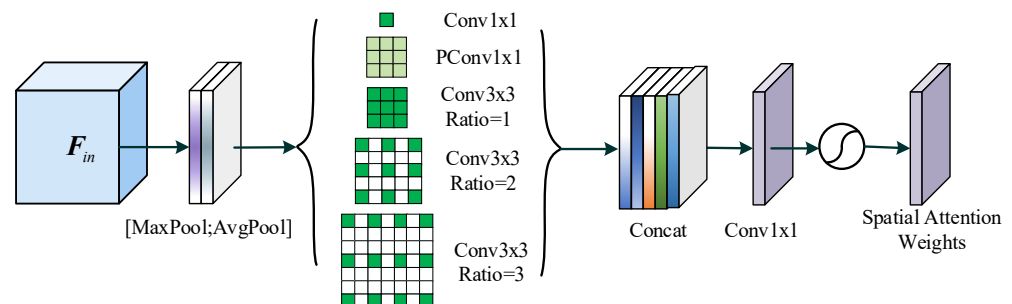


**Figure 8.** Spatial Attention Improvement module.

After the multi-scale convolution operations are completed, the output feature maps are concatenated and fused to obtain a more comprehensive and information-rich feature representation, denoted as  $A_{cat}$ . Subsequently, a  $1 \times 1$  convolution layer is applied for channel reduction, which decreases the computational cost and number of parameters while improving efficiency. Finally, the Sigmoid activation function is employed for normalization to generate the spatial attention weights, as described in Equations (8) and (9).

$$A_{cat} = \text{Concat}[Conv_{(1,1)}(.), Conv_{(3,1)}(.), Conv_{(3,2)}(.), Conv_{(3,3)}(.), PConv(.)] \quad (8)$$

$$F_S = \text{Sigmoid}(f^{1 \times 1}(A_{cat})) \quad (9)$$

where  $PConv$  denotes a convolution layer with a kernel size of  $1 \times 1$ , and  $f^{1 \times 1}$  denotes the convolution layer.

## 2.5. MGFPN

A conventional FPN propagates high-level semantics downward and merges them with backbone features at finer resolutions. However, the conventional FPN considers only unidirectional information flow, which leads to certain limitations. To address this issue, researchers proposed the Path Aggregation Feature Pyramid Network (PAFPN). While preserving the original FPN architecture, PAFPN introduces an additional bottom-up pathway, thereby enabling more comprehensive and in-depth feature fusion. Nevertheless, both the traditional FPN and the improved PAFPN mainly focus on feature fusion between adjacent levels while overlooking broader information interaction [27]. To overcome this limitation, the GFPN neck structure was proposed, which can efficiently exchange and fuse semantic information and spatial information, thus enhancing feature representation capability [28].

In ship detection, particularly in infrared maritime imagery, ship targets may be small, low-contrast, and characterized by weak or ambiguous boundaries. Although the original GFPN improves information exchange across multiple feature levels, its standard fusion topology does not explicitly prioritize high-resolution spatial cues or provide a dedicated stride-4 prediction path. Repeated feature fusion and downsampling may therefore weaken fine-grained boundary information and limit the representation of small or weakly expressed ship targets. To address these architectural limitations, MGFPN modifies the GFPN fusion topology to strengthen the interaction between shallow spatial features and deep semantic features and introduces an additional P2 prediction branch with a stride of 4. The modified fusion topology is designed to improve cross-level feature propagation, whereas the P2 branch preserves finer spatial details for small or weakly represented targets. The P2 branch is not specific to infrared imagery, but it is particularly relevant to the low-contrast and weak-boundary targets encountered in the infrared dataset. For an input resolution of  $640 \times 640$ , MGFPN outputs feature maps with resolutions of  $160 \times 160$ ,  $80 \times 80$ ,  $40 \times 40$ , and  $20 \times 20$ . The architecture of the attention-enhanced MGFPN neck is shown in Figure 9.

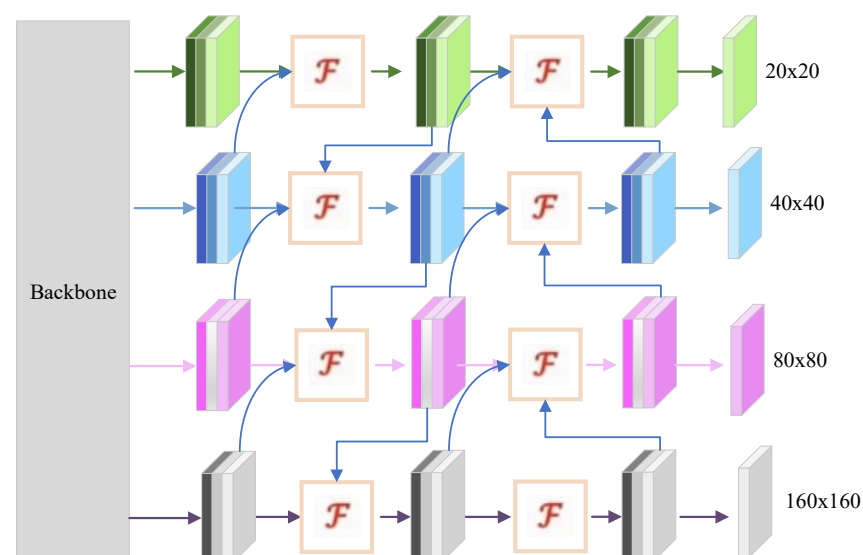


**Figure 9.** Architecture of the attention-enhanced MGFPN neck used in the complete MCSwin-YOLOv8 model. Different arrow styles indicate different feature-transmission paths.

The complete attention-enhanced neck extends the standalone MGFPN by inserting MFPCBAM into selected feature-fusion nodes to recalibrate multi-scale features. The additional  $160 \times 160$  prediction branch is part of the standalone MGFPN definition and is

therefore not attributed to MFPCBAM. During feature fusion, feature maps from different stages are first adjusted to a consistent scale by upsampling. This allows the neck to emphasize ship-related semantic responses and suppress irrelevant sea-surface background activations. The attention-enhanced features are then fused through  $1 \times 1$  convolutions for channel alignment. The fused features are subsequently downsampled and repeatedly integrated across scales until all feature levels are aggregated. This design strengthens both top-down semantic transmission and bottom-up localization information. The MGFPN structure not only enriches the diversity and expressive capacity of ship features but also enhances the feature extraction capability of the MCSwin-YOLOv8 network.

### 3. Results

To evaluate the effectiveness and generalization ability of the proposed MCSwin-YOLOv8, experiments were conducted on two ship detection datasets: the public SeaShips dataset and a private infrared maritime ship dataset. The SeaShips dataset was used to evaluate the model under visible-light maritime scenes, while the infrared dataset was used to verify its detection capability under weak texture, low-contrast, and complex maritime conditions. An identical training and evaluation protocol was applied to every model included in the comparison.

#### 3.1. Datasets and Experimental Settings

This section introduces the two datasets used in the experiments and describes the implementation details. The object distributions of the two datasets are summarized in Table 1.

**Table 1.** (a) SeaShips dataset; (b) infrared maritime ship dataset.

| (a)   |                    |          |            |      |        |
|-------|--------------------|----------|------------|------|--------|
|       | Category           | Training | Validation | Test | Total  |
| 1     | ore carrier        | 542      | 586        | 1071 | 2199   |
| 2     | container ship     | 245      | 219        | 437  | 901    |
| 3     | bulk cargo carrier | 489      | 468        | 995  | 1952   |
| 4     | general cargo ship | 393      | 368        | 744  | 1505   |
| 5     | fishing boat       | 510      | 492        | 1188 | 2190   |
| 6     | passenger ship     | 100      | 126        | 248  | 474    |
| Total |                    | 2279     | 2259       | 4683 | 9221   |
| (b)   |                    |          |            |      |        |
|       | Category           | Training | Validation | Test | Total  |
| 1     | liner              | 1020     | 101        | 291  | 1412   |
| 2     | bulk carrier       | 4103     | 416        | 1258 | 5777   |
| 3     | warship            | 1831     | 222        | 494  | 2547   |
| 4     | sailboat           | 3482     | 347        | 1106 | 4935   |
| 5     | canoe              | 1090     | 107        | 367  | 1564   |
| 6     | container ship     | 473      | 64         | 124  | 661    |
| 7     | fishing boat       | 6642     | 704        | 1734 | 9080   |
| Total |                    | 18,641   | 1961       | 5374 | 25,976 |

##### 3.1.1. SeaShips Dataset

SeaShips [29] is a public visible-light ship detection dataset constructed from maritime video frames. It contains six ship categories, including ore carrier, container ship, bulk cargo carrier, general cargo ship, fishing boat, and passenger ship. The publicly available subset used in this study contains 9221 annotated ship targets. Following the official dataset partition, the predefined training, validation, and test subsets were directly adopted in this

study without redistributing the samples. The official protocol recommends dividing the dataset into training, validation, and test sets at an approximate ratio of 1:1:2. To ensure consistency with the officially recommended evaluation protocol, the same split ratio was adopted in this study. The original annotations were converted from PASCAL VOC format to YOLO format for model training and evaluation.

### 3.1.2. Infrared Maritime Ship Dataset

The infrared maritime ship dataset was jointly provided by China State Shipbuilding Corporation No. 717 Research Institute and Guide Infrared Technology Co., Ltd. It contains 8402 infrared images with a resolution of  $384 \times 288$  pixels and includes seven ship categories: liner, bulk carrier, warship, sailboat, canoe, container ship, and fishing boat. The dataset contains 25,976 annotated ship targets and was divided into training, validation, and test sets with a ratio of 7:1:2. During the partitioning process, the category distributions were kept as consistent as possible across the three subsets to reduce potential sampling bias. This partition provides sufficient samples for model training while retaining independent validation and test sets for evaluating the generalization and robustness of the model under practical infrared maritime conditions. The original PASCAL VOC annotations were also converted into YOLO format for model training.

### 3.1.3. Implementation Details

All experiments were conducted using Ubuntu 18.04, Python 3.8, and PyTorch 2.10. The experiments were conducted on a hardware setup featuring an NVIDIA GeForce RTX 4090 GPU with 24 GB of memory (NVIDIA Corporation, Santa Clara, CA, USA). The input images were resized to  $640 \times 640$ . Each model was trained for 300 epochs using SGD, with a batch size of 8, an initial learning rate of 0.01, and a momentum of 0.937. The training parameters used in all experiments are summarized in Table 2.

**Table 2.** Network training parameters.

| Item       | Specification         | Parameter             | Value |
|------------|-----------------------|-----------------------|-------|
| GPU        | GeForce RTX 4090 24 G | Initial learning rate | 0.01  |
| CPU        | Intel Core i9-13900K  | Batch size            | 8     |
| CUDA       | 11.3                  | Optimizer             | SGD   |
| cuDNN      | 8.0                   | Momentum              | 0.937 |
| Input size | $640 \times 640$      | Epochs                | 300   |

### 3.2. Evaluation Metrics

Precision, Recall, F1-score, AP, and mAP were used to evaluate detection performance. Precision measures the proportion of correctly detected positive samples among all predicted positive samples, while Recall measures the proportion of correctly detected positive samples among all ground-truth positive samples. They are defined as

$$P = \frac{TP}{TP + FP} \quad (10)$$

$$R = \frac{TP}{TP + FN} \quad (11)$$

$$F1 = 2 \times \frac{P \times R}{P + R} \quad (12)$$

$$mAP = \frac{1}{m} \sum_{i=1}^m AP_i \quad (13)$$

where TP, FP, and FN denote true positives, false positives, and false negatives, respectively. For each class, AP corresponds to the integral of its precision–recall curve; averaging these class-level values gives mAP. We report AP averaged at IoU = 0.5 and across IoU thresholds of 0.50–0.95, with the latter evaluated at increments of 0.05. During the evaluation process, the default confidence threshold of 0.25 and the IoU threshold of 0.70 provided by the YOLO framework were adopted to calculate Precision, Recall, and the F1-score, ensuring consistency and reproducibility with standard object detection evaluation protocols.

### 3.3. Ablation Experiments

The revised ablation study evaluates standalone MGFPN, MFPCBAM, and RepMC-Swin configurations, as well as the combined MGFPN + MFPCBAM configuration and the complete model. This design separates the marginal contribution of each component from the performance of their combined architecture. The experimental results are shown in Table 3. Since the model architecture is the same on both datasets, the numbers of parameters and FLOPs are reported together for a fair comparison of accuracy and computational cost.

**Table 3.** (a) Ablation results on the SeaShips dataset. (b) Ablation results on the infrared maritime ship dataset.

| (a) |       |         |           |       |       |         |              |         |        |
|-----|-------|---------|-----------|-------|-------|---------|--------------|---------|--------|
| No. | MGFPN | MFPCBAM | RepMCSwin | P     | R     | mAP@0.5 | mAP@0.5:0.95 | Params  | FLOPs  |
| 1   | ×     | ×       | ×         | 94.7% | 91.2% | 96.0%   | 68.7%        | 11.12 M | 28.5 G |
| 2   | ✓     | ×       | ×         | 94.1% | 92.8% | 96.8%   | 71.1%        | 12.00 M | 38.3 G |
| 3   | ×     | ✓       | ×         | 94.9% | 92.3% | 96.5%   | 73.4%        | 14.57 M | 45.1 G |
| 4   | ×     | ×       | ✓         | 95.5% | 91.8% | 96.2%   | 74.3%        | 14.59 M | 33.7 G |
| 5   | ✓     | ✓       | ×         | 94.0% | 92.7% | 96.9%   | 71.2%        | 15.85 M | 51.4 G |
| 6   | ✓     | ×       | ✓         | 94.9% | 92.9% | 97.0%   | 74.6%        | 15.47 M | 43.5 G |
| 7   | ×     | ✓       | ✓         | 95.6% | 92.6% | 96.8%   | 74.8%        | 16.67 M | 53.7 G |
| 8   | ✓     | ✓       | ✓         | 95.7% | 93.2% | 97.4%   | 75.1%        | 19.29 M | 56.7 G |
| (b) |       |         |           |       |       |         |              |         |        |
| No. | MGFPN | MFPCBAM | RepMCSwin | P     | R     | mAP@0.5 | mAP@0.5:0.95 | Params  | FLOPs  |
| 1   | ×     | ×       | ×         | 92.0% | 85.4% | 91.1%   | 63.1%        | 11.12 M | 28.5 G |
| 2   | ✓     | ×       | ×         | 89.6% | 86.4% | 92.0%   | 61.3%        | 12.00 M | 38.3 G |
| 3   | ×     | ✓       | ×         | 91.9% | 87.1% | 92.7%   | 62.6%        | 15.57 M | 45.1 G |
| 4   | ×     | ×       | ✓         | 91.7% | 87.0% | 92.1%   | 65.1%        | 14.59 M | 33.7 G |
| 5   | ✓     | ✓       | ×         | 90.3% | 87.7% | 92.9%   | 63.4%        | 15.85 M | 51.4 G |
| 6   | ✓     | ×       | ✓         | 90.8% | 88.3% | 93.1%   | 65.5%        | 15.47 M | 43.5 G |
| 7   | ×     | ✓       | ✓         | 91.6% | 88.8% | 93.5%   | 65.8%        | 16.67 M | 53.7 G |
| 8   | ✓     | ✓       | ✓         | 91.4% | 91.0% | 94.1%   | 66.9%        | 19.29 M | 56.7 G |

Note: ✓ indicates that the corresponding module is included, whereas × indicates that it is not included.

The category-wise AP results further demonstrate the effectiveness of the proposed modules. On the SeaShips dataset, MCSwin-YOLOv8 achieves the highest AP in most categories, especially for passenger ship and fishing boat, indicating that the proposed feature fusion and attention mechanisms improve the recognition of visually similar ship categories. On the infrared maritime ship dataset, the AP of fishing boat increases from 81.1% to 91.5%, and the AP of sailboat increases from 79.4% to 86.5%. These improvements show that the proposed model is particularly effective for categories with weak boundaries, large scale variations, and high background interference.

As shown in Figure 10, MCSwin-YOLOv8 exhibits a more concentrated diagonal distribution than the original YOLOv8 model, indicating improved class-wise recognition performance. The improvements are particularly evident for sailboats and fishing boats,

which are more susceptible to weak boundaries, visual similarity, and maritime background interference. Meanwhile, the reduced off-diagonal responses between these categories and the background indicate that the proposed RepMCSwin backbone and attention-enhanced MGFPN neck improve the discrimination of weak ship contours from background responses. These observations are consistent with the category-wise AP improvements reported in Table 3.

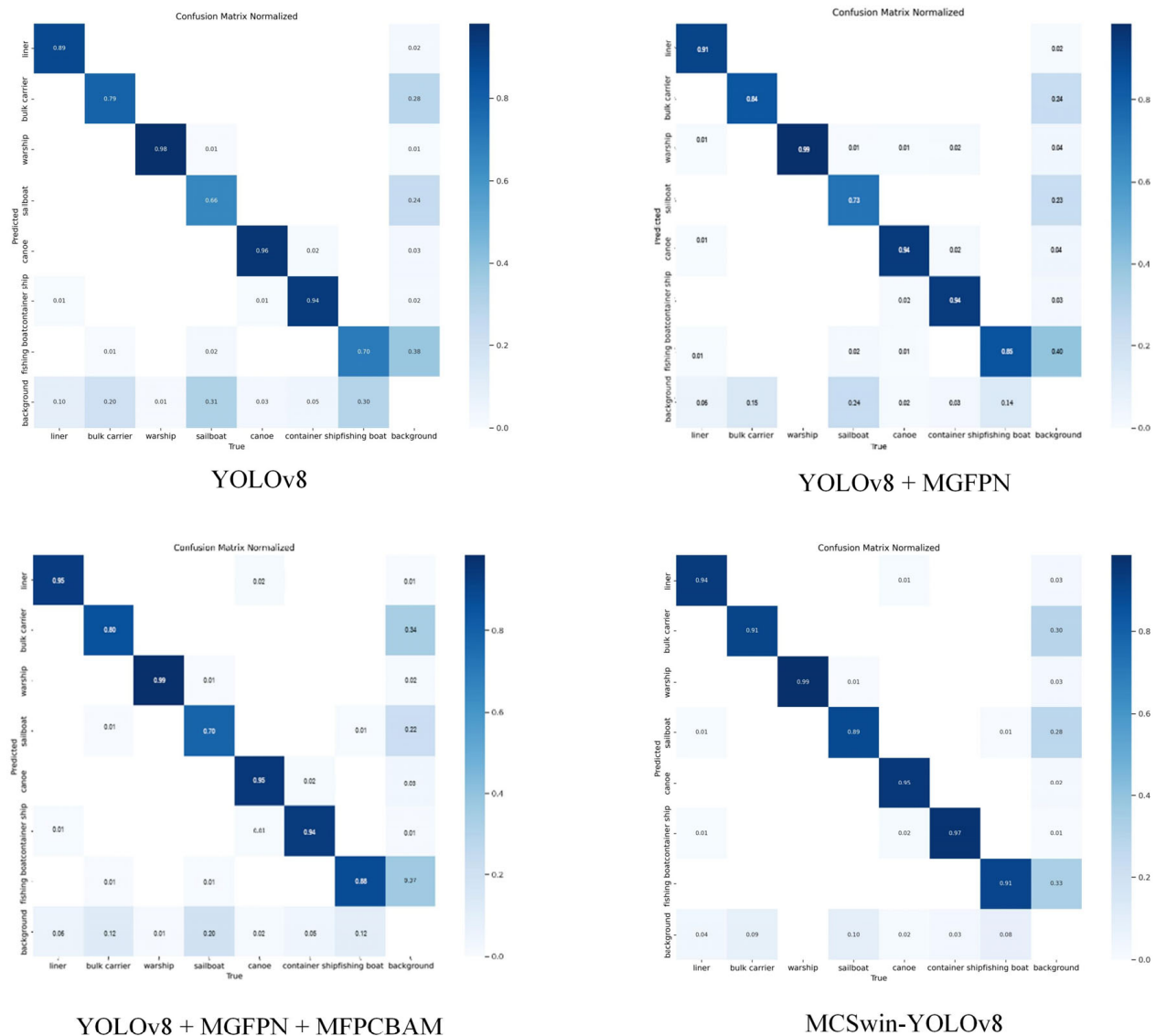


**Figure 10.** Confusion matrix of infrared offshore ship dataset ablation experiment.

### 3.4. Comparison with Mainstream Detectors

To further evaluate the competitiveness of MCSwin-YOLOv8, comparative experiments were conducted against several mainstream object detectors, including YOLOv5, YOLOv7, YOLOv8, YOLOv9, YOLOv10, RT-DETR, Faster R-CNN, Cascade R-CNN, RetinaNet, DCNv2, and SSD. The comparison results on the SeaShips dataset and the infrared maritime ship dataset are reported in Tables 4 and 5, respectively. All compared models were trained using the same dataset splits, input resolution, training epochs, optimizer, learning rate, batch size, data-augmentation strategy, and evaluation pipeline.

**Table 4.** Comparison with mainstream detectors on the SeaShips dataset.

| Model         | F1    | mAP@0.5 | mAP@0.5:0.95 | Params  | FLOPs    |
|---------------|-------|---------|--------------|---------|----------|
| YOLOv5s       | 91.1% | 94.6%   | 69.2%        | 7.03 M  | 16.0 G   |
| YOLOv7        | 93.3% | 96.1%   | 68.2%        | 37.21 M | 105.2 G  |
| YOLOv8s       | 92.9% | 96.0%   | 68.7%        | 11.12 M | 28.5 G   |
| YOLOv9s       | 94.1% | 96.7%   | 74.6%        | 9.74 M  | 39.6 G   |
| YOLOv10s      | 92.5% | 96.3%   | 73.2%        | 8.07 M  | 24.8 G   |
| RT-DETR       | 92.1% | 94.2%   | 66.8%        | 32.81 M | 108.0 G  |
| Faster RCNN   | 83.2% | 95.1%   | 65.5%        | 41.15 M | 91.03 G  |
| Cascade RCNN  | 87.2% | 95.8%   | 72.1%        | 68.94 M | 118.8 G  |
| RetinaNet     | 73.0% | 87.6%   | 61.2%        | 36.21 M | 82.57 G  |
| DCNv2         | 84.9% | 95.5%   | 67.3%        | 148.7 M | 104.2 G  |
| SSD           | 79.0% | 89.2%   | 63.2%        | 24.41 M | 138.19 G |
| MCSwin-YOLOv8 | 94.4% | 97.4%   | 75.1%        | 19.29 M | 56.7 G   |

**Table 5.** Comparison with mainstream detectors on the infrared maritime ship dataset.

| Model         | F1    | mAP@0.5 | mAP@0.5:0.95 | Params  | FLOPs   |
|---------------|-------|---------|--------------|---------|---------|
| YOLOv5s       | 85.6% | 88.9%   | 55.8%        | 7.03 M  | 16.0 G  |
| YOLOv7        | 89.5% | 93.1%   | 63.5%        | 37.21 M | 105.2 G |
| YOLOv8s       | 88.6% | 91.1%   | 63.1%        | 11.12 M | 28.5 G  |
| YOLOv9s       | 88.5% | 91.0%   | 64.8%        | 9.74 M  | 39.6 G  |
| YOLOv10s      | 88.7% | 92.1%   | 64.4%        | 8.07 M  | 24.8 G  |
| RT-DETR       | 82.9% | 87.3%   | 59.3%        | 32.81 M | 108.0 G |
| Faster RCNN   | 85.4% | 89.1%   | 57.3%        | 41.15 M | 91.03 G |
| Cascade RCNN  | 85.4% | 89.9%   | 60.1%        | 68.94 M | 118.8 G |
| RetinaNet     | 69.8% | 82.2%   | 49.8%        | 36.21 M | 82.57 G |
| DCNv2         | 83.2% | 89.7%   | 57.6%        | 148.7 M | 104.2 G |
| SSD           | 74.9% | 84.7%   | 52.7%        | 24.41 M | 138.2 G |
| MCSwin-YOLOv8 | 91.2% | 94.1%   | 66.9%        | 19.29 M | 56.7 G  |

As shown in Table 4, MCSwin-YOLOv8 achieves the best overall performance on the SeaShips dataset, with an F1-score of 94.4%, mAP@0.5 of 97.4%, and mAP@0.5:0.95 of 75.1%. Relative to YOLOv8, MCSwin-YOLOv8 gains 1.4 percentage points in mAP@0.5 and 6.4 percentage points in mAP@0.5:0.95. Although the model introduces additional parameters and FLOPs, it achieves higher detection accuracy than both one-stage and two-stage detectors.

As shown in Table 5, MCSwin-YOLOv8 also achieves the best detection accuracy on the infrared maritime ship dataset. It obtains an F1-score of 91.2%, mAP@0.5 of 94.1%, and mAP@0.5:0.95 of 66.9%, outperforming the YOLOv8 baseline by 3.0% and 3.8% in mAP@0.5 and mAP@0.5:0.95, respectively. These results indicate that the proposed model can effectively handle challenging infrared maritime scenarios with weak texture, low contrast, and large scale variations.

### 3.5. Inference Speed and Runtime Efficiency Analysis

To evaluate runtime efficiency, the inference speed of each compared detector was measured on the same hardware platform, an NVIDIA GeForce RTX 4090 GPU with 24 GB of memory (NVIDIA Corporation, Santa Clara, CA, USA), using the same input resolution (640 × 640) and batch size (1) and is reported in frames per second (FPS). Figure 11 presents the precision–speed and recall–speed trade-offs on the infrared maritime ship dataset.

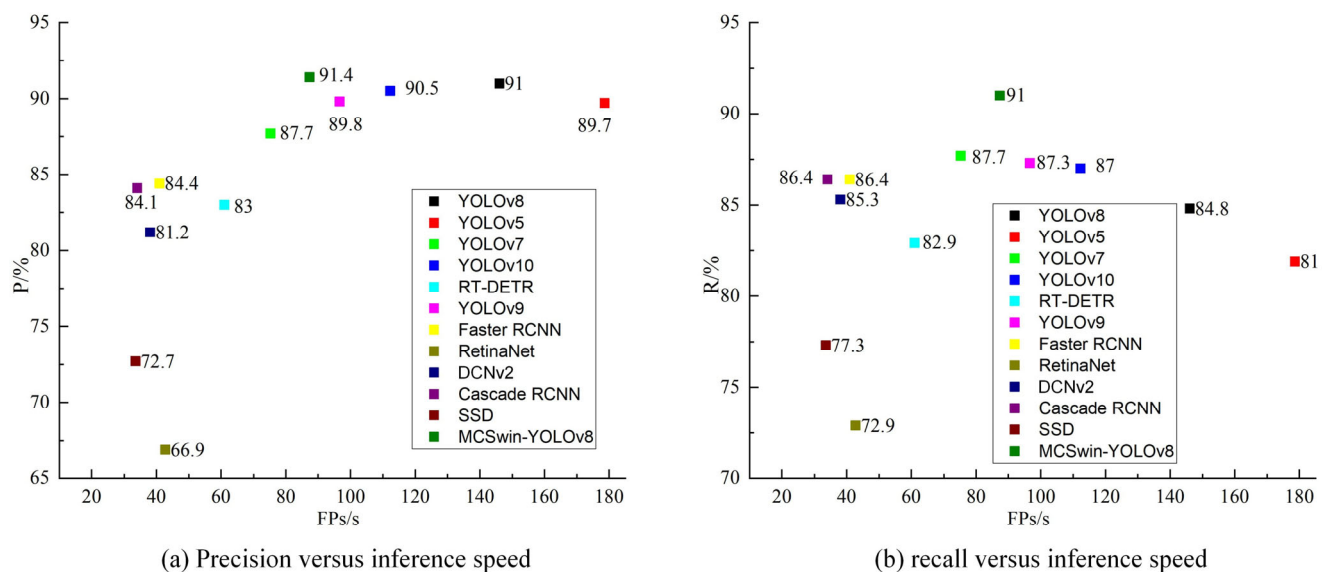


**Figure 11.** Comparison of (a) precision and (b) recall against inference speed (FPS) for different detectors on the infrared maritime ship dataset. Speeds were measured on an RTX 4090 with a  $640 \times 640$  input and batch size 1.

As shown in Figure 11, MCSwin-YOLOv8 achieves the highest precision (91.4%) among all compared models while maintaining an inference speed of 87.3 FPS. Although its speed is slightly lower than those of the lighter YOLOv5, YOLOv8, YOLOv9, and YOLOv10 models, it consistently delivers the best precision; compared with YOLOv5, for example, precision is improved by 1.7 percentage points. Against the remaining main-stream detectors (RT-DETR, Faster R-CNN, DCNv2, RetinaNet, Cascade R-CNN, and SSD), MCSwin-YOLOv8 is superior in both metrics: it runs approximately  $2.6\times$  faster than SSD, and its precision is 24.5 percentage points higher than that of RetinaNet. Figure 11 shows a similar pattern for recall: MCSwin-YOLOv8 achieves the highest recall (91.0%) at 87.3 FPS. These results indicate that the accuracy improvements brought by RepMCSwin, MFPCBAM, and MGFPN do not come at a prohibitive runtime cost, and that the re-parameterized design keeps the model within a practical speed range for maritime surveillance.

### 3.6. Qualitative Visualization

To further examine the detection behavior of the proposed model under different imaging conditions and target distributions, three representative cases are selected for visual comparison: a visible-light multi-target scene and a visible-light single-target scene from the SeaShips test set, together with an infrared scene containing a low-contrast target with a weak boundary. All comparison models are evaluated on the same input images. The analysis focuses on missed detections, duplicate detections, category predictions, bounding-box localization, and confidence scores.

Figure 12 presents a visible-light multi-target scene containing three ore carriers. MCSwin-YOLOv8 correctly detects all three ships with confidence scores of 0.93, 0.92, and 0.82, whereas YOLOv5 and YOLOv10 miss one target and Cascade R-CNN and DCNv2 produce additional detections. This example shows that the complete model can maintain high target coverage and a low false-positive rate when multiple ships with different scales appear in the same image. From the perspective of module functionality, MGFPN exchanges information across feature levels and transfers high-level semantic cues to high-resolution features, thereby preserving responses from ships at different scales and reducing missed detections of small targets. RepMCSwin enhances local structures and ship boundaries through convolution-oriented multi-scale modeling, which helps distinguish

neighboring ships and maintain complete bounding boxes. MFPCBAM applies parallel channel and spatial attention to suppress non-target responses from the sea background, thereby reducing the likelihood of additional or duplicate detections.



**Figure 12.** Visual comparison of different detectors in a multi-target scene from the SeaShips test set. The non-English text is an embedded timestamp/source watermark from the original dataset image; it is not an annotation or model prediction.

Figure 13 shows a visible-light image containing a single general cargo ship. MCSwin-YOLOv8 correctly detects and localizes the target without producing additional bounding boxes, assigning a confidence score of 0.94. YOLOv5, YOLOv10, and RT-DETR also detect the target, with confidence scores of 0.86, 0.93, and 0.89, respectively, whereas several two-stage and conventional detectors produce duplicate detections. Because the target in this scene has a moderate scale, the small-target recovery effect of MGFPN is less prominent than in Figure 12. Nevertheless, its cross-level fusion helps maintain consistency between semantic and spatial information. MFPCBAM highlights ship-related regions from both the channel and spatial dimensions while weakening interference from sea texture and background structures. This behavior is consistent with the absence of duplicate responses in the complete model. At the same time, the local contour and edge enhancement of RepMCSwin supports the generation of a compact and stable bounding box.



**Figure 13.** Visual comparison of different detectors in a single-target scene from the SeaShips test set. The non-English text is an embedded timestamp/source watermark from the original dataset image; it is not an annotation or model prediction.

As shown in Figure 14, the canoe exhibits low contrast and a weak boundary relative to the surrounding infrared maritime background. MCSwin-YOLOv8 correctly localizes the canoe with a confidence score of 0.96 and does not generate duplicate detections. YOLOv5 and YOLOv8 also localize the target but assign lower confidence scores of 0.64 and 0.74, respectively. YOLOv7 and Faster R-CNN each assign a confidence score of 0.90, while YOLOv9, YOLOv10, and RT-DETR each assign a confidence score of 0.95. Although Cascade R-CNN assigns a confidence score of 0.98 to the correct target, it produces an additional false-positive bounding box with a confidence score of 0.52. RetinaNet generates multiple overlapping detections and incorrectly classifies part of the target as a container ship. Therefore, a high confidence score for one correctly detected bounding box does not necessarily indicate reliable overall detection; missed detections, false positives, duplicate predictions, category correctness, and localization quality should also be considered.

The three examples broaden the qualitative comparison across visible-light and infrared imaging conditions and across multi-target, single-target, and weak-boundary scenes. Nevertheless, these visual results are illustrative and should not be interpreted as a comprehensive evaluation of model accuracy. In particular, the confidence score shown for an individual bounding box is not equivalent to dataset-level detection accuracy. The qualitative observations also cannot isolate the contribution of RepMCSwin, MFPCBAM, or MGFPN, because all three components are simultaneously present in the complete model.

Conclusions regarding overall performance and component effectiveness should therefore be based primarily on the quantitative comparison and ablation results.

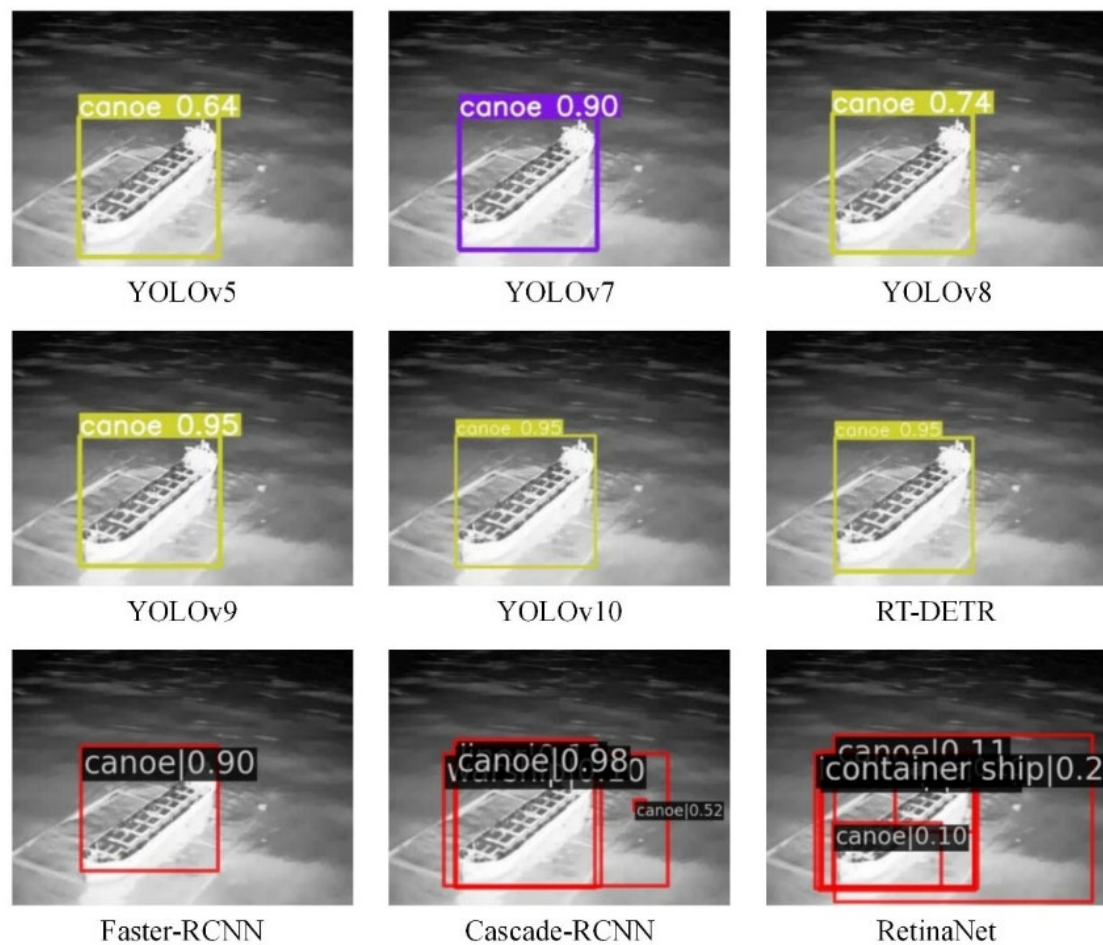


**Figure 14.** Visual comparison of different detectors for a weak-boundary canoe target in the infrared maritime ship test set.

## 4. Discussion

### 4.1. Effectiveness of the Proposed Components

The qualitative results provide a diagnostic analysis of different error sources in the detection pipeline. RepMCSwin is introduced in the backbone to construct scale-aware and boundary-sensitive feature representations. MFPCBAM applies parallel channel and spatial attention to select discriminative responses and suppress irrelevant maritime background activations. MGFPN operates in the neck and promotes information exchange among feature levels, allowing high-level semantic information and low-level spatial details to be used jointly.

On the SeaShips dataset, the three standalone configurations exhibit distinct performance signatures. MGFPN increases recall from 91.2% to 92.8% and mAP@0.5 from 96.0% to 96.8%, indicating that cross-scale feature exchange helps recover ship targets that may be missed by the original feature pyramid. However, precision decreases to 94.1%, suggesting that broader multi-scale aggregation may also introduce additional background responses. MFPCBAM contributes more strongly to feature discrimination and localization quality. Its mAP@0.5:0.95 reaches 73.4%, representing a gain of 4.7 percentage points over the baseline, while both precision and recall also improve. RepMCSwin produces the highest strict-IoU performance among the single-module configurations, increasing mAP@0.5:0.95

to 74.3%, or 5.6 percentage points above the baseline, and achieving the highest precision of 95.5%. These distinct metric patterns indicate that MGFPN mainly supports target recovery, MFPCBAM performs feature selection and background suppression, whereas RepMCSwin more directly improves boundary localization under scale variation.

The infrared dataset further demonstrates why these functions are necessary. Because infrared images contain low contrast, weak texture, and blurred target boundaries, MGFPN alone increases recall by 1.0 percentage point and  $\text{mAP}@0.5$  by 0.9 percentage points but decreases precision by 2.4 percentage points and  $\text{mAP}@0.5:0.95$  by 1.8 percentage points. This result indicates that simply expanding cross-scale information flow can propagate both target evidence and background noise in weak-target scenes. MFPCBAM improves  $\text{mAP}@0.5$  by 1.6 percentage points and recall by 1.7 percentage points, but its strict- $\text{IoU}$  performance remains slightly below the baseline. Thus, attention improves target response and category discrimination but cannot independently resolve unstable boundary localization. RepMCSwin, in contrast, increases  $\text{mAP}@0.5:0.95$  by 2.0 percentage points, confirming the value of multi-scale convolution and local edge modeling for weak-boundary infrared targets.

The two-component configurations provide direct evidence of complementarity. On SeaShips, combining MGFPN with RepMCSwin increases recall to 92.9%, which is 1.1 percentage points higher than that of RepMCSwin alone, while  $\text{mAP}@0.5$  reaches 97.0%. This indicates that the cross-scale recovery capability of MGFPN compensates for the stronger emphasis of RepMCSwin on local representation and precise localization. Conversely, adding RepMCSwin to MGFPN increases  $\text{mAP}@0.5:0.95$  from 71.1% to 74.6%, showing that boundary-sensitive backbone features can reduce the localization uncertainty introduced by multi-scale fusion. The MFPCBAM + RepMCSwin configuration achieves 95.6% precision and 92.6% recall, with  $\text{mAP}@0.5:0.95$  reaching 74.8%. This result demonstrates a strong complementarity between attention-based feature selection and boundary modeling. By comparison, the MGFPN + MFPCBAM configuration provides only limited additional improvement, suggesting that feature fusion and attention alone are insufficient to fully improve localization without the fine-grained local representation supplied by RepMCSwin.

The infrared results reveal an even clearer interaction among the modules. When MGFPN is combined with RepMCSwin,  $\text{mAP}@0.5:0.95$  increases from 61.3% with MGFPN alone to 65.5%, while recall reaches 88.3%. This indicates that RepMCSwin constrains and refines the multi-scale information aggregated by MGFPN through boundary-sensitive local features. The MFPCBAM + RepMCSwin configuration reaches 93.5%  $\text{mAP}@0.5$  and 65.8%  $\text{mAP}@0.5:0.95$ , representing gains of 0.8 and 3.2 percentage points over MFPCBAM alone. RepMCSwin therefore compensates for the limitation of attention in precise localization. Meanwhile, combining MGFPN with MFPCBAM increases precision from 89.6% to 90.3% relative to MGFPN alone, indicating that MFPCBAM can partially suppress the background false positives introduced by broad multi-scale aggregation. These results suggest that MGFPN expands the coverage of target evidence, MFPCBAM filters irrelevant responses, and RepMCSwin restores boundary and localization details. The three components consequently address three related but non-equivalent requirements: finding the target, distinguishing it from the background, and accurately enclosing it.

The complete model demonstrates that the third component is valuable because it compensates for the remaining limitation of the preceding pair, rather than merely increasing the parameter count. On SeaShips, compared with the strongest two-component configuration, MFPCBAM + RepMCSwin, the complete model increases recall from 92.6% to 93.2%,  $\text{mAP}@0.5$  from 96.8% to 97.4%, and  $\text{mAP}@0.5:0.95$  from 74.8% to 75.1%, while precision also improves slightly. On the infrared dataset, the complete model increases recall from 88.8% to 91.0%,  $\text{mAP}@0.5$  from 93.5% to 94.1%, and  $\text{mAP}@0.5:0.95$  from 65.8%

to 66.9%, with only a minor 0.2-percentage-point decrease in precision. The particularly large recall gain in the infrared setting indicates that, after RepMCSwin provides boundary-sensitive representations and MFPCBAM filters background interference, MGFPN can further exploit cross-level semantic and spatial information to recover weak, small, or low-contrast ships.

#### 4.2. Comparison with Mainstream Detectors and Practical Implications

Compared with the mainstream one-stage and two-stage detectors evaluated under the same experimental settings, MCSwin-YOLOv8 achieved the best overall detection accuracy on both datasets. On SeaShips, its F1-score, mAP@0.5, and mAP@0.5:0.95 reached 94.4%, 97.4%, and 75.1%, respectively, exceeding the results of YOLOv5, YOLOv7, YOLOv8, YOLOv9, YOLOv10, RT-DETR, and the evaluated two-stage detectors. Its advantage over recent models such as YOLOv9 and YOLOv10 was particularly evident under the stricter mAP@0.5:0.95 metric, indicating more reliable target localization across different IoU thresholds.

A similar performance pattern was observed on the infrared maritime ship dataset. MCSwin-YOLOv8 achieved an F1-score of 91.2%, an mAP@0.5 of 94.1%, and an mAP@0.5:0.95 of 66.9%. The model outperformed YOLOv9 and YOLOv10 in all three metrics and also achieved higher detection accuracy than RT-DETR, Faster R-CNN, Cascade R-CNN, RetinaNet, DCNv2, and SSD. These results demonstrate that the proposed feature-learning strategy remains effective when the imaging modality changes from visible light to infrared. The consistent improvements across the two datasets indicate that MCSwin-YOLOv8 can learn scale-aware and boundary-sensitive representations rather than relying only on modality-specific appearance information.

The visualization experiments provide qualitative support for the quantitative results. In challenging infrared scenes, MCSwin-YOLOv8 produces fewer missed detections and false detections than the comparison models, particularly for targets with weak boundaries and strong background interference. RepMCSwin strengthens local edge and multi-scale feature extraction, MFPCBAM suppresses irrelevant background responses, and MGFPN facilitates the interaction of semantic and spatial information across different feature levels. Their combination consequently improves both category discrimination and localization reliability in complex maritime environments.

The accuracy improvements are accompanied by an increase in model complexity. MCSwin-YOLOv8 contains 19.29 M parameters and requires 56.7 G FLOPs, compared with 11.12 M parameters and 28.5 G FLOPs for the original YOLOv8. Nevertheless, the proposed model remains substantially smaller than several two-stage detectors, including Faster R-CNN and Cascade R-CNN, while achieving higher detection accuracy. These results indicate that the additional computational cost is converted into meaningful improvements in feature representation, difficult-category recognition, and high-IoU localization. Consequently, MCSwin-YOLOv8 provides a competitive accuracy–complexity trade-off for maritime surveillance and ship safety applications.

## 5. Conclusions

This study proposed MCSwin-YOLOv8 to address the difficulties associated with ship detection under large scale variations, high inter-class visual similarity, weak target boundaries, and complex maritime backgrounds. The model integrates three complementary components. First, the RepMCSwin backbone enhances multi-scale representation and fine-grained boundary modeling through a convolution-oriented architecture. Second, MFPCBAM applies parallel channel and spatial attention to strengthen ship-related responses and suppress background interference. Third, MGFPN improves cross-scale

feature interaction and preserves information that is important for detecting small and weakly represented ships.

Experiments on the public SeaShips dataset and a private infrared maritime ship dataset demonstrated the effectiveness of the proposed approach. MCSwin-YOLOv8 achieved an F1-score of 94.4%, an mAP@0.5 of 97.4%, and an mAP@0.5:0.95 of 75.1% on SeaShips. On the infrared dataset, it achieved an F1-score of 91.2%, an mAP@0.5 of 94.1%, and an mAP@0.5:0.95 of 66.9%. The substantial improvements for difficult categories, particularly fishing boats and sailboats in infrared images, indicate that the proposed model is effective in detecting ships with weak boundaries and visually similar appearances. The ablation experiments further confirmed that RepMCSwin, MFPCBAM, and MGFPN contribute complementary improvements to feature representation, background suppression, and multi-scale fusion.

Several limitations should also be acknowledged. First, the experiments were conducted on only two datasets, and the private infrared dataset may not fully represent the diversity of real maritime environments. The robustness of the model under fog, rain, strong sea clutter, glare, occlusion, dense traffic, and long-distance small-target conditions therefore remains to be verified. Second, the present evaluation focuses primarily on detection accuracy and computational complexity; inference latency, memory consumption, energy usage, and stability on embedded maritime hardware were not systematically evaluated. Third, machine learning models can be highly sensitive to random weight initialization and data partitioning strategies, which may affect experimental robustness and reproducibility [30,31]. Due to computational resource constraints during the revision period, reporting the mean and standard deviation over several runs was not feasible. Future work should expand the infrared dataset, conduct cross-dataset and multi-seed validation, and evaluate the model under a broader range of adverse maritime conditions. Model pruning, knowledge distillation, lightweight convolution, and hardware-aware deployment should also be investigated to achieve a better balance between detection accuracy and real-time performance.

**Author Contributions:** Conceptualization, Y.R. and G.W.; methodology, G.W.; formal analysis, G.W.; data curation, G.W.; writing—original draft preparation, G.W.; writing—review and editing, Y.R.; funding acquisition, Y.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Ministry of Agriculture and Rural Affairs of China grant number B050102.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The SeaShips dataset used in this study is publicly available from its original source. The infrared maritime ship dataset is subject to restrictions imposed by the data providers and is available from the corresponding author upon reasonable request and with permission from the providers. They can be requested from the author at wgh\_dlou@163.com.

**Acknowledgments:** The authors would like to thank China State Shipbuilding Corporation No. 717 Research Institute and Guide Infrared Technology Co., Ltd., for supporting the infrared maritime ship dataset used in this study. This work was supported by the special fiscal fund project of the Ministry of Agriculture and Rural Affairs of China (B050102).

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

|            |                                                                                          |
|------------|------------------------------------------------------------------------------------------|
| AIS        | Automatic Identification System                                                          |
| AP         | Average Precision                                                                        |
| C2f        | Cross-Stage Partial Bottleneck with Two Convolutions                                     |
| CBAM       | Convolutional Block Attention Module                                                     |
| CBS        | Convolution–Batch Normalization–SiLU Block                                               |
| CNN        | Convolutional Neural Network                                                             |
| ConvMLP    | Convolutional Multilayer Perceptron                                                      |
| CPU        | Central Processing Unit                                                                  |
| CUDA       | Compute Unified Device Architecture                                                      |
| DCNv2      | Deformable Convolutional Network Version 2                                               |
| F1         | F1 Score                                                                                 |
| FLOPs      | Floating-Point Operations                                                                |
| FN         | False Negative                                                                           |
| FP         | False Positive                                                                           |
| FPN        | Feature Pyramid Network                                                                  |
| GAN        | Generative Adversarial Network                                                           |
| GFPN       | Generalized Feature Pyramid Network                                                      |
| GPU        | Graphics Processing Unit                                                                 |
| IoU        | Intersection over Union                                                                  |
| LN         | Layer Normalization                                                                      |
| mAP        | Mean Average Precision                                                                   |
| MCBot      | Multi-Scale Convolutional Bottleneck                                                     |
| MCSwin     | Multi-Scale Convolutional Swin Block                                                     |
| MFPCBAM    | Multi-Feature Parallel Convolutional Block Attention Module                              |
| MGFPN      | Modified Generalized Feature Pyramid Network                                             |
| MLP        | Multilayer Perceptron                                                                    |
| PAFPN      | Path Aggregation Feature Pyramid Network                                                 |
| PConv      | Partial Convolution                                                                      |
| PASCAL VOC | Pattern Analysis, Statistical Modelling and Computational Learning Visual Object Classes |
| R-CNN      | Region-Based Convolutional Neural Network                                                |
| RDSC       | Reverse Depthwise Separable Convolution                                                  |
| RepMCSwin  | Re-Parameterized Multi-Scale Convolutional Swin Backbone                                 |
| RepVGG     | Re-Parameterized Visual Geometry Group Network                                           |
| RT-DETR    | Real-Time Detection Transformer                                                          |
| SAR        | Synthetic Aperture Radar                                                                 |
| SGD        | Stochastic Gradient Descent                                                              |
| SPP        | Spatial Pyramid Pooling                                                                  |
| SPPF       | Spatial Pyramid Pooling–Fast                                                             |
| SSD        | Single Shot MultiBox Detector                                                            |
| SW-MSA     | Shifted-Window Multi-Head Self-Attention                                                 |
| TP         | True Positive                                                                            |
| W-MSA      | Window-Based Multi-Head Self-Attention                                                   |
| YOLO       | You Only Look Once                                                                       |

## References

1. Sun, D.; Zhang, Y.; Han, X.; Xia, Z.; Wang, G. Improved YOLOv5s-based military target detection algorithm for complex scenes. *J. Proj. Rocket. Missiles Guid.* **2025**, *45*, 45–52. (In Chinese)
2. Ke, X.; Zhang, X.; Zhang, T.; Shi, J.; Wei, S. SAR ship detection based on Swin Transformer and feature-enhanced feature pyramid network. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Kuala Lumpur, Malaysia, 17–22 July 2022; pp. 2163–2166.

3. Li, K.; Zhang, M.; Xu, M.; Tang, R.; Wang, L.; Wang, H. Ship detection in SAR images based on feature-enhanced Swin Transformer and adjacent feature fusion. *Remote Sens.* **2022**, *14*, 3186. [\[CrossRef\]](#)
4. Zha, M.; Qian, W.; Yang, W.; Xu, Y. Multifeature transformation and fusion-based ship detection with small targets and complex backgrounds. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [\[CrossRef\]](#)
5. Huang, L.; Wang, F.; Zhang, Y.; Xu, Q. Fine-grained ship classification by combining CNN and Swin Transformer. *Remote Sens.* **2022**, *14*, 3087. [\[CrossRef\]](#)
6. Xie, Y.; Zhu, L.; Li, Y.; Hu, G.; Chen, J. Ship Small Target Detection Method Based on an Improved YOLOv4. *Command. Control Simul.* **2025**, 1–8. (In Chinese). Available online: <https://link.cnki.net/urlid/32.1759.TJ.20250821.1118.034> (accessed on 18 August 2026). [\[PubMed\]](#)
7. Cao, Z.; Ma, Y.; Cheng, X.; Hu, W. Inland ship target detection method based on PEW-YOLOv8. *J. Transp. Inf. Saf.* **2025**, *43*, 36–43. (In Chinese) [\[CrossRef\]](#)
8. Chen, Z.; Chen, D.; Zhang, Y.; Cheng, X.; Zhang, M.; Wu, C. Deep learning for autonomous ship-oriented small ship detection. *Saf. Sci.* **2020**, *130*, 104812. [\[CrossRef\]](#)
9. Zhou, J.; Jiang, P.; Zou, A.; Chen, X.; Hu, W. Ship target detection algorithm based on improved YOLOv5. *J. Mar. Sci. Eng.* **2021**, *9*, 908. [\[CrossRef\]](#)
10. Han, X.; Zhao, L.; Ning, Y.; Hu, J. ShipYOLO: An enhanced model for ship detection. *J. Adv. Transp.* **2021**, 1–11. [\[CrossRef\]](#)
11. Liu, T.; Pang, B.; Zhang, L.; Yang, W.; Sun, X. Sea surface object detection algorithm based on YOLOv4 fused with reverse depthwise separable convolution for USV. *J. Mar. Sci. Eng.* **2021**, *9*, 753. [\[CrossRef\]](#)
12. Jin, M.; Li, X.; Zhang, L. Ship target detection algorithm based on enhanced YOLOv4. *Ship Eng.* **2022**, *44*, 100–106. (In Chinese)
13. Zhao, Y.; Guo, H.; Jiao, H.; Zhang, J. Application of YOLOv4 integrated with hybrid-domain attention in ship detection. *Comput. Mod.* **2021**, *9*, 75–82. (In Chinese)
14. Luo, F.; Liu, Y.; He, D. Multi-scale ship detection with adaptive feature fusion in complex scenes. *J. Comput. Appl.* **2023**, *43*, 1–10. (In Chinese) [\[CrossRef\]](#)
15. Guo, L.; Yu, H.; Zhou, Z. Nearshore ship detection method based on SimAM attention mechanism. *J. Harbin Inst. Technol.* **2022**, *55*, 14–21. (In Chinese)
16. Li, J.; Zhang, D.; Fan, Y.; Yang, J. Lightweight ship target detection algorithm based on improved YOLOv5. *J. Comput. Appl.* **2023**, *43*, 1–10. (In Chinese) [\[CrossRef\]](#)
17. Cui, X.; Zheng, Q.; Li, J.; Jiang, B.; Li, S.; Liu, J. A parallel convolutional neural network-transformer model for underwater target recognition based on multimodal feature learning. *Proc. Inst. Mech. Eng. Part M. J. Eng. Marit. Environ.* **2024**, *238*, 943–953. [\[CrossRef\]](#)
18. Yu, N.; Fan, X.; Deng, T.; Mao, G. Ship detection algorithm in complex backgrounds based on multi-head self-attention. *J. Zhejiang Univ. (Eng. Sci.)* **2022**, *56*, 2392–2402. (In Chinese)
19. Lu, D.; Tang, H.; Teng, L.; Tan, J.; Wang, M.; Tian, Z.; Wang, L. Multiscale Feature-Based Infrared Ship Detection. *Appl. Sci.* **2024**, *14*, 246. [\[CrossRef\]](#)
20. Zhang, P.; Zhu, P.; Sun, Z.; Ding, J.; Zhang, J.; Dong, J.; Guo, W. Research on Improved Lightweight YOLOv5s for Multi-Scale Ship Target Detection. *Appl. Sci.* **2024**, *14*, 6075. [\[CrossRef\]](#)
21. Gao, F.; Tian, Y.; Wu, Y.; Zhang, Y. ST-YOLOv8: Small-Target Ship Detection in SAR Images Targeting Specific Marine Environments. *Appl. Sci.* **2025**, *15*, 6666. [\[CrossRef\]](#)
22. Zeng, X.; Li, J.; Xiong, R. Foggy Ship Detection with Multi-Scale Feature and Attention Fusion. *Appl. Sci.* **2026**, *16*, 1475. [\[CrossRef\]](#)
23. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in Transformer. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 15908–15919.
24. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*; IEEE: New York, NY, USA, 2021; pp. 10012–10022.
25. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
26. Chen, J.; Kao, S.; He, H.; Zhuo, W.; Wen, S.; Lee, C.-H.; Chan, S.-H.G. Run, don't walk: Chasing higher FLOPS for faster neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, BC, Canada, 17–24 June 2023; pp. 12021–12031.
27. Zhang, Y.; Xie, F.; Huang, L.; Shi, J.; Yang, J.; Li, Z. A lightweight one-stage defect detection network for small objects based on dual attention mechanism and PAFPN. *Front. Phys.* **2021**, *9*, 708097. [\[CrossRef\]](#)
28. Jiang, Y.; Tan, Z.; Wang, J.; Sun, X.; Lin, M.; Li, H. GiraffeDet: A heavy-neck paradigm for object detection. *arXiv* **2022**, arXiv:2202.04256.
29. Shao, Z.; Wu, W.; Wang, Z.; Du, W.; Li, C. Seaships: A large-scale precisely annotated dataset for ship detection. *IEEE Trans. Multimed.* **2018**, *20*, 2593–2604. [\[CrossRef\]](#)

30. Wang, S. Automated Fault Detection and Diagnosis of AHUs via Tabular-Based Methods Using Operational Data from a Large Office Building. *J. Comput. Civ. Eng.* **2026**, *40*. [[CrossRef](#)]
31. Wang, S. Fault-class coverage-aligned combined training for AFDD of AHUs across multiple buildings. *Sci. Rep.* **2025**, *15*, 41192. [[CrossRef](#)] [[PubMed](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.