Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.047062 Published Online: 18 March 2024

ARTICLE

Unmanned Ship Identification Based on Improved YOLOv8s Algorithm

Chun-Ming Wu[1], Jin Lei[1,*], Wu-Kai Liu[1], Mei-Ling Ren[1] and Ling-Li Ran[2]

[1]Key Laboratory of Modern Power System Simulation and Control & Renewable Energy Technology, School of Electrical Engineering, Northeast Power University, Jilin, 132012, China

[2]School of Biomedical Engineering, Taiyuan University of Technology, Taiyuan, 030600, China

*Corresponding Author: Jin Lei. Email: 2202200376@neepu.edu.cn

Received: 23 October 2023 Accepted: 29 November 2023

ABSTRACT

Aiming at defects such as low contrast in infrared ship images, uneven distribution of ship size, and lack of texture details, which will lead to unmanned ship leakage misdetection and slow detection, this paper proposes an infrared ship detection model based on the improved YOLOv8 algorithm (R_YOLO). The algorithm incorporates the Efficient Multi-Scale Attention mechanism (EMA), the efficient Reparameterized Generalized-feature extraction module (CSPStage), the small target detection header, the Repulsion Loss function, and the context aggregation block (CABlock), which are designed to improve the model's ability to detect targets at multiple scales and the speed of model inference. The algorithm is validated in detail on two vessel datasets. The comprehensive experimental results demonstrate that, in the infrared dataset, the YOLOv8s algorithm exhibits improvements in various performance metrics. Specifically, compared to the baseline algorithm, there is a 3.1% increase in mean average precision at a threshold of 0.5 (mAP (0.5)), a 5.4% increase in recall rate, and a 2.2% increase in mAP (0.5:0.95). Simultaneously, while less than 5 times parameters, the mAP (0.5) and frames per second (FPS) exhibit an increase of 1.7% and more than 3 times, respectively, compared to the CAA_YOLO algorithm. Finally, the evaluation indexes on the visible light data set have shown an average improvement of 4.5%.

KEYWORDS

Unmanned ships; R_YOLO; EMA; CSPStage; YOLOv8s

1 Introduction

With the advancement of the marine economy, the utilization of unmanned ship equipment has become prevalent in various domains including disaster rescue and relief. In particular, infrared imaging technology has gained significant application in detecting crewless ships due to its exceptional characteristics, such as robust anti-interference capability and all-weather operability [1]. Due to the absence of intricate texture details and the uneven distribution of sizes among infrared ships, the task of accurately identifying targets becomes significantly complex. Therefore, the target detection's accuracy and detection speed are considered the primary prerequisites for improving the intelligence of uncrewed vessels [2].

Deep learning-based algorithms for target detection have demonstrated a significant performance improvement in detecting infrared ships. A large number of target detection algorithms exist, such as two-stage algorithms (region-based convolutional neural networks (R-CNN) [3]), and one-stage algorithms (YOLO [4], Single shot multibox detector (SSD) [5]). Currently, single-stage algorithms are extensively employed in diverse scenarios. For instance, the enhanced YOLOv5s algorithm is utilized to identify mushroom log pollution [6]. Additionally, the optimized YOLOv7 algorithm is employed to recognize traffic signs in autonomous vehicles [7], while the detection of strawberry maturity is achieved by utilizing the YOLOv8 algorithm [8]. In this paper, the YOLOv8s algorithm (https://github.com/ultralytics/ultralytics) is selected as the foundational algorithm. This algorithm is known for its consideration of both accuracy and inference speed. However, a significant research challenge lies in extracting the complete set of semantic features for small targets without compromising the extraction of semantic information for larger and medium-sized targets. In addressing this issue, a range of approaches have been previously suggested. For instance, Chen et al. [9] introduced a lightweight algorithm for detecting garbage on water surfaces based on improved YOLOv5s. This approach offers a promising solution for real-time litter detection on water surfaces. Tang et al. [10] have proposed a method for elevator button recognition that combines YOLOv5. This approach offers a viable solution to efficiently identifying elevator buttons by service robots. Han et al. [11] employed a bidirectional feature pyramid network as a feature extraction network to efficiently combine high-level semantic information with underlying spatial features. This approach aims to improve the detection accuracy of safety helmets worn by non-motorized individuals.

The paper is organized as follows: Section 2 shows the state of the research, existing problems, and solutions for Ship target detection. Section 3 provides a brief overview of the characteristics of the two datasets and YOLOv8. Subsequently, a detailed description of the improved six modules is provided. Section 4 is an experiment and the result of the analysis. The dissertation concludes with our conclusions and prospects for future work.

2 Related Work

Currently, there are two main categories of ship image detection methods based on AI algorithms. The first category involves traditional neural network algorithms, which include detecting ships in infrared images, manually screening ship texture and other features, and subsequently inputting these feature parameters into the traditional neural network. The other approach relies on deep learning convolutional neural network algorithms, such as Swin Transformerv2 [12]. The infrared image is directly input into the deep convolutional neural network structure, and the network can automatically learn the semantic features in the infrared image without manual intervention. Then, the ship's position in the image is determined according to the learned position information to realize end-to-end intelligent detection and positioning from the original infrared ship image to the position of the output ship to avoid the influence of human factors on feature screening and parameterization [13]. Therefore, this paper further explores the automated infrared ship target detection algorithm.

To address the issue of lightweight marine ship detection models, Cheng et al. [14] introduced the improved YOLOv5 model, which aims to enhance detection speed. However, it is important to note that its AP (0.5:0.95) is only 48%. Feng et al. [15] introduced a lightweight network of multi-scale feature fusion Transformer by incorporating the attention mechanism. The study yielded favorable outcomes across four different datasets. Zhang et al. [16] proposed the lightweight Yolov5l model. Comparing the L model to the current model, it is observed that the parameter amount is reduced

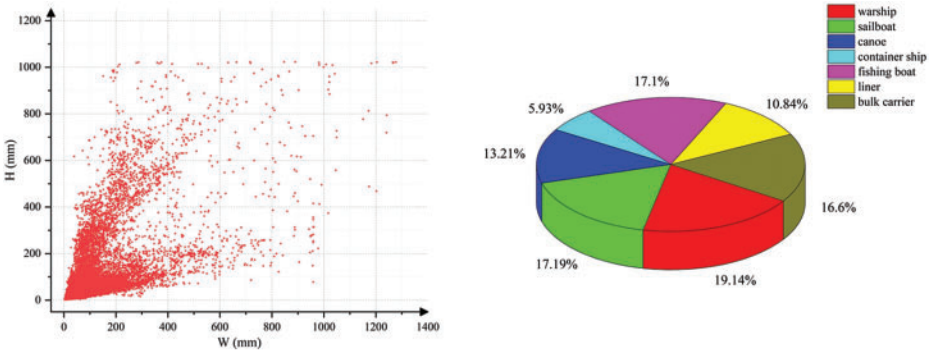
by 50%. However, the mAP (0.5) is only 92.03%. Aiming to address the issue of inadequate multi-scale detection performance in ship detection models, Wu et al. [17] proposed a multi-scale feature fusion module to fuse deep, shallow, local and global features to improve the accuracy of multi-scale ship detection in complex environments. Kong et al. [18] implemented the task adaptive hybrid migration strategy and incorporated the Space-Adjusted module into the prototype network. The proposed methodology can enhance the precision in identifying unfamiliar categories of ships. Guo et al. [19] incorporated the Inception branch and the Softmax function into the CNN network to identify different types of ships. Their study yielded promising outcomes across various public datasets. Ye et al. [20] introduced a Combined-Attention-Augmented (CAA) technique to capture long-distance contextual information of small targets. The mAP (0.5) achieves a value of 94.81%. However, it should be noted that the model parameters are relatively large, and the detection speed is slow. Consequently, this may render the model unsuitable for deployment on mobile terminals in the final stage.

High detection speed and recognition accuracy are crucial in achieving the intellectualization of uncrewed ships. Therefore, the network structure of YOLOv8 is improved in the following manner: Firstly, the introduction of the repulsion loss function aims to address the issue of occlusion in ship detection. Furthermore, the inclusion of the small target detection head and CABlock module serves to augment the model's capacity to capture the semantic characteristics of diminutive targets. Finally, the EMA attention mechanism and the CSPStage module are incorporated into the model to dynamically fuse and extract semantic information at multiple scales.

3 Materials and Methods

3.1 Dataset Analysis and Data Processing

This paper uses the infrared maritime vessel dataset provided by Shandong Yantai Arrow Optoelectronics Technology Co., Ltd. (China), which contains several scenarios of cruise ships, bulk carriers, warships, sailboats, and other targets on the harbors and seashores. This dataset has a total of 8002 images, and the resolution of each image is 640 × 512. The sample images in the dataset are divided according to 8:1:1, i.e., 6738:632:632. The size distribution of the ship label box is depicted in Fig. 1a, and it is evident that the dataset falls under the classification of multi-scale target detection. The proportion of ship types is depicted in Fig. 1b, with fishing vessels accounting for the largest proportion.



(a) Scatter Diagram (b) Pie Chart

Figure 1: Dataset analysis

This paper also utilizes an additional dataset of visible light ship data sourced from various publicly available datasets and contains only one category of ships [14]. The dataset comprises a total of 3200 images. We divided our dataset into a training set, a validation set and a test set according to the ratio of 7:1:2.

3.2 YOLOv8 Network Architecture

The YOLO algorithm is a one-stage target detection algorithm that converts object detection into a regression problem, and its algorithm is widely used due to its high detection speed and low-cost overhead [21]. YOLOv8 has faster detection and higher accuracy than previous versions. As shown in Fig. 2, the YOLOv8 network structure includes data preprocessing, trunk network, neck network, and detection head.

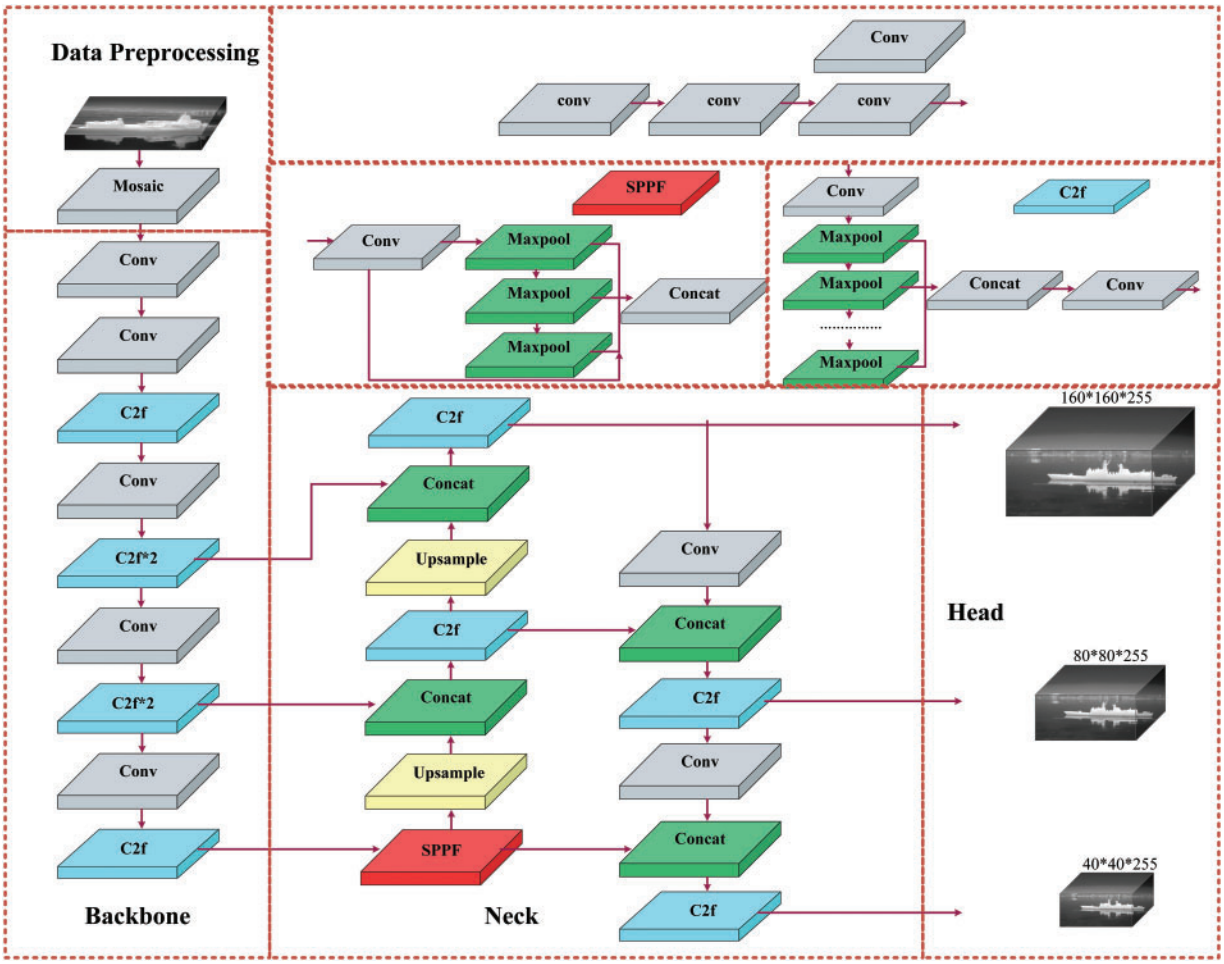


Figure 2: Framework of YOLOv8 algorithm

3.3 R_YOLO Detection Model

The R_YOLO detection model framework is shown in Fig. 3. R_YOLO consists of three main components: data preprocessing, network training, and model inference. The enhanced module can

be succinctly described as follows: (a) Incorporate a small target detection layer. (b) An EMA attention mechanism was incorporated between the torso and the neck. (c) Incorporate the repulsion loss function. (d) The CSPStage module is being introduced. (e) Embedding the CABlock module.

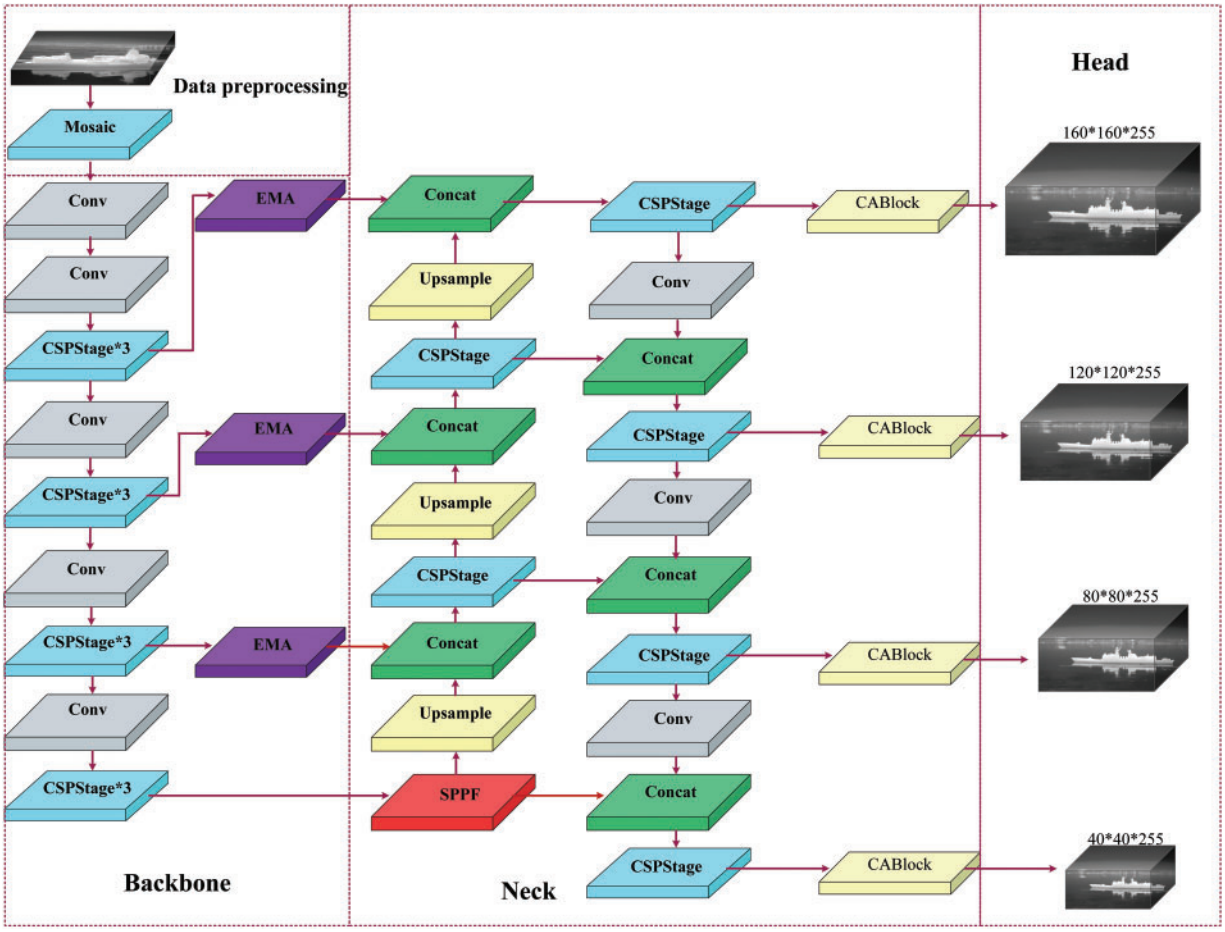


Figure 3: Framework of R_YOLO algorithm

3.3.1 Adding the Small Target Detection Layer

Learning useful features in deep networks is difficult due to the small pixel proportion of small target vessels. There are two main reasons: (1) During the feature extraction process, the feature map undergoes multiple downsampling operations, resulting in a reduction of feature information. (2) With the proliferation of network layers, certain ship characteristics may be compromised, thereby hindering the achievement of precise positioning. The inclusion of small target detection can offer comprehensive positional information and mitigate the issue of missed detection in ship detection.

3.3.2 EMA Attention Mechanism

This paper introduces the EMA attention module [22] to integrate multi-scale semantic information and mitigate superficial image noise. To optimize the network's performance, it is partitioned into feature grouping and multi-scale structure. The EMA attention mechanism is depicted in Fig. 4.

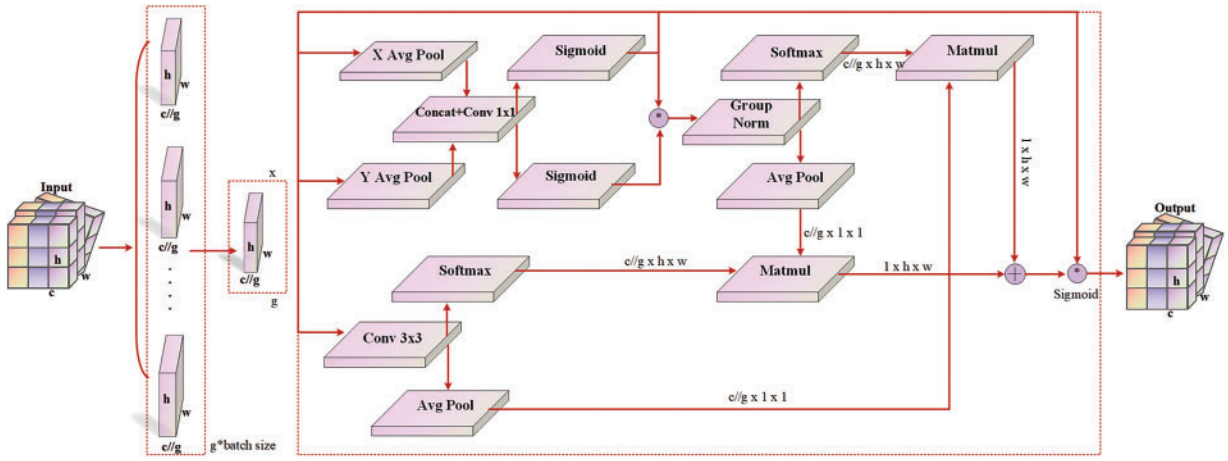


Figure 4: EMA attention mechanism

(1) For a feature map with input dimensions $H \times W \times C$, EMA first maps the given input features set of types that can be represented by $X_i \in R_i^{G//C\times H\times W}$. The EMA attention mechanism uses three parallel routes to extract attention weight descriptors for grouped feature graphs; two of the parallel paths are 1×1 convolutional branches, and the third parallel path is a 3×3 convolutional branch.

(2) For a feature map with input dimension $G//H \times W \times C$, the EMA attention mechanism first uses a pooling kernel of size to encode the coordinates of each channel along the horizontal and vertical directions, respectively. Direction-aware attention features $\frac{1}{H}\sum_{0<j<H} X_c(j,w)$ and $\frac{1}{w}\sum_{0<j<w} X_c(h,i)$ are acquired and transformed into C through a shared 1×1 convolution, as in Eq. (1).

$$f = \delta\left(F_1\left[\frac{1}{H}\sum_{0<j<H} X_c(j,w), \frac{1}{W}\sum_{0<i<w} X_c(h,i)\right]\right) \tag{1}$$

This paper aggregates the two-channel attention maps within each group by simple multiplication and implements different cross-channel interaction functions between two parallel routes in a 1×1 branch, as shown in Eq. (2).

$$h_c(i,j) = X_c(i,j) \times f_{(i)}^{w} \times f_{(j)}^{h} \tag{2}$$

f in the above equation is the intermediate feature map of spatial information in the horizontal and vertical directions; [. , .] represents the spatial connection operation; while representing the non-linear activation function used by the algorithm. $X_c(i,j)$ is the input image.

(3) 3×3 branching via 3×3 convolution captures local cross-channel interactions to expand the feature space, and EMA can adjust the importance of different channels. Also, the precise spatial structure information is retained in the channel, as shown in Eq. (3).

$$M = f\left(\frac{1}{H}\sum_{0<j<H} X_c(j,w)\right) \times \delta_s(G(h_c(i,j))) + \delta_s\left(\frac{1}{H}\sum_{0<j<H} X_c(j,w)\right) \times f(G(h_c(i,j))) \tag{3}$$

(4) The final output of the EMA module, as in Eq. (4).

$$Y_c(i,j) = \delta(M \times X_c(i,j)) \tag{4}$$

In the above equation, $Y_c(i,j)$ is the output image, δ_s denotes the Softmax activation operation, and G denotes group normalization.

3.3.3 Repulsion Loss

Given the significant occurrence of numerous vessels impeding each other in the dockyard setting. To address the aforementioned issue, this study proposes using a repulsion loss function [23]. As demonstrated in Eq. (5), it comprises three distinct elements.

$$L = L_{Atter} + \alpha \times L_{RepGT} + \beta \times L_{RepBox} \tag{5}$$

where L_{Atter} is the attractive term, while L_{RepGT} and L_{RepBox} are the repulsive terms, and the coefficients α and β are used as weights to balance the auxiliary loss.

L_{Atter}: Attractive losses are typically measured between ground truth frames by the $Smooth_{L_1}$ distance metric using existing bounding box regression techniques, requiring that the prediction frame be close to its real frame. Its smoothing coefficient is set to 3 for the best results, as in Eq. (6).

$$L_{Atter} = \frac{\sum_{P \in P+} Smooth_{L_1}\left(B^P, G^P_{Atter}\right)}{|P_+|} \tag{6}$$

Given a proposal P, this paper assigns the ground truth frame of the largest IoU as the specified target, where $G^P_{Attr} = \arg\max_{G \in \vartheta} IoU(G,P)$, B^P is the prediction frame from the proposed P regression.

L_{RepGT}: The exclusion loss requires that the prediction frame move away from the proposals of its neighboring ground-truth objects, which are not its targets. As in Eq. (7).

$$L_{RepGT} = \frac{\sum_{P \in P+} Smooth_{\ln}\left(IoG\left(B^P, G^P_{Rep}\right)\right)}{|P_+|} \tag{7}$$

where $Smooth_{\ln} = \begin{cases} -\ln(1-x) & x \le \sigma \\ \dfrac{x-\sigma}{1-\sigma} & x > \sigma \end{cases}$. Given a proposal $P \in P_+$, the repel ground reality object is defined as the ground reality object with the largest IoU area in addition to its designated target. Where $G^P_{Rep} = \underset{G \in \vartheta \setminus \{G^P_{Attr}\}}{\arg\max} IoU(G,P)$, $IoG(B,G) = \dfrac{area(B \cap G)}{area(G)}$ is the overlap between B and G, intersected by the ground reality box.

L_{RepBox}: Since NMS can significantly affect the detection of overlapping or juxtaposition of ships, L_{RepBox} is further proposed as shown in Eq. (8).

$$L_{RepBox} = \frac{\sum_{i \neq j} Smooth_{\ln}\left(IoU\left(B^{P_i}, B^{P_j}\right)\right)}{\sum_{i \neq j} I\left[IoU\left(B^{P_i}, B^{P_j}\right) > 0\right] + \varepsilon} \tag{8}$$

where I is a constant. It is hoped that the overlap of prediction boxes B and C will be as small as possible, which makes the detector more robust to ship overlap scenarios.

3.3.4 CSPStaget

To facilitate the comprehensive exchange of high-level semantic and low-level spatial information of the ship, and to meet the requirements of real-time target detection, a CSPStaget feature extraction module is proposed [24], as illustrated in Fig. 5. CSPStaget is a feature extraction module that is based

on the GFPN [25]. This module not only enhances the interaction between features, but also considers the latency that is generated due to the over-complexity of the module. It is accomplished by considering the following three aspects: (1) In terms of overall feature fusion, this paper reduces floating-point operations by flexibly controlling the number of channels at different scales under the constraint of limited computational overhead. (2) Removing the GFPN enhances feature interactions by queen-fusion, which brings a lot of extra up-sampling operations and reduces unnecessary delay time. (3) In the feature fusion block, the CSPNet [25] is employed to replace the original feature fusion method that relies on 3×3 convolution. This replacement is combined with the re-parameterization mechanism and the high-efficiency layer aggregation network to enhance accuracy without significantly increasing computational overhead.

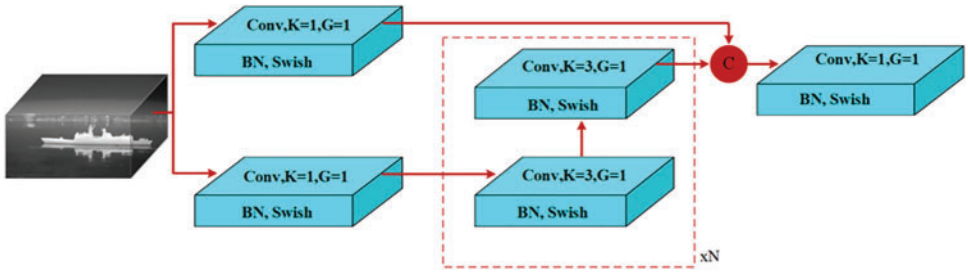


Figure 5: CSPStage

3.3.5 CABlock

After the network has aggregated various levels of features, there remains residual local spatial information within the feature pyramid. In previous studies, researchers have incorporated multiple visual attention blocks within the network's backbone to enhance the overall perceptual acuity. However, this design introduces too much useless background information, increasing the computational overhead. Hence, the introduction of the CABlock module aims to acquire comprehensive spatial context information at a global level, thereby further enhancing the features. Where pixel feature information is sufficiently rich, it is inhibited from aggregating features from other spatial locations, thus effectively fusing local and global features while reducing information confusion [26]. CABlock's framework is shown in Fig. 6. In each module, the pixel-by-pixel spatial contexts are aggregated in the following way, as shown in Eq. (9).

$$Q_i^j = P_i^j + \alpha_i^j . \sum_{j=1}^{N_i} \left[\frac{\exp\left(w_k P_i^j\right)}{\sum_{m=1}^{N_i} \exp\left(w_k P_i^m\right)} . w_v P_i^j \right] \tag{9}$$

$$\alpha_i^j = \frac{\exp\left(w_a P_i^j\right)}{\sum_{n=1}^{N_i} \exp\left(w_a P_i^n\right)} \tag{10}$$

where α_i^j is shown in Eq. (10), Q_i^j and P_i^j denote the input and output feature maps for level i in the feature pyramid. Each feature map consists of N_i pixels. $j, m \in \{1, N_i\}$ denotes the index of each pixel, and w_k and w_v are the linear transformation matrices used to project the feature map. The above formulation simplifies the widely used self-attention mechanism by replacing the matrix multiplication between query and key with a linear transformation that significantly reduces the computational overhead.

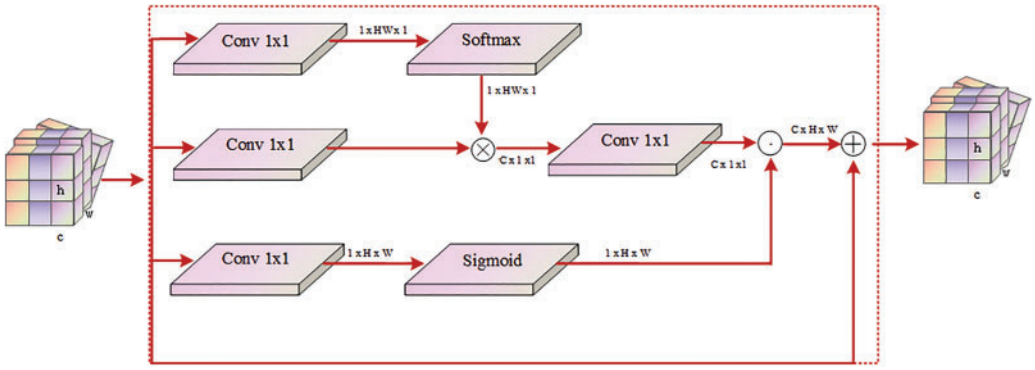


Figure 6: CABlock

3.4 Training Parameter Settings

In the training process of the YOLOv8 model, this study employed the stochastic gradient descent (SGD) algorithm to optimize the loss function. Considering t the utilization of computer memory, the batch size has been determined as 32, while the number of threads has been set to 16. To achieve the optimal model, it is necessary to perform 220 training iterations. It is advisable to initialize the random seed to 1 and conclude the final 20 rounds of mosaic enhancement. The computer configuration utilized in the experiment is presented in Table 1.

Table 1: Computer configuration

Platform	Configuration information
System	Ubuntu 20.04
GPU	NVIDIA GeForce RTX A5000(24G)
CPU	15 vCPU AMD EPYC 7543 32-Core Processor
Language	Python 3.8.0
GPU calculate platform	CUDA 11.8 cuDNN 8.2.0
Deep learning framework	Pytorch 2.0.0

3.5 Evaluation Indicators

Precision, Recall, and mAP serve as critical metrics for evaluating the accuracy of a network. The precision metric denotes the ratio of accurate predictions in the prediction outcomes, while the Recall signifies the ratio of accurate predictions among all targets. Each category's cumulative score is determined by the P-R curve, and the AP values of all categories are averaged to obtain mAP.

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

$$AP = \int_0^1 p(r)\, dr \tag{13}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{14}$$

where TP is True Positive, FP is False Positive, FN is False Negative, p(r) is the function of the P-R curve, and K is the number of categories. In this paper, we also use the number of parameters, floating point operations (FLOPs), and FPS to evaluate the model complexity. Where FLOPs denotes the amount of computation required by the model and FPS is the number of frames per second to fill the image.

4 Materials and Methods

4.1 Experimental Settings

In order to demonstrate the superiority of the R_YOLO model, three sets of experiments were conducted to validate the model (Table 2). (1) Ablation experiments were performed to assess the effects of five improved protocols (M1–M5). (2) The proposed R_YOLO model is compared with YOLOv5 [2], YOLOv6 [27], YOLOv7 [28], and YOLOv8 algorithms. This article solely focuses on comparing the models of N, S, M, and L sizes with the consideration of their potential deployment on mobile terminals. (3) Furthermore, the R_YOLO model is compared with eight popular target detection algorithms such as SDD, DAMOYOLO [25], YOLO NAS (https://github.com/Deci-AI/super-gradients/), RetinaNet [29], Faster R-CNN [4], EfficientDet-D3 [30], YOLOX [31], and CAA_YOLO [20]. (4) Finally, a comparison is conducted between the R_YOLO algorithm and the YOLOv8 algorithm using various datasets.

Table 2: Configuration of the three experiments used for model comparison

Experiments	Settings
Ablation	M1: Small target detection head M2: EMA attention mechanism M3: CABlock M4: Repulsion loss M5: CSPStaget
YOLO series algorithm comparison experiment (N, S, M, L)	S0: YOLOv5 S1: YOLOv6 S2: YOLOv7 S3: YOLOv8
Different methods	N0: SDD N1: DAMOYOLO N2: YOLO NAS N3: Faster-RCNN N4: EfficientDet-D3 N5: RetinaNet N6: YOLO X

(Continued)

Table 2 (continued)

Experiments	Settings
	N7: CAA_YOLO N8: R_YOLO
Different datasets	Infrared ship dataset Ship dataset

4.2 Ablation Experiment

To assess the efficacy of the R_YOLO model, ablation experiments were conducted and compared with the original YOLOv8s algorithm (S0). The results of the ablation study are presented in Table 3. The table displays six scenarios (S0–Si), each representing different combinations of five improved strategies (M1–M5). No strategy is implemented in S0 (the reference method), whereas all five strategies are implemented in Si (the proposed method).

Table 3: Ablation experiments

Models	M1	M2	M3	M4	M5	mAP (0.5)	mAP (0.5:0.95)	GFLOPs	P	R	Parameters
S0						0.927	0.664	27.4	0.923	0.875	11110853
S1	1					0.940	0.674	37.8	0.926	0.911	11132732
S2	1	1				0.947	0.681	37.6	0.923	0.914	11146412
S3	1	1	1			0.950	0.680	37.6	0.922	0.931	11146412
S4	1	1	1	1		0.952	0.681	38.4	0.925	0.902	11846580
S5	1	1	1	1	1	**0.958**	**0.686**	55.2	**0.934**	**0.927**	17813482

In this study, we employ the sequential stacking module for comparison. Firstly, incorporating a small target detection layer (M1) into the network yields the most substantial enhancement in mAP. Compared to the S0 (basic model), the mAP exhibits an approximate increase of 1%. Secondly, the EMA attention module (M2) is incorporated to enable the adaptive fusion of multi-scale semantic features. Compared to the initial values of S0 and S1, the observed increase was 2% and 0.7%, respectively. Furthermore, the Repulsion Loss function (M3) has been incorporated, and it should be noted that mutual occlusion of ships is only observed in the port scene. Compared to S2, the mAP only exhibited a marginal increase of 0.3%. The integration of local and global features is further enhanced by embedding the CABlock module (M4). Due to the interplay among modules, the mAP (0.5) only slightly improves. Finally, the CSPStaget module (M5) has been replaced to enhance the model's capability to extract multi-scale features. Compared to S4, the mAP exhibited an increase of approximately 0.6%. At the same time, there are improvements observed in both accuracy and recall rate to varying extents.

4.3 YOLO Series Algorithm Comparison Experiment (N, S, M, L)

To demonstrate the efficacy of R-YOLO, the R-YOLO algorithm is exclusively compared to four distinct-sized models (N, S, M, L) from the YOLO series algorithms. R-YOLO outperforms YOLOv8, YOLOv7, YOLOv6, and YOLOv5 across all versions on four metrics, i.e., Precision, Recall, mAP (0.5), and mAP (0.5:0.95).

Fig. 7 compares the R_YOLO algorithm with other YOLO algorithms in terms of precision, recall, and mAP. The graph's horizontal axis represents the number of parameters, while the vertical axis represents the performance indicators being compared. The proximity of a point in the graph to the upper left corner directly correlates with the effectiveness of the comparison for this particular indicator. As the number of parameters increases, there is no positive correlation between the two types of mAP values depicted in Figs. 7a and 7b. The enhanced R YOLO algorithm outperforms other algorithms in terms of both mAP (0.5) and mAP (0.5:0.95) values. This observation also underscores the intricacy of the dataset, as it is unable to improve the mAP values through parameter stacking. Figs. 7c and 7d depict the graphs comparing precision and recall. The disparity in precision between the R_YOLO algorithm and other algorithms may not be readily apparent. However, in terms of recall, the R_YOLO algorithm surpasses all others significantly. As demonstrated in Table 4, the four indicators of R _ YOLO outperform other algorithms within an accepted range of parameters. This superiority is primarily attributed to incorporating the CSPStaget module and the EMA attention mechanism.

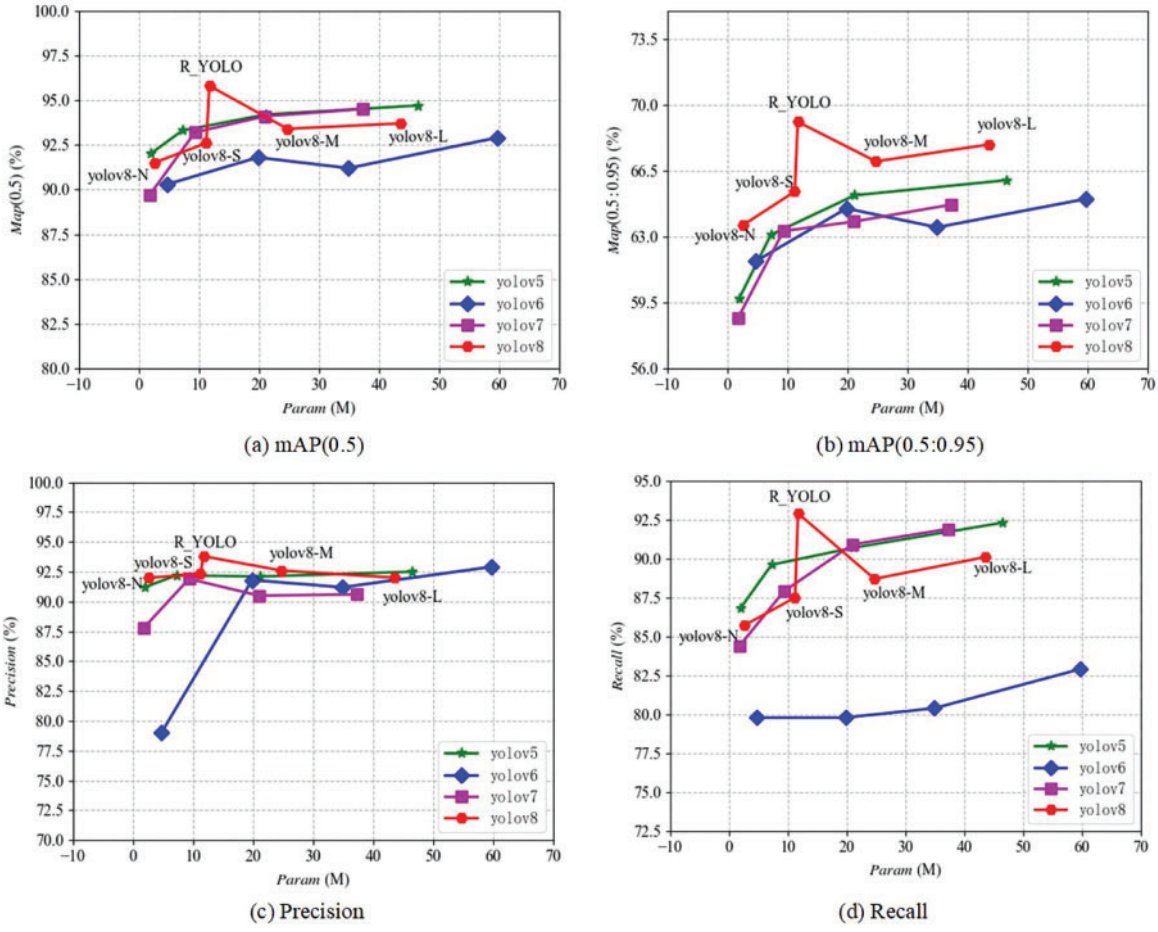


Figure 7: Performance comparison

Table 4: Comparison of YOLOv8, YOLOv7, YOLOv6 and YOLOv5 models

Models	Precision	Recall	Parametes (M)	mAP (0.5)	mAP (0.5:0.95)	GFLOPs
YOLOv8n	0.920	0.857	2.50	0.915	0.636	6.7
YOLOv8s	0.923	0.875	11.11	0.927	0.664	27.4
YOLOv8m	0.926	0.887	24.68	0.934	0.670	61.3
YOLOv8l	0.920	0.901	43.61	0.937	0.679	164.8
YOLOv7n	0.878	0.844	1.77	0.897	0.587	4.3
YOLOv7s	0.919	0.879	9.34	0.932	0.633	26.7
YOLOv7m	0.905	0.909	20.96	0.941	0.638	59.5
YOLOv7l	0.906	0.919	37.3	0.942	0.647	105.2
YOLOv6n	0.740	0.798	4.70	0.903	0.617	11.4
YOLOv6s	0.918	0.798	19.40	0.918	0.645	45.3
YOLOv6m	0.912	0.804	36.60	0.912	0.635	85.8
YOLOv6l	0.929	0.820	59.60	0.929	0.650	150.7
YOLOv5n	0.912	0.868	1.77	0.920	0.597	4.2
YOLOv5s	0.922	0.896	7.03	0.933	0.631	15.8
YOLOv5m	0.921	0.907	20.88	0.942	0.652	47.9
YOLOv5l	0.925	0.923	46.5	0.947	0.66	107.7
R_YOLO	**0.934**	**0.927**	17.5	**0.958**	**0.686**	56.8

4.4 Different Deep Learning Detection Algorithms

Compared with other algorithms such as CAA-YOLO and YOLO NAS, the R_YOLO algorithm shows better performance in terms of mAP (0.5), FPS, and model size. As depicted in Fig. 8, the x-axis represents the FPS of the model, the y-axis represents mAP (0.5), and the size of the circle in the figure corresponds to the number of model parameters. It can be seen that the comprehensive performance of the R_YOLO algorithm is the best, followed by the YOL_NAS algorithm.

As indicated in Table 5, R_YOLO achieved the highest mAP (0.5). Both Faster R-CNN and RetinaNet employ Resnet 50 as the underlying feature extraction layer. However, the feature maps generated by Resnet 50 are single-layered and possess a relatively low resolution. Consequently, these feature maps are unable to adequately capture the intricate details of small objects, leading to a significant number of missed detections. Compared to CAA_YOLO, R_YOLO demonstrates a significant reduction in model complexity, with a remarkable decrease of 82.2% in the number of parameters. Additionally, R_YOLO exhibits an improvement of 1.7% in mAP (0.5) and a notable enhancement in model recognition speed, surpassing CAA_YOLO by more than three times and EfficientDet-D3 by more than nine times. The present paper introduces the R_YOLO model as a solution to address the challenges associated with the large number of parameters in CAA_YOLO, slow detection speed, and high cost of mobile deployment.

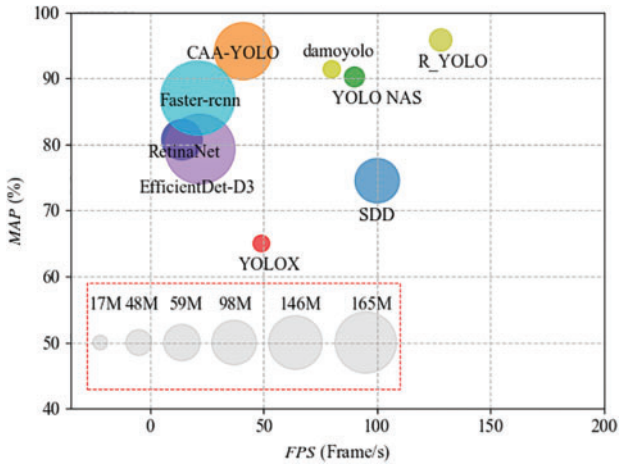


Figure 8: Comparison of different target detection algorithms

Table 5: Different target detection algorithms

Model	Framework	mAP (0.5)	Parameters (M)	FPS
Faster-RCNN	ResNet50+FPN	87.0	165	21
RetinaNet	ResNet50+FPN	79.3	146	22
EfficientDet-D3	EfficientNet+BiFPN	80.7	48.5	14
SDD	ResNet50+FPN	74.5	59	100
DAMOYOLO	damoyolo_tinynasL20_T_436.pth	92.4	8.19	80
YOLO X	YOLOX_S	65	8.94	49
YOLO NAS	YOLO_NAS_S	90.2	11.9	83
CAA_YOLO	CAA_YOLO	94.1	98.5	41
R_YOLO	R_YOLO	**95.8**	17.5	**128**

To further substantiate the efficacy of the enhanced algorithm, several images have been chosen from the Infrared Ship dataset to visualize heat maps. Fig. 9a represents the original dataset, and Figs. 9b and 9c respectively show the heatmap visualization results of the YOLOv8s and the R_YOLO algorithm used for ship detection. The figure illustrates that the YOLOv8s algorithm network does not effectively capture the features of ships. In contrast, the R_YOLO algorithm demonstrates the ability to effectively suppress the impact of background information on object detection, thereby enhancing the network's focus on the distinctive characteristics of various ship types.

4.5 Different Datasets

In the present study, two ship datasets have been selected to evaluate the proposed model's dependability and efficacy. The primary distinction lies in the contrasting environmental conditions, with one being situated in the visible light spectrum and the other operating within the infrared range. As depicted in Figs. 10 and 11, the comparison primarily focuses on the mAP (0.5) and mAP (0.5:0.95). The red curve illustrates the R_YOLO algorithm, while the blue curve corresponds to the YOLOv8s algorithm. The R_YOLO algorithm demonstrates superior performance compared to the

basic YOLOv8 algorithm in terms of mAP (0.5) and mAP (0.5:0.95). Among them, on the visible light dataset, the average increase is 4.5 %.

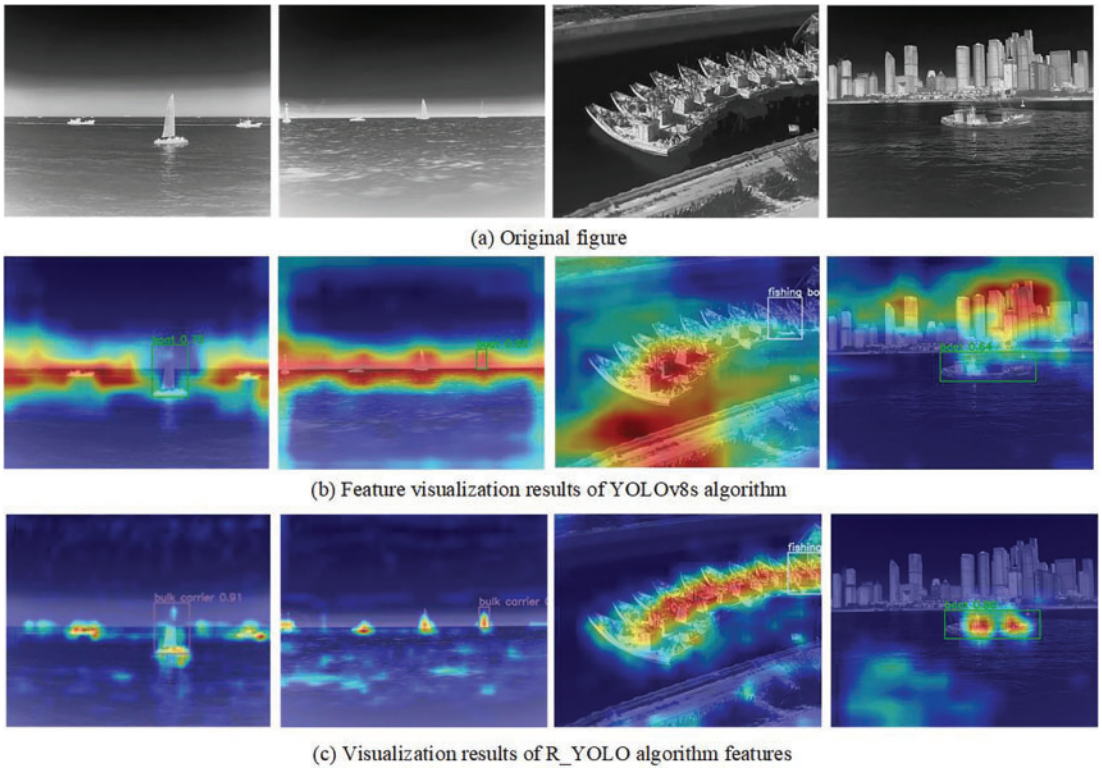


Figure 9: Comparison of infrared ship feature visualization results

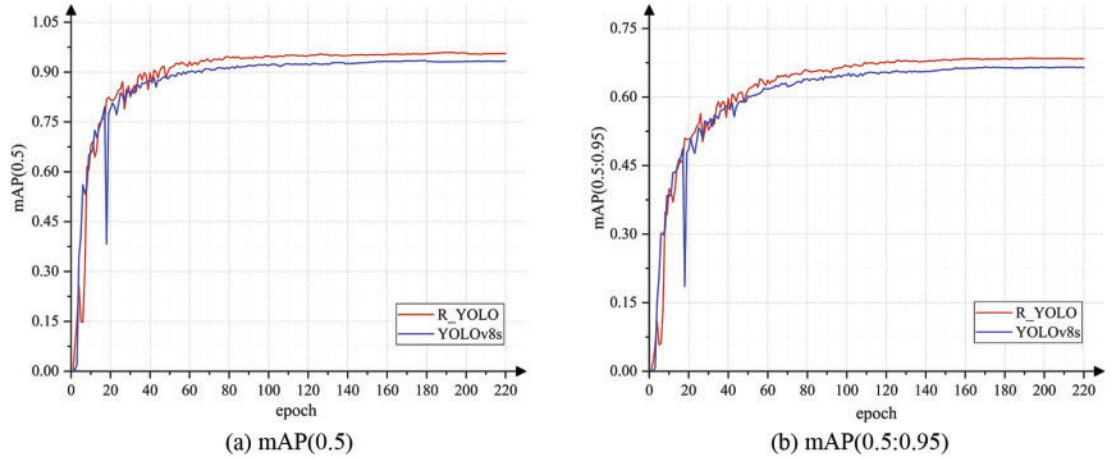


Figure 10: Infrared ship dataset

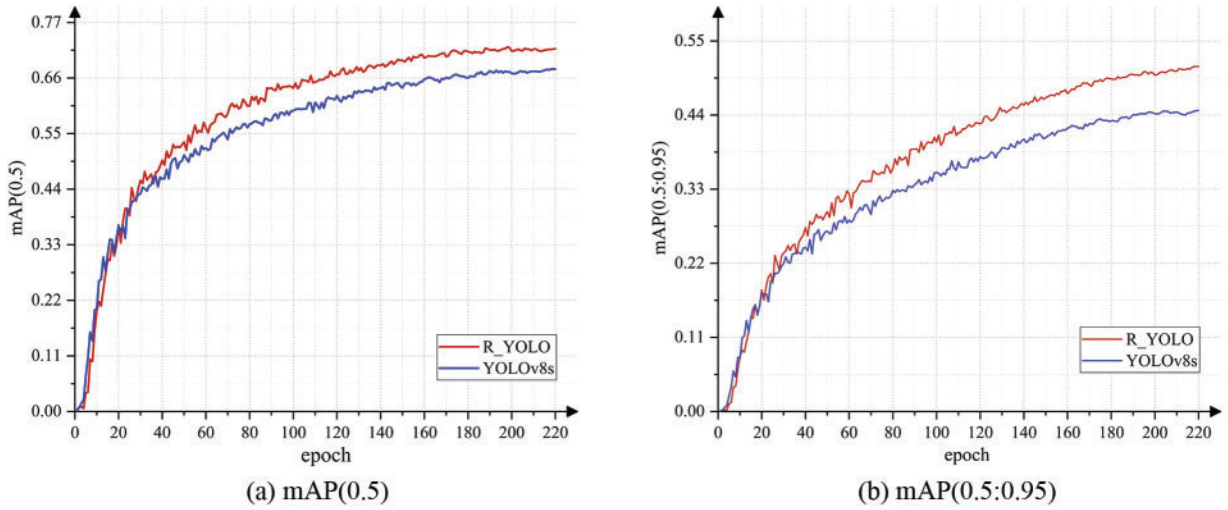


Figure 11: Ship dataset

5 Conclusion

In this paper, we present a novel approach for the identification of unmanned ships utilizing the R_YOLO algorithm. The proposed method integrates the CSPStaget and EMA attention modules, introduces a small target detection head and CABlock module, and incorporates an occlusion loss function. Based on the aforementioned experimental findings, the R_YOLO algorithm, as proposed in this study, has the potential to substantially enhance both the precision of model recognition and the speed of model inference. Compared to the CAA_YOLO algorithm, the proposed method achieves a 1.2% increase in mAP (0.5) while reducing the number of parameters by 82%. This trade-off effectively balances target detection accuracy and the need for faster reasoning speed, achieving a significant milestone in developing unmanned ship intelligence. However, it is important to acknowledge that this method does have certain limitations. In the future, we must address the following issues: (1) Considering the economic implications of model deployment, it is necessary to further compress the model. For instance, the implementation of pruning techniques [2] can be employed as illustrative examples. (2) Enhancing the recognition rate of occluded targets is crucial to achieve improved results. One effective approach is to incorporate an occlusion attention module [24].

Acknowledgement: Thanks to the infrared dataset provided by Shandong Yantai Arrow Optoelectronics Technology Co., Ltd.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Study conception and design: Chun-Ming Wu, Jin Lei; data collection, analysis, and interpretation of results: Ling-Mei Ren, Wu-Kai Liu; draft manuscript preparation: Ling-Li Ran. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data and materials used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] D. D. Ma, L. L. Dong, R. L. Gao and W. H. Xu, "Recent advancements in long-distance marine infrared target detection: Latest method and future perspectives," *Infrared Physics & Technology*, vol. 133, pp. 104729, 2023.

[2] C. M. Wu, Y. Q. Sun, T. J. Wang and Y. L. Liu, "Underwater trash detection algorithm based on improved YOLOv5s," *Journal of Real-Time Image Processing*, vol. 19, no. 5, pp. 911–920, 2022.

[3] S. Q. Ren, K. M. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[4] P. Y. Jiang, E. G. Daji, F. Y. Liu, Y. Cai and B. Ma, "A review of yolo algorithm developments," *Procedia Computer Science*, vol. 199, pp. 1066–1073, 2022.

[5] W. Liu, D. Anguelov, E. Dumitru, C. Szegedy, R. Scott *et al.,* "SSD: Single shot multibox detector," in *Proc. Computer Vision-ECCV 2016: 14th European Conf.*, Amsterdam, The Netherlands, pp. 21–37, 2016.

[6] X. Chen, W. Tan, Q. Wu, F. Zhang, X. Guo *et al.,* "Contamination identification of lentinula edodes logs based on improved YOLOv5s," *Intelligent Automation & Soft Computing*, vol. 37, no. 3, pp. 3143–3157, 2023.

[7] V. Srivastava, S. Mishra and N. Gupta, "Automatic detection and categorization of road traffic signs using a knowledge-assisted method," *Procedia Computer Science*, vol. 218, pp. 1280–1287, 2023.

[8] S. Z. Yang, W. Wang, G. Sheng and Z. P. Deng, "Strawberry ripeness detection based on YOLOv8 algorithm fused with LW-Swin transformer," *Computers and Electronics in Agriculture*, vol. 215, pp. 108360, 2023.

[9] Z. Chen, C. Huang, L. Duan and B. Tan, "Lightweight surface litter detection algorithm based on improved yolov5s," *Computers, Materials & Continua*, vol. 76, no. 1, pp. 1085–1102, 2023.

[10] X. Tang, C. Wang, J. Su and C. Taylor, "An elevator button recognition method combining YOLOv5 and OCR," *Computers, Materials & Continua*, vol. 75, no. 1, pp. 117–131, 2023.

[11] G. Han, M. C. Zhu and X. C. Zhao, "Method based on the cross-layer attention mechanism and multiscale perception for safety helmet-wearing detection," *Computers and Electrical Engineering*, vol. 95, pp. 107458, 2021.

[12] Z. Liu and H. Hu, "Swin Transformer V2: Scaling up capacity and resolution," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, New Orleans, LA, USA, pp. 11999–12009, 2022.

[13] L. S. Xu, S. H. Dong, H. T. Wei and J. W. Huang, "Defect signal intelligent recognition of weld radiographs based on YOLO V5-IMPROVEMENT," *Journal of Manufacturing Processes*, vol. 99, pp. 373–381, 2023.

[14] S. X. Cheng, Y. Zhu and S. Wu, "Deep learning based efficient ship detection from drone-captured images for maritime surveillance," *Ocean Engineering*, vol. 285, pp. 115440, 2023.

[15] K. Y. Feng, L. Lun, X. Wang and X. Cui, "LRTransDet: A real-time SAR ship-detection network with lightweight ViT and multi-scale feature fusion," *Remote Sensing*, vol. 15, pp. 5309, 2023.

[16] Y. M. Zhang, "A lightweight multi-target detection method for infrared remote sensing image ships," *Journal of Network Intelligence*, vol. 8, pp. 534–544, 2023.

[17] Z. Chen and C. Liu, "Multi-scale ship detection algorithm based on YOLOv7 for complex scene SAR images," *Remote Sensing*, vol. 15, no. 8, pp. 2071, 2023.

[18] Y. Kong, Y. Zhang and X. Peng, "Few-shot high-resolution range profile ship target recognition based on task-specific meta-learning with mixed training and meta embedding," *Remote Sensing*, vol. 15, pp. 5301, 2023.

[19] H. N. Guo and R. Long, "A marine small-targets classification algorithm based on improved convolutional neural networks," *Remote Sensing*, vol. 15, no. 11, pp. 2917, 2023.

[20] J. Ye, Z. Y. Yuan, C. Qian and X. Q. Li, "CAA-YOLO: Combined-attention-augmented YOLO for infrared ocean ships detection," *Sensors*, vol. 22, no. 10, pp. 3782, 2022.

[21] B. T. Jiang, X. F. Ma, Y. Lu and Y. Li, "Ship detection in spaceborne infrared images based on convolutional neural networks and synthetic targets," *Infrared Physics & Technology*, vol. 97, pp. 229–234, 2018.
[22] O. Y. Da, H. Su and G. Z. Zhang, "Efficient multi-scale attention module with cross-spatial learning," in *Proc. ICASSP 2023–2023 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Rhodes Island, The Greece, pp. 1–5, 2023.
[23] Z. P. Yu, "YOLO-FaceV2: A scale and occlusion aware face detector," arXiv preprint arXiv:2208.02019, 2022.
[24] X. Z. Xu, Y. Q. Jiang, W. H. Chen, Y. L. Huang and Y. Zhang, "DAMO-YOLO: A report on real-time object detection design," arXiv preprint arXiv:2211.15444, 2022.
[25] C. Y. Wang and H. Y. Liao, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops*, Seattle, WA, USA, pp. 390–391, 2020.
[26] Y. Liu, H. F. Li, C. Hu, S. Luo, Y. Luo *et al.,* "Learning to aggregate multi-scale context for instance segmentation in remote sensing images," arXiv preprint arXiv:2111.11057, 2021.
[27] C. Y. Li, L. L. Li, H. L. Jiang, K. H. Weng and Y. F. Geng, "YOLOv6: A single-stage object detection framework for industrial applications," arXiv preprint arXiv:2209.02976, 2022.
[28] C. Y. Wang, A. Bochkovskiy and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," arXiv preprint arXiv:2207.02696, 2022.
[29] T. Y. Lin, G. Y. Priya, R. Girshick, K. M. He, D. Piotr *et al.,* "Focal loss for dense object detection," in *Proc. of the IEEE Int. Conf. on Computer Vision*, Venice, Italy, pp. 2999–3007, 2017.
[30] M. X. Tan and R. M. Pang, "Efficientdet: Scalable and efficient object detection," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 10778–10787, 2020.
[31] G. Zheng, S. T. Liu and F. Wang, "Yolox: Exceeding yolo series in 2021," arXiv preprint arXiv:2107.08430, 2021.