Ocean Engineering 332 (2025) 121447

Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng

Research paper

Visual perception for long-distance and small target detection in autonomous maritime navigation

Ruolan Zhang [a,b], Xingchen Ji [b], Sean Loughney [a,*], Jin Wang [a], Zaili Yang [a]

[a] Navigation College of Dalian Maritime University, Dalian, China
[b] Liverpool Logistics, Offshore and Marine Research Institute, James Parsons Building, Byrom Street, Liverpool, L3 3AF, UK

ARTICLE INFO

Keydords:
Object detection
Visual navigation aid
Small target
Computer version
Autonomous ships

ABSTRACT

In the pursuit of advancing autonomous maritime navigation, this study aimed to develop a novel architecture designed to enhance the detection accuracy of distant and small targets under the constraints of real-time performance and robustness. Through the innovative integration of the Convolutional Block Attention Module (CBAM) into the detection model's backbone, the study achieved superior feature extraction capabilities tailored for the complexities of maritime environments. Further optimization of the Spatial Pyramid Pooling (SPP) module ensured model compactness and computational efficiency, vital for deployment on edge devices. A key methodological novelty lay in the incorporation of the S-IoU loss function, which offers superior bounding box regression accuracy over the traditional Generalized Intersection over Union, directly contributing to more precise navigation and effective obstacle avoidance. The proposed enhancements collectively yielded a 5.1 % increase in mAP@50 %, accompanied by an 11.2 % reduction in model parameters and a 12.6 % decrease in computational complexity (GFLOPs). These findings underscore the potential of the presented architecture to significantly contribute to maritime safety, presenting an optimized solution for collision avoidance and navigation assistance in congested sea routes and adverse weather conditions.

1. Introduction

The relentless progression in autonomous ship navigation technologies increasingly depends on computer vision and augmented visual perception capabilities. This reliance is evidenced by the critical need for small target detection mechanisms, essential for navigating complex maritime environments (Tang et al., 2022). Such systems are vital for ships to proficiently identify and respond to numerous, distant objects, such as small boats, buoys, and floating debris, particularly in narrow channels or congested ports. These detection systems must maintain high precision to ensure vessels adhere to their designated routes and comply with navigational protocols, crucial for collision avoidance and the safety of vessels and their crews (Ghazali et al., 2024).

Unlike traditional navigational tools such as AIS (Automatic Identification Systems) and radar, which primarily detect large, registered vessels and provide limited data on small or non-traditional targets, vision-based methods can capture a broader range of objects (Zhao et al., 2024). These include small boats, buoys, and floating debris, crucial in narrow channels or congested ports where high precision is vital for safety and adherence to navigational protocols.

Vision-based detection systems are particularly advantageous for bridge navigation as they offer superior dynamic range and the ability to detect both metallic and non-metallic objects, including those that are not typically covered by radar or AIS (AliAkbarpour et al., 2024). This capability is critical in congested or complex environments where traditional sensors might fail to provide sufficient resolution or discernment of small targets. Moreover, vision-based systems can be more cost-effective, requiring lower maintenance and installation costs compared to radar systems, which are generally more complex and expensive to operate and maintain.

Advancing autonomous maritime navigation requires detecting small, distant targets like nearby vessels, light spots, buoys, and navigational obstacles (Zhang et al., 2024). These are critical for safety and present significant challenges when viewed from a ship's bridge. Fig. 1 shows the characteristics of small maritime targets. These targets are difficult to detect because they appear small at a distance, blend into complex maritime backgrounds, and are affected by environmental conditions like fog and waves. Additionally, their dynamic nature, influenced by wind and currents, requires robust, real-time detection methods. This study develops a novel architecture aimed at enhancing

* Corresponding author.
E-mail address: s.loughney@ljmu.ac.uk (S. Loughney).

https://doi.org/10.1016/j.oceaneng.2025.121447
Received 18 December 2024; Received in revised form 21 March 2025; Accepted 1 May 2025
Available online 11 May 2025

the accuracy of detecting these crucial yet challenging targets, integrating advanced computational techniques to improve both navigational safety and operational efficiency in autonomous maritime systems.

Furthermore, the term "small objects" in maritime contexts has two primary definitions: physically small objects in the real world and those classified by image size standards, such as the MS-COCO threshold of 32 × 32 pixels, known as 'small objects' (Lin et al., 2014). However, maritime detection standards often require prioritizing objects critical for navigational safety over less critical items like floating debris (Shen et al., 2024). The need for robust detection algorithms becomes more pronounced under challenging weather conditions where small targets are harder to detect (Zhang et al., 2022a).

As maritime navigation technologies evolve, there is a growing focus on developing efficient detection methods. Techniques such as the Sliding Window and Image Pyramid have been foundational, although each comes with significant computational demands. Recent advancements aim to streamline these methods to better accommodate the computational limits of modern technology while enhancing the efficiency and accuracy of small object detection systems (Hirzel et al., 2017; Qin et al., 2021; Pang et al., 2019).

Recent advancements in computing capabilities and algorithms have driven a significant shift from traditional object detection methods like Sliding Windows and Image Pyramids to deep learning-based techniques. Traditional methods were known for their high computational demands, especially when processing large images, which slowed processing speeds and made real-time applications challenging (Kheradmandi and Mehranfar, 2022). In contrast, deep learning approaches such as Faster R-CNN (Girshick, 2015), SSD (Single Shot Multibox Detector) (Liu et al., 2016), Transformers (Vaswani et al., 2017), and YOLO (You Only Look Once) (Redmon et al., 2016) have substantially enhanced the efficiency and accuracy of small object detection. These methods demonstrate superior performance in handling diverse and complex detection tasks in maritime environments.

Faster R-CNN and SSD are particularly noted for their accuracy and real-time detection capabilities, suitable for dynamic maritime operations that require rapid responses. The Transformer model, with its ability to capture comprehensive contextual information, is well-suited for large-scale maritime surveillance. In contrast, YOLO excels in speed by detecting objects in a single forward pass across the entire image, which streamlines the detection process compared to the multi-step operations of its counterparts.

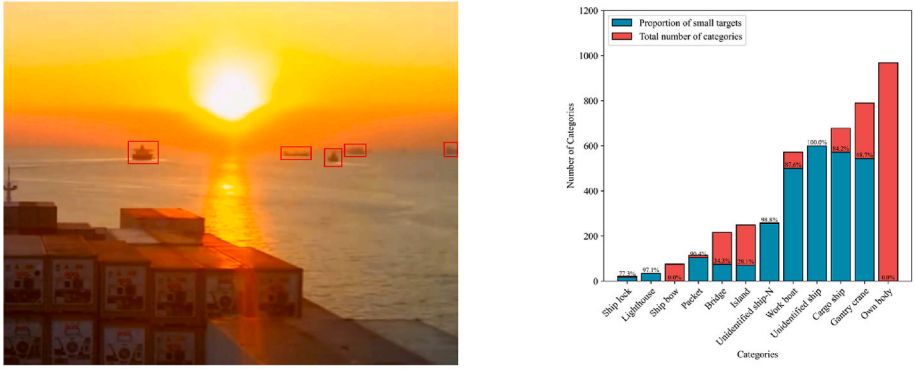
Despite their advantages, deep learning methods face limitations, primarily their substantial computational resource demands. Complex models like Transformers require extensive datasets and considerable computational power, which can limit their deployment on devices with constrained resources (Han et al., 2022). Additionally, models like SSD sometimes struggle to detect very small targets due to limited resolution, which can hinder their effectiveness in accurately identifying such objects.

These insights highlight the need for ongoing enhancements in deep learning techniques to balance model complexity, computational efficiency, and detection capabilities in practical applications. This balance is crucial for advancing maritime navigation and surveillance systems, making them more robust and responsive to operational demands.

The integration of computer vision technologies, particularly Convolutional Neural Networks, has markedly enhanced the precision and robustness of small target detection in diverse domains, including maritime environments (Xu et al., 2023). Innovations such as those by Saleh et al. (2022), who analyzed deep learning methods for underwater small habitat fish video analysis, demonstrate the shift from traditional manual monitoring to more advanced, automated systems that provide crucial ecological insights. Similarly, Zhao et al. (2024) advanced object detection using unmanned aerial vehicles (UAVs), reviewing over 200 studies to illustrate significant progress in field perception and small target detection, which are vital for maritime surveillance and research.

In the realm of small target detection, Akyon et al. (2022) provided SAHI (Slicing Aided Hyper Inference), which enhances image processing by implementing a strategy of slicing large images and utilizing overlapping slices, this model is adept at merging detection results post-processing, making it particularly effective in environments populated with densely arranged small targets. Xu et al. (2022) presented DAMO-YOLO, which stands out with its efficient backbone network design complemented by MAE self-supervised pre-training, which together foster leading-edge detection performance adaptable to various target sizes. Furthermore, the RT-DETR (Real-Time Detection Transformer) framework introduces an end-to-end detection architecture that incorporates IoU-aware query selection and efficient feature interaction, achieving a balance between real-time performance and accuracy (Zhao et al., 2024).

Despite the general progress, specific adaptations for ship navigation are less common. Most existing research focuses on broad methodologies like multi-scale feature fusion and adaptability to changing environmental conditions (Zhang et al., 2023, Chen, Shin), often neglecting the unique aspects of maritime navigation. However, Moosbauer et al. (2019) addressed these specific needs using the Singapore Maritime Dataset to improve object segmentation for maritime settings, enhancing the training of models like the weakly supervised recursive Mask R-CNN.



(a) (b)

Fig. 1. (a) illustrates small targets as viewed from a ship's bridge, highlighting that, unlike small targets in public datasets, maritime small targets frequently exhibit varying lengths and widths. In figure (b), the red bars denote the total number of items per category within our ShipNav dataset, while the blue bars signify the count of small targets within each category of the ShipNav dataset. The percentages reflect the ratio of small targets relative to their respective categories. In the categories of "Ship bow" and "Own body," there are no small targets.

Further expanding on these advancements, Kiefer et al. (2023) introduced the SeaDronesSee Object Detection v2 benchmark, incorporating a broader range of categories and camera perspectives, significantly enriching the framework for evaluating maritime computer vision algorithms. This benchmark and other developments highlight the ongoing evolution of computer vision in the maritime sector, driven by the sector's unique requirements and the potential for significant advancements in ship navigation and maritime surveillance.

This paper presents a groundbreaking method for small target detection, and it introduces a refined version of a one-stage object detection framework, enhanced with the CBAM for superior feature extraction and an optimized SPP module to boost efficiency. Additionally, this approach incorporates the innovative S-IoU loss function, designed to achieve unparalleled precision in target detection. These modifications were specifically crafted to meet the distinctive demands of maritime navigation, thereby establishing a new standard in the field. Through rigorous experimental evaluation, our optimized one-stage object detection model demonstrated exceptional proficiency in overcoming the intricacies associated with detecting small targets on water. The model's effectiveness was attributed to three principal new contributions.

a) Integration of SimSPPF and GhostConv Modules: This combination effectively retained information across channels while simultaneously reducing computational demands, ensuring efficient and comprehensive feature analysis.
b) Fusion of the CMAB Attention Mechanism: By integrating the CMAB attention mechanism, the new framework not only upheld the precision and velocity of small target detection but also diminished the model's overall parameter count. This balance between efficiency and effectiveness was critical for real-time maritime navigation applications.
c) Adoption of the S-IoU Loss Function: The introduction of the S-IoU loss function significantly refined target recognition capabilities. This enhancement boosted the regression accuracy of bounding boxes, enabling the precise identification of small target types and substantially improving overall detection performance.

This paper is organized as follows. Detecting small targets in maritime navigation is not only a challenge for assisted and autonomous driving, which is why we conduct this research. Section 2 provides an overview of related work. In Section 3, the algorithms were developed. They began by analyzing the principles of the visual object detection algorithm YOLOv5 and were then followed by the strategies for improving the YOLOv5 algorithm. Section 4 discussed the experimental results, and finally, Section 5 summarized the findings and outlined future work.

2. Related works

2.1. Datasets

Over the past twenty years, the development of numerous datasets has significantly advanced multi-object and specific object detection tasks. Well-known datasets such as ImageNet (Krizhevsky et al., 2012), PASCAL VOC (Everingham et al., 2010), and COCO have played pivotal roles in the recognition and detection of multiple static objects. However, the application of these models in real-world navigation scenarios encountered substantial challenges. While many researchers aimed to enhance object detection accuracy using these publicly available datasets, it was observed that existing methods predominantly recognized only broad categories, such as ships, which did not suffice for the nuanced requirements of practical navigation systems.

Furthermore, the complexity of maritime environments necessitated different perspectives for effective monitoring, including shore-based and vessel-based views, each fulfilling unique requirements. Shao et al. (2018) introduced SeaShips, a comprehensive dataset for ship type detection in the Yangtze River basin, limited to six primary ship categories. X Y Zhou et al., 2021) presented the Water Surface Object Detection Dataset (WSODD), a benchmark for identifying a variety of water surface objects. While these datasets were beneficial, they highlighted the challenges in real-time maritime navigation monitoring.

Nevertheless, there existed a notable gap in datasets from the perspective of the ship bridge, an essential viewpoint for the visual perception of long-distance small targets in maritime navigation. This absence significantly hampered the task of visual perception, especially in detecting small targets over long distances, emphasizing the need for specialized datasets tailored to the unique demands of maritime navigation.

To address this shortfall, this work introduces the "ShipNav" dataset, designed specifically for evaluating the performance of vision-based scenarios and tasks in ship navigation. As shown in Fig. 2, the ShipNav dataset gathered from the ship bridge perspective, encompasses twelve ship bridge acquisition classes. It accounted for various weather conditions and rich shore-based backgrounds, incorporating an extensive collection of video data from major global shipping routes. Moreover, a high proportion of small targets was deliberately included to reflect the challenges encountered in maritime visual perception. The dataset categorized objects encountered during practical ship navigation at sea into two types: navigation-aid objects and obstacle objects. Navigation-aid objects included "Work boat," "Cargo ship," "Own body," "Unidentified ship," "Unidentified ship-N," "Packet," "Ship bow," and "Island." Meanwhile, obstacle objects comprised "Gantry crane," "Lighthouse," "Bridge," and "Ship lock." This classification was designed to closely mimic the visual perception tasks faced by maritime navigators, offering a tool for enhancing object detection algorithms in the context of ship navigation.

2.2. One-stage object detection

The evolution of computer vision over the last decade was significantly tied to the expansion of datasets and the adoption of Convolutional Neural Networks. These technologies underpinned breakthroughs in various domains, including face recognition (Adjabi et al., 2020), object detection (Zou et al., 2023), robot vision (Qiao et al., 2021), and autonomous driving (Chen et al., 2022). Deep Learning, a cornerstone of Artificial Intelligence (AI), diversified network architectures, enabling the automatic and efficient extraction of features from expansive datasets.

Among the real-time object detection frameworks, the YOLO series stood out for its efficiency and accuracy. Starting with YOLOv1 (Redmon et al., 2016), which introduced an end-to-end real-time detection approach by converting the detection task into a regression problem, the series evolved significantly. YOLOv2 (Redmon and Farhadi, 2017) enhanced multi-class detection, YOLOv3 (Redmon and Farhadi, 2018) improved small object detection with multi-scale feature maps, and YOLOv4 (Bochkovskiy et al., 2020) introduced advanced training techniques for better generalization. YOLOv5 (Redmon et al., 2016), known for its speed and efficiency, opted for a lightweight design and the PyTorch framework for increased usability. YOLOv6 (Li et al., 2022) and YOLOv7 (Wang et al., 2023) introduced innovations in autonomous delivery and pose estimation, respectively. YOLOv8 (G Jocher et al., 2023) further expanded the series' capabilities to encompass a wider array of computer vision tasks.

For the model used in this paper, YOLOv5 was selected as the baseline model because of its balanced characteristics, including a low parameter count, lightweight design, and the requirement for real-time processing in complex maritime scenarios. Despite the existence of YOLOv8, the extensive documentation and demonstrated effectiveness of YOLOv5 across various datasets made it the preferred choice for this research.

Currently, more and more researchers have become increasingly

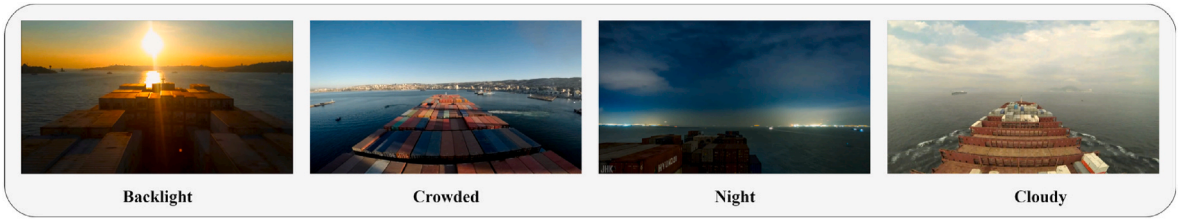


Fig. 2. Illustration of the ShipNav dataset, featuring various weather scenarios and complex maritime environments.

interested in enhancing the small object detection accuracy using YOLOv5. Innovations such as TPH-YOLOv5 (Zhu et al., 2021), which incorporated Transformer Prediction Heads for improved scale detection, and YOLO-Z (Benjumea et al., 2021), which focused on optimizing feature map handling through various Feature Pyramid Networks, exemplified the ongoing efforts to refine YOLOv5. Additionally, the integration of SPD-Conv into YOLOv5 by Raja Sunkara et al. (Sunkara and Luo, 2022) demonstrated the potential for significant advancements in detecting low-resolution images and small objects, further illustrating the dynamic research landscape surrounding one-stage object detection. YOLOv11 (Jocher et al., 2023) is the latest edition in the Ultralytics YOLO series, incorporating new features and improvements to further enhance performance and flexibility. Taking various factors into consideration, we have chosen YOLOv5m as our baseline model.

3. Methodology

In the proposed multi-scale object detection architecture designed specifically for maritime object detection, the methodology unfolded as follows: Initially, an original image served as the input, with the aim of outputting detected objects, each delineated by bounding boxes, and accompanied by class labels and confidences. The process began with a deep convolutional network enhanced with Cross-Stage Partial (CSP) structures for efficient feature extraction, generating a rich set of feature maps from the input image. Subsequently, these feature maps underwent a transformation through a module designed to refine feature representation while minimizing computational costs.

To further enrich the feature fusion process, a combination of the CBAM and Ghost Convolution techniques was utilized. This approach emphasized crucial features and ensured computational efficiency, significantly enhancing the feature set. This enriched feature set then fed into the final detection module, which utilized an advanced loss function designed to improve the accuracy of bounding box predictions, class identifications, and confidence assessments.

3.1. The model of GSimSPPF (simplified SPPF with GhostConv)

In this section, the "GSimSPPF" model is divided into four subsections, with the aim of reducing parameter count and improving small object detection accuracy. The improved principles of the spatial pyramid pooling module are elaborated upon in detail.

3.1.1. The model of spatial pyramid pooling (SPPF)

The SPP module was proposed primarily to address the issue of convolutional neural networks handling images of different scales. It was widely used in computer vision tasks such as object detection and image classification. Moreover, the introduction of the SPPF module aimed to further optimize the performance in handling small targets. It introduced a feature focusing mechanism, which enabled more effective attention and utilization of important regions in the feature map, thereby improving the accuracy and performance of small target detection. The SPPF module in YOLO series algorithms enabled multi-scale feature fusion and receptive field enhancement, and received feature maps of different sizes from three MaxPool layers, with kernel sizes of 5, 9, and 13. Then they were output at a fixed size, enabling the network to train at multiple scales. Due to the similarity in appearance of small objects and the variability of the navigational environment, there were false detections in small object detection. In addition, it also suffered from drawbacks such as computational complexity, limited receptive field range, and information redundancy. The structure diagram of the SPP and SPPF module is shown in Fig. 3.

The introduction to the SPPF module is as follows and the mathematical formula for the Conv layer is:

$$Conv(x) = f(Norm(Conv2d(x))) \tag{1}$$

The $f(\cdot)$ is the activation function, Norm is batch normalization, and Conv2d is the convolution operation.

The mathematical formula for the SPPF layer is:

$$SPPF(x) = cv2([cv1(x), MP(x, k_s), MP(MP(x, k_s), k_s), MP(x, k_s)]) \tag{2}$$

where MP is the max pooling operation, k_s represents the kernel size of the max pooling kernel.

3.1.2. The model of similarity-based Spatial Pyramid Pooling Fusion (SimSPPF)

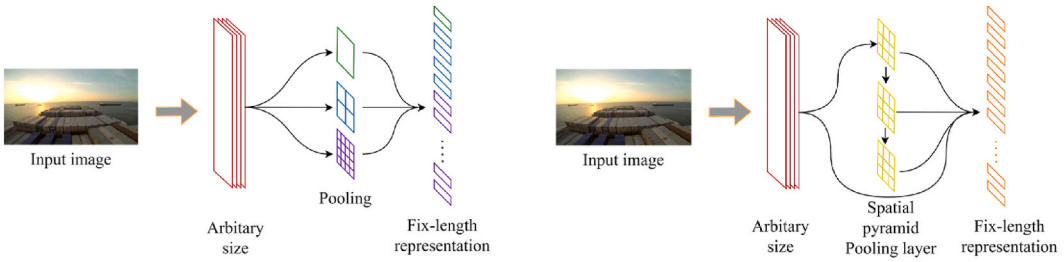
To address the issues of large parameter count and high complexity in the area, a new module called GSimSPPF fusion is proposed. The SimSPPF (Liu et al., 2016) module aims to enhance feature extraction capabilities and address challenges in computer vision models. The pooling kernels of size 9 and 13 in the SimSPPF module are represented by kernels of size 5, significantly reducing computational costs. Firstly, SimSPPF achieves efficient feature pooling and fusion without compromising performance by adopting a similarity-based pooling approach to reduce computational complexity. Secondly, it enhances feature representation by considering the similarity between different spatial pyramid levels. It takes into account relationships and similarities between different pyramid levels to capture more discriminative and context-aware features. Finally, SimSPPF merges contextual information by leveraging the similarity between different spatial pyramid levels. This enables the model to capture global context and context dependencies, thereby improving object recognition and localization.

3.1.3. More features from cheap operations with GhostNet

GhostNet (Han et al., 2020) is a novel neural network architecture proposed by Huawei Noah's Ark Lab. Similar to Google's MobileNet, GhostNet is designed for lightweight and compact networks, particularly for hardware and mobile applications, but it outperforms MobileNet. GhostNet is based on the Ghost module, which has the unique feature of not altering the size and channel dimensions of the convolutional output feature map. However, it significantly reduces the overall computational load and the number of parameters. In simple terms, GhostNet's main contribution lies in reducing computational load, improving runtime speed, and minimizing the decrease in accuracy. Moreover, this modification is applicable to any convolutional network, as it does not change the size of the output feature map. The structure diagram of the GhostNet module is shown in Fig. 4.

3.1.4. Spatial pyramid pooling module base on GSimSPPF

In summary, the SimSPPF module based on GhostNet addresses the



(a) The architecture of Spatial Pyramid Pooling (SPP) module. (b) The architecture of Spatial Pyramid Pooling Network (SPPF).

Fig. 3. SPPF module has been optimized compared to the traditional SPP module in terms of feature focusing, dynamic feature resampling, and global information integration, thus achieving better performance in computer vision tasks such as object detection and image classification.

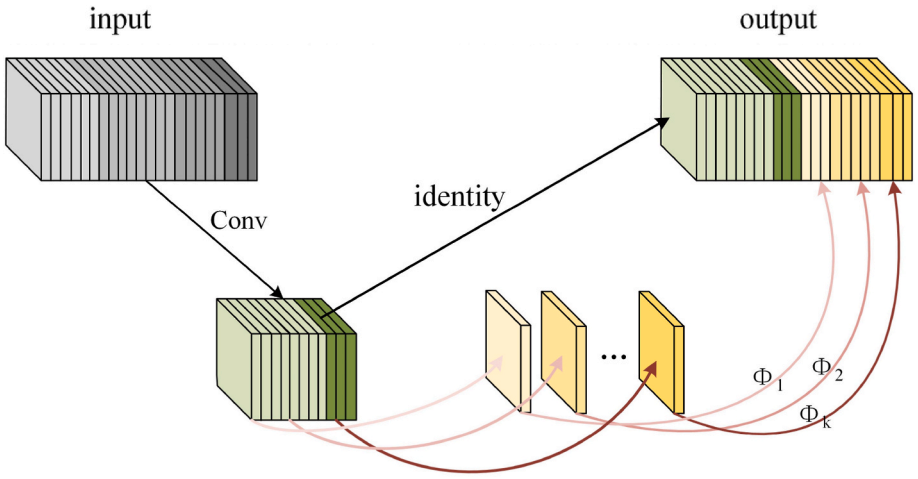


Fig. 4. The model structure diagram of GhostConv.

limitations of the SPPF module in terms of computational complexity, receptive field range, and information redundancy. The model diagram is shown in Fig. 5. It achieves these improvements by reducing computational complexity, enhancing feature representation, and merging contextual information based on the similarity between different spatial pyramid levels. This ultimately reduces the probability of misidentifying small object types and decreases the number of parameters and FLOPs introduced by adding convolutional layers.

3.2. Feature extraction module based on CBAM with GhostConv moudles (FECG)

In computer vision, feature fusion can achieve favorable complementarity among multiple features, resulting in more robust and accurate recognition results. The Neck section of baseline model adopts the Feature Pyramid Network with Path Aggregation Network (PAN) structure to achieve multi-scale feature fusion. The FPN structure propagates strong semantic features from top to bottom, while the PAN structure propagates strong positional features from bottom to top and aggregates parameters from different detection layers in different backbone layers. However, the structure uses the same CBS module as the backbone. While it can well maintain the model's feature extraction capability, it also leads to a higher number of model parameters and computational complexity. Therefore, this paper integrates GhostNet and proposes the FECG structure to address these issues.

If the requirement is to generate n feature maps, where the input

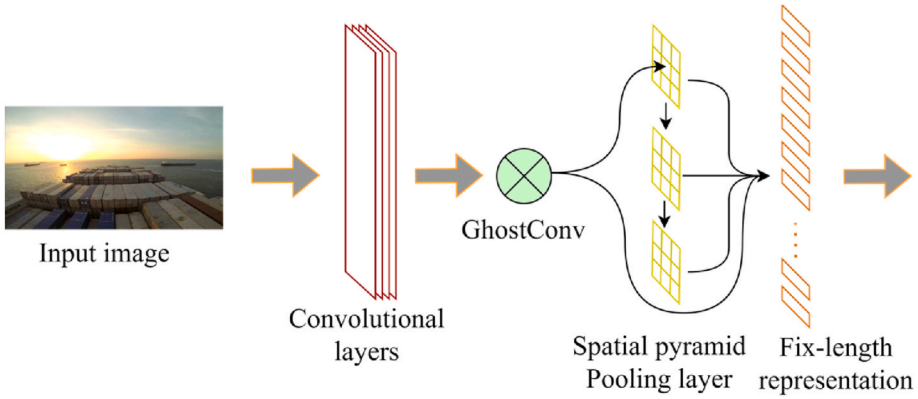


Fig. 5. The model structure diagram of GSimSPPF.

image size is $h \times w \times c$ represents the image height, w represents the image width, and c represents the input channels). In this equation, FLOPs represents the number of floating-point operations required for a convolutional layer. Here, n denotes the number of input feature maps, h' and w' represent the height and width of the output feature maps respectively, c is the number of channels in each input feature map, and $k \times k$ denotes the kernel size of the convolutional filter. Therefore, the formula calculates the total number of floating-point operations by multiplying these factors together, the FLOPs (floating-point operations) for regular convolution are as follows:

$$FLOPS = n \times h' \times w' \times c \times w \times k \tag{3}$$

The work by (Han et al., 2020) on exploration of feature maps revealed notable similarities among them, leading to a strategic bifurcation of the feature maps into two distinct categories. The initial step involves employing GhostConv to produce $m(m \leq n)$ intrinsic feature maps via standard convolutional processes. Subsequently, each of these intrinsic feature maps is subject to a linear transformation (Φ), culminating in the generation of corresponding Ghost feature maps for each intrinsic one. This secondary set of feature maps acts as the 'Ghost' of the initial set, encapsulating the essence of the process as follows:

$$y_{ij} = \Phi_{i,j}\left(y_i'\right), \forall i = 1, \ldots, m, j = 1, \ldots, s, \tag{4}$$

The y_i' represents the i feature map among the m intrinsic feature maps, $\Phi_{i,j}$ represents the j linear transformation of the i intrinsic feature map, which implies that each intrinsic feature map can have multiple Ghost maps. y_{ij} represents the n feature maps. The relationship between the feature maps can be described as follows (Han et al., 2020):

$$n = m \times s \tag{5}$$

In the GhostConv module, there is a fixed identity transformation (as shown in Fig. 4.). Therefore, there are $s-1$ effective linear transformations. Combining with equation (5), obtain the following (Han et al., 2020):

$$m \times (s-1) = \frac{n}{s}(s-1) \tag{6}$$

The number of FLOPs generated by the regular convolution operation in the first part is as follows (Han et al., 2020):

$$\text{FLOPs} = \frac{n}{s} \times h' \times w' \times c \times k \times k \tag{7}$$

In the subsequent segment of the linear operation, when utilizing a kernel size of $d \times d$. It is no longer denoted as $k \times k$ because we're using a different kernel size, denoted as $d \times d$, in this particular context. The notation $k \times k$ typically represents the size of the convolutional filter kernel. However, in this instance, we're referring to a different size denoted by $d \times d$. The corresponding FLOPs incurred can be expressed as follows (Han et al., 2020):

$$\text{FLOPs} = (s-1)\frac{n}{s} \times h' \times w' \times c \times d \times d \tag{8}$$

The total number of FLOPs can be calculated as follows (Han et al., 2020):

$$\text{FLOPs} = \frac{n}{s} \times h' \times w' \times c \times k \times k + (s-1)\frac{n}{s} \times h' \times w' \times c \times d \times d \tag{9}$$

The FLOPs ratio between regular convolution and GhostConv can be expressed as follows (Han et al., 2020):

$$\frac{F_n}{F_s} = \frac{n \times h' \times w' \times c \times k \times k}{\frac{n}{s} \times h' \times w' \times c \times k \times k + (s-1)\frac{n}{s} \times h' \times w' \times d \times d} \tag{10}$$

Where F_n and F_s represent the FLOPs of regular convolution and GhostConv, respectively. In practical applications, to improve the energy efficiency of CPUs or GPUs, the kernel sizes d and k are similar. In this equation, the terms d and k represent different kernel sizes. Specifically: $d \times d$ refers to the kernel size used in the second part of the linear operation. $k \times k$ denotes the kernel size typically used in convolutional layers. The distinction between d and k is crucial because they represent different dimensions of the convolutional filter. In the context of the equation, $d \times d$ is used when discussing a specific part of the operation where a different kernel size is applied, while $k \times k$ is a more general representation of the kernel size conventionally used in convolutional layers. The formula is shown as follows (Han et al., 2020):

$$\frac{F_n}{F_s} = \frac{c \times k \times k}{\frac{1}{s} \times c \times k \times k + \frac{(s-1)}{s} \times d \times d} \approx \frac{s \times c}{s + c - 1} \approx s \tag{11}$$

From the above formula, it can be observed that the FLOPs of the GhostConv module is only $1/s$ of regular convolution, which effectively reduces the FLOPs of the model.

3.2.1. Convolutional Block Attention Module

The C3 module in the baseline model backbone is a critical component for learning features of small objects. It concatenates two feature vectors from the Bottleneck and standard convolutional layer CBS branches. However, the concatenation of these branches in the C3 module fails to capture fine-grained ship image features, such as edge information or texture features. It also does not adequately consider more detailed feature information from the convolutional kernel, resulting in an inability to capture subtle differences between different target types. Consequently, this leads to missed detections and false positives when detecting small objects. CBAM (Woo et al., 2018) addresses this limitation by integrating channel and spatial information, while utilizing an adaptive learning method, enabling the network to more accurately focus on the spatial location of the target. Coupled with its ability to capture abstract features of small objects, this helps enhance the network's performance in detecting small targets. The approximate structure of the CBAM module is shown in Fig. 6.

The model of CBAM can prevent the loss of partial target features caused by operations involving feature map dimensionality changes. It enables adaptive cross-channel information interaction. It illustrates the overall structure after the integration of the CBAM module. As can be observed, the output of the convolutional layers first undergoes channel attention to obtain weighted results. Subsequently, it undergoes spatial attention, and then the results are obtained through weighted aggregation. Given a feature map, CBAM infers attention maps independently along channel and spatial dimensions. These attention maps are then multiplied with the input feature map to perform adaptive feature refinement. According to the experiments in reference, integrating CBAM into the maritime target detection model significantly improves the model's performance, demonstrating the effectiveness of this module.

3.2.2. Feature pyramid networks based on FECG

The structure of FECG is shown in Fig. 7. It replaces the CBS modules in FPN + PAN with GhostConv modules to reduce the model's parameters and FLOPs. Additionally, CBAM is introduced to fuse shallow and deep feature maps at the pixel level, enhancing the network's adaptive capabilities and diversifying feature maps. This, in turn, improves the accuracy of extracting ship features from the network. The FECG structure reduces the number of model parameters and FLOPs while maintaining the accuracy and speed of small object detection. This paves the way for future deployment on mobile devices with limited computational resources.

3.3. S-IoU based loss function

The effectiveness of object detection is a crucial issue in computer vision tasks, heavily relying on the definition of the loss function that evaluates the accuracy of ML model predictions. Traditional loss functions for object detection mainly aggregate bounding box regression

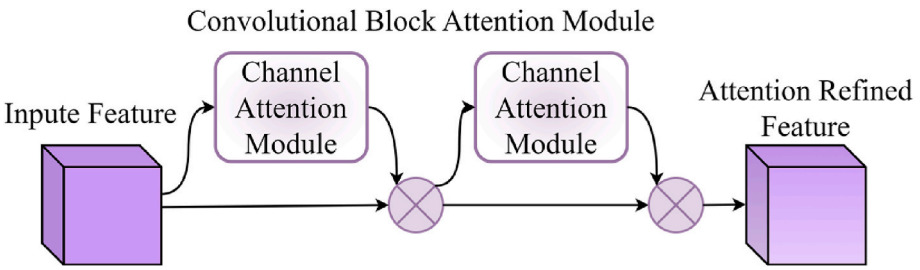


Fig. 6. Partial structural diagram of the CBAM attention mechanism.

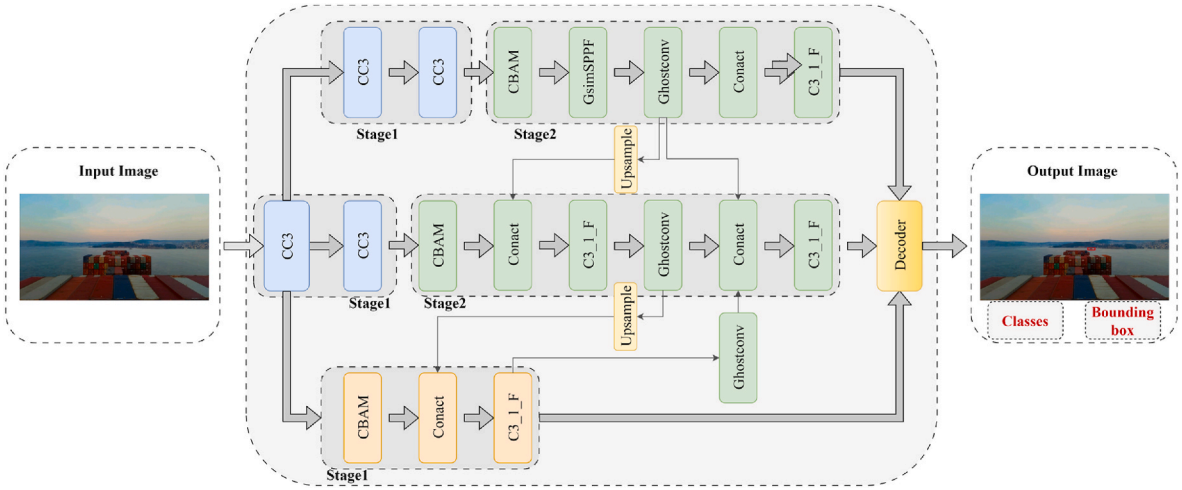


Fig. 7. Structural schematic diagram of the FECG module.

metrics, such as distances between predicted and ground truth boxes (e.g., GIoU (Rezatofighi et al., 2019), CIoU (Zheng et al., 2020), EIoU (Zhang et al., 2022a), overlap areas, and aspect ratios. However, these loss functions fail to consider the deviation between the required ground truth box and the predicted "experimental" box. This limitation might lead to slower convergence during the training process and reduced efficiency, as predicted boxes may drift, resulting in a poorer model. To address these challenges, this paper introduces a novel loss function called SIoU (Gevorgyan, 2022), which mitigates the penalization metric by incorporating angles between the required regressions. By introducing directionality in the cost of the loss function, faster convergence can be achieved during training, improving inference performance for faster and more accurate convergence. This, in turn, enhances the accuracy of small object detection.

The SIoU loss function consists of four cost functions: angle cost, distance cost, shape cost, and IoU cost. The idea behind merging this angle-aware low-frequency component is to minimize the number of variables related to the "ambiguity" associated with distance. Essentially, the model will attempt to align predictions first along the X or Y axis (whichever is closer) and then proceed to approach along their respective axes.

The angle loss calculation strategy is illustrated in Fig. 8. Given the target box B and the regression box B^{GT}, if the angle between B and B^{GT} is smaller than α, which converges towards the minimum value α; otherwise, it converges towards β.

$$\wedge = 1 - 2\,sin^2\left(arcsin(x) - \frac{\pi}{4}\right) \tag{12}$$

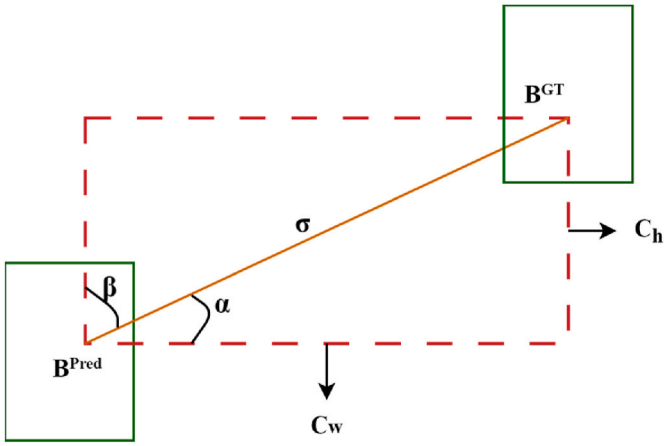


Fig. 8. The calculation scheme for the angle cost contribution in the loss function.

$$x = \frac{c_h}{\sigma} = sin\,\alpha$$
$$\sigma = \sqrt{\left(b^{gt}_{c_x} - b_{c_x}\right)^2 + \left(b^{gt}_{c_y} - b_{c_y}\right)^2} \tag{13}$$
$$c_h = max\left(b^{gt}_{c_y}, b_{c_y}\right) - min\left(b^{gt}_{c_y}, b_{c_y}\right)$$

The derivation leads to the conclusion that the loss is twice the sine value of the current angle, which is reasonable. When the angle is 0°, it directly regresses to the left. When α is $\pi/4$, the sine value of $(\pi/4 \times 2)$ equals 1, which is the maximum value. Thus, it holds true that $\alpha < \pi/4$.

Distance cost:

$$\Delta = \sum_{t=x,y} (1 - e^{-\gamma \rho_t}) \rho_x = \left(\frac{b_{c_x}^{gt} - b_{c_x}}{c_w} \right)^2, \rho_y = \left(\frac{b_{c_y}^{gt} - b_{c_y}}{c_h} \right)^2, \gamma = 2 - \Lambda \quad (14)$$

shape cost:

$$\Omega = \sum_{t=w,h} (1 - e^{-\omega_t})^{\theta} \quad (15)$$

where

$$\omega_w = \frac{|w - w^{gt}|}{max(w, w^{gt})}, \omega_h = \frac{|h - h^{gt}|}{max(h, h^{gt})} \quad (16)$$

This part of the formula observes whether the current regression box shape is similar to the labeled box.

The final loss function:

$$L_{\text{box}} = 1 - IoU + \frac{\Delta + \Omega}{2} L = W_{box} L_{box} + W_{cls} L_{els} \quad (17)$$

In summary, the SIoU loss function improves accuracy by simultaneously penalizing misalignments in orientation, distance, and shape, which guides the model to converge more quickly to the optimal bounding box positions. The combination of these cost components reduces the drift of predicted boxes and results in a higher mAP, particularly beneficial for small object detection. Experimental results summarized in Table 1 confirm that the use of SIoU leads to faster convergence and improved accuracy compared to traditional loss functions.

3.4. Nav-Yolo

In summary, this paper presents a one-stage object detection model that is better suitable for small objects. The model architecture, as shown in Fig. 9, incorporates the CBAM attention mechanism module to enhance the feature extraction capability of the network. Additionally, it combines a feature pyramid structure based on attention mechanisms. By ensuring fast inference speed and reducing the model's parameter count, the proposed model achieves improved accuracy, effectively addressing the challenges of detecting small objects and mitigating the issue of missed detections in the existing small object detection models.

4. Experimental evaluation

Experiments were conducted on a server equipped with an NVIDIA Tesla V100 GPU (32 GB of RAM) running a 64-bit Ubuntu operating system. The ShipNav dataset was randomly divided into training, validation, and test sets in an 80:20 ratio. For model training, we used a batch size of 16 and an initial learning rate of 0.01. The optimizer employed was Stochastic Gradient Descent (SGD) with a momentum of 0.9 and a weight decay (L2 regularization) factor of 0.0005 to mitigate overfitting. The model, which consists of 427 layers, was trained for 200 epochs. These hyperparameter settings were chosen based on preliminary experiments and are consistent with configurations reported in related studies to balance convergence speed and generalization performance.

Moreover, ablation experiments were conducted to compare different modules of the improved model. From Table 1 and it can be observed that the introduction of SimSPPF and SIoU slightly improved the accuracy of baseline model while reducing the number of model parameters. From the ablation experiments, it can be observed that incorporating the improved modules into the original model resulted in a reduction in parameter count and an improvement in model accuracy. Typically, behind a CNN network, there are fully connected layers that require a fixed input size. Therefore, input images are often resized to a fixed dimension, which may lead to geometric distortion and impact accuracy. The SimSPPF module addresses this issue by pooling feature maps of different scales, fixing them into feature vectors of the same length, and then passing them to fully connected layers. Although the structure of the fully connected layers after the convolutional layers is fixed, in practice, the input image size may not meet the required dimensions. Traditional methods involve cropping and warping, which can distort the original features. However, the SimSPPF layer divides the feature map of candidate regions into multiple grids of different sizes and performs max-pooling within each grid. This allows subsequent fully connected layers to receive a fixed input. By doing so, the SimSPPF layer improves accuracy while reducing the number of model parameters, making it suitable for deployment on mobile devices.

Table 1
The impact of different optimization strategies on the model.

Model	mAP (@50 %)	FPS(f/s)	Params (MB)	GFLOPs
Baseline	65.1	50.1	82.9	51.1
Baseline + SimSPPF	67.0	49.8	77.7	47.5
Baseline + GhostCon	65.8	51.1	81.6	50.3
Baseline + CBAM	65.2	49.7	82.9	51.1
Baseline + SIoU	65.6	51.2	82.9	51.1

The reduction in model parameters and computational complexity not only enhances the efficiency of the model but also substantially affects the consumption of computational resources such as CPU and GPU usage, and power consumption. To begin with, the reduced number of model parameters directly translates to a lower memory footprint on devices. This reduction is beneficial for edge devices with limited RAM and storage capacity, allowing them to run advanced models without necessitating additional memory. For instance, the decrease in parameter size from 82.9 MB in the baseline model to 73.6 MB in our optimized model implies that less memory is required to store the model weights, thereby freeing up resources for other processes or enabling the deployment of multiple models simultaneously on the same device. Furthermore, the reduction in computational complexity, measured in GFLOPs, indicates a lower requirement for computational power. Lower GFLOPs mean that the model requires fewer floating-point operations per second, which directly correlates with less CPU and GPU utilization during inference. This is particularly important for real-time applications where rapid response times are critical, and high computational demands can lead to latency issues. In our experimental setup, the optimized model demonstrated a decrease in GFLOPs from 51.1 to 44.6 when compared to the baseline. This 12.7 % reduction signifies that our model can operate more efficiently under the computational constraints of edge devices.

The incorporation of GhostConv has improved the accuracy of the model but slightly reduced the detection speed. By stacking convolutional layers, rich feature information, including redundant information, can be captured, aiding the network in a more comprehensive understanding of the data. Therefore, the GhostConv module extracts rich feature information through conventional convolutional operations while using a more cost-effective linear transformation to generate redundant feature information. This approach effectively reduces the required computational resources, simplifies the model's design, and facilitates its industrial deployment.

In this section, various optimization strategies were evaluated by comparing the improved model with baseline model, using a dataset with an input image size of 640*640. As shown in Table 2, compared to the baseline model, the improved model's mAP@50 increased by 5.1 %, the number of parameters decreased by 11.2 %, and the fps slightly decreased. The analysis suggests that the improved model has advantages in detection accuracy and model parameters across three different input image sizes. It reduces the number of model parameters while improving detection accuracy. Deployment on mobile devices requires consistent computational capability and complexity, considering factors such as power consumption, heat dissipation, size, cost, and security requirements. Reducing the number of model parameters can significantly decrease power consumption and reduce heat dissipation.

Furthermore, Fig. 10 presents a comparative analysis of the training

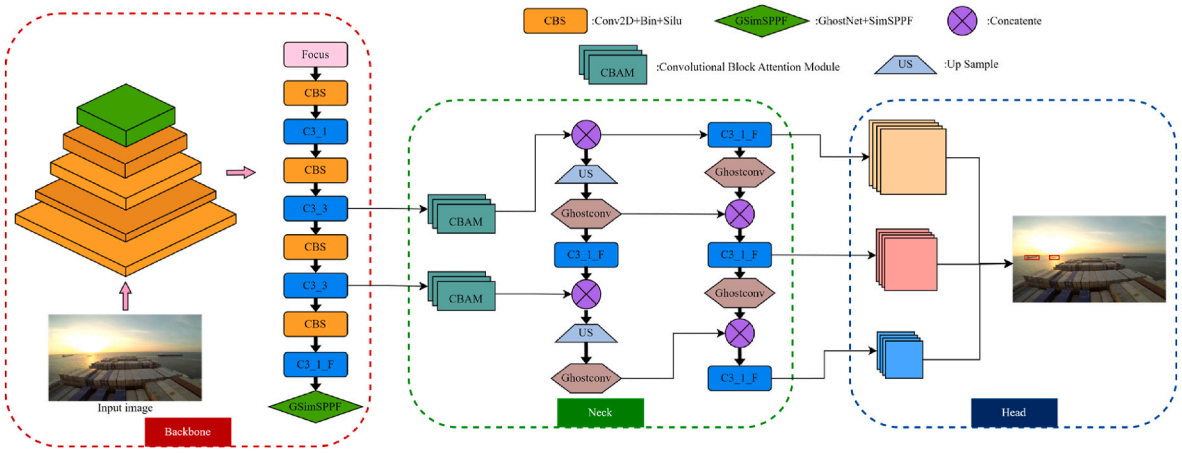


Fig. 9. The structure diagram of the Nav-YOLO model, integrating the GsimSPPF module and FECG module.

Table 2
The comparative experimental results of different models.

Model	Input Size	Precision	Recall	mAP (@50 %)	FPS	Params
Baseline	448*448	65.3	60.8	63.6	57.4	82.9
Ours	448*448	64.9	61.3	**63.7**	**56.8**	**73.6**
Baseline	544*544	66.2	61.1	64.4	53.6	82.9
Ours	544*544	67.7	63.3	**65.3**	**51.5**	**73.6**
Baseline	640*640	68.6	64.1	65.1	50.1	82.9
Ours	640*640	71.2	67.3	**68.6**	**49.6**	**73.6**

and validation box loss curves over 200 epochs for both the baseline object detection model and the improved model "Ours". The curves are clearly labeled as Training Loss and Validation Loss to ensure consistency. Both models show a rapid decrease in loss during the initial epochs, which indicates effective initial learning. As training proceeds, the improved model consistently exhibits lower loss values than the baseline model in both training and validation phases. This observation suggests that the improved model has enhanced learning efficiency and superior generalization capabilities. The improvements in the model, which include architectural changes, advanced regularization techniques, and the use of the optimized SIoU loss function, contribute to a better fit to the training data and improved adaptability to unseen validation data. These characteristics underscore the potential of the improved model to deliver more accurate and reliable object detection in practical scenarios.

From Table 3, it can be observed that the improved model exhibits a significant reduction in GFLOPs compared to RT-DETRv2-M, with a 12.7 % decrease compared to baseline and with a 34.4 % decrease compared to YOLOv11. Combining the information in the table, this characteristic makes the model more lightweight.

In terms of small object detection, the new improved model demonstrates notable advancements compared to baseline model. Specifically, Table 4 shows a 4.3 % improvement for 'Work boat' and a 6.2 % improvement for 'Cargo ship' compared to the baseline model. The

Table 3
GFLOPs of different models.

Model	RT-DETRv2-M	Baseline	DAMO-YOLO	SAHI-YOLO	YOLOv11	Ours
GFLOPs	100	51.1	61.8	78.9	68.0	**44.6**

Table 4
mAP@50(%) of small target and night target.

Model	Input Size	Work boat	Cargo ship	Un ship-N
Baseline	448*448	59.5	66.1	63.7
YOLOv11	448*448	60.2	57.6	54.9
Ours	448*448	**61.1**	**69.8**	**71.9**
Baseline	544*544	59.9	66.6	64.5
YOLOv11	544*544	60.8	58.9	54.2
Ours	544*544	**61.8**	**70.6**	**72.7**
Baseline	640*640	60.3	67.0	66.5
YOLOv11	640*640	61.0	60.8	55.4
Ours	640*640	**62.9**	**71.2**	**73.4**

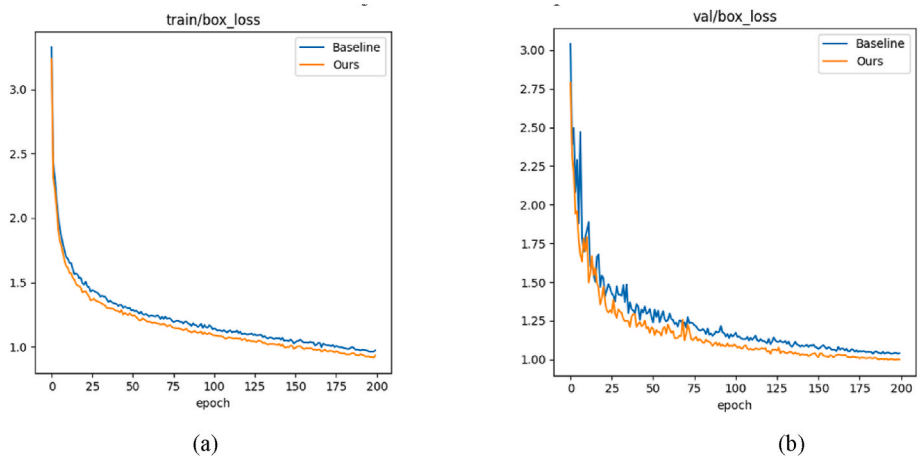


Fig. 10. (a) Comparative analysis of training and validation loss over 200 epochs for baseline and improved object detection models.

improved model outperforms the latest YOLOv11 in small object detection. In scenarios involving the movement of ships in ports, small objects are often located in extremely small areas within the image and are highly susceptible to environmental interference. Even a one-pixel deviation during network prediction can have a significant impact on small objects. Additionally, compared to larger or medium-sized objects, small objects have lower resolution, providing limited information and presenting challenges in extracting discriminative features. Our improved model effectively addresses common issues associated with small objects, including partial occlusion, blurriness, incompleteness, and difficulty in identification. The comparison between the detection performance of the improved model and the original model is shown in Fig. 11. The figure clearly shows that the improved model surpasses mainstream models in detecting small targets in terms of precision.

Moreover, there is a 10.2 % improvement for "Unidentified ship-N" in nighttime small object detection. Due to low illumination levels, high grayscale values, reduced color discrimination ability, and significant interference factors (including large shadow areas), nighttime conditions pose challenges. Additionally, existing nighttime image acquisition largely relies on infrared devices, which suffer from low resolution, poor depth perception, and limited texture information. These factors collectively hinder the detection of objects such as ships during nighttime operations. By enhancing the backbone network and improving the performance of the object detector, this work partially addresses the challenges posed by unknown light sources at sea, thereby improving the accuracy of ship recognition in water environments.

The proposed model significantly outperforms existing methods in maritime object detection by demonstrating robustness to environmental variations, superior real-time processing capabilities, and scalable performance across different maritime settings. Enhanced by the integration of CBAM and SPP modules, and further optimized with the S-IoU loss function, the model achieves a notable improvement in detecting small and distant targets, even under challenging conditions like low light and high interference. These advancements are quantified by a 5.1 % increase in mAP@50 %, alongside reductions in model parameters and computational complexity, making it highly effective for real-world applications in autonomous maritime navigation.

5. Discussion

The real-time multi-scale object detection framework has been enhanced to mitigate the shortcomings related to contextual information for distant small targets, addressing the critical need for prompt responses in vehicular navigation contexts. Initially, we incorporated the SimSPPF and GhostConv modules, which effectively retain information across each channel while minimizing computational demands, thus boosting the model's processing velocity. Subsequently, the CMAB attention mechanism was amalgamated with the feature fusion network, striking a balance between precision in detecting small targets and operational speed, in addition to streamlining the model's parameters. Finally, the refined model, employing the SIoU loss function, has significantly enhanced the accuracy of bounding box regression and the precision of small object detection with equivalent parameters. From the perspective of crews, this represents a notable advancement.

5.1. Critical issue of the long-distance and small target detection

The changing marine environment significantly impacts small object detection, especially in global shipping route scenarios, necessitating models with strong generalization capabilities. In the context of ship-bridge scenes, long-range target detection may be affected by background interference, including city lights, port facilities, and fishing areas, introducing challenges to the algorithm. Firstly, when a vessel navigates through areas with bright city lights, the intense urban illumination can cause distant targets to become blurred or difficult to discern in the image. This may require the algorithm to adapt to varying

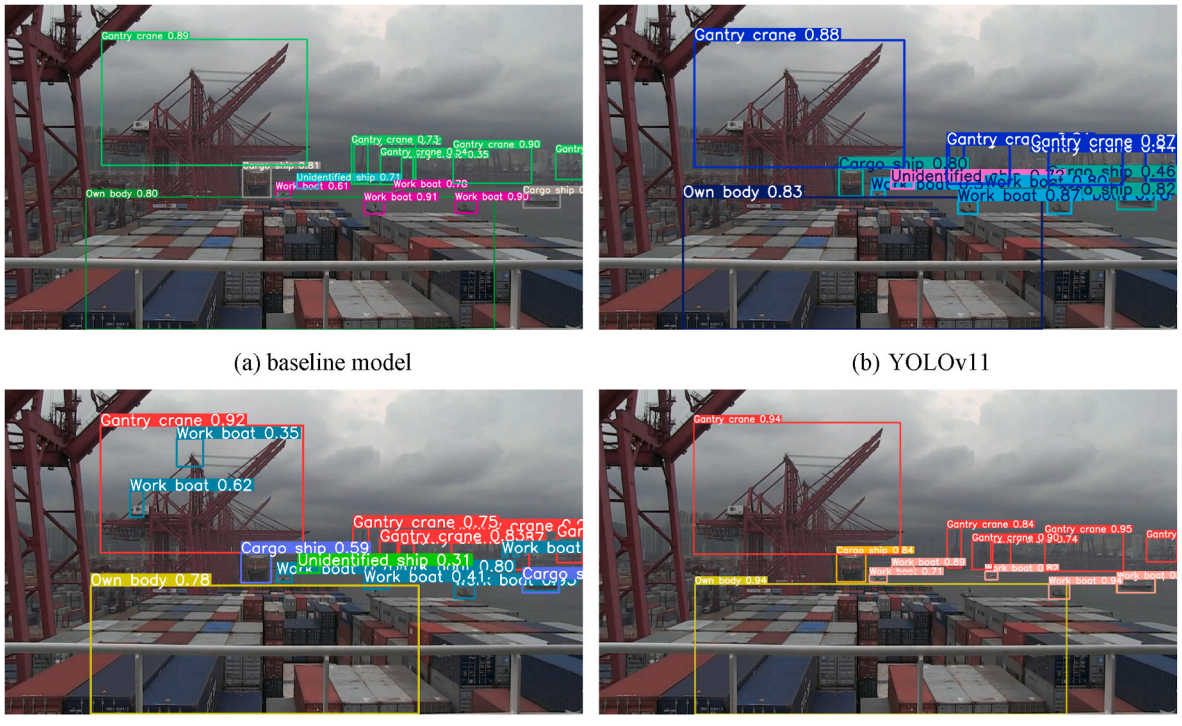


Fig. 11. The figure (a) illustrates the detection results using the baseline model and (b) displays the results of small object detection by YOLOv11. (c) shows the detection results of SAHI-Yolo, while (d) depicts the detection performance of the improved model. Through comparison, the improved model demonstrates a significant enhancement in the detection of small objects. nts.

lighting conditions for effective target detection. Secondly, port areas typically feature various facilities such as lighthouses, cranes, and docks. These structures may resemble the target or introduce complex textures and edges in the image, complicating target detection. Lastly, in fishing areas, there may be numerous fishing boats, buoys, and other objects that can be easily confused with distant targets. The intricate aquatic environment in such areas may pose challenges for target detection algorithms to accurately identify distant vessels. Moreover, Various conditions, such as different weather scenarios or crowded maritime environments, must also be considered to enhance the generalizability analysis of the model. Due to space limitations in the paper, these discussions will be addressed in future analyses.

5.2. Challenges and non-commercial applications of advanced detection systems in maritime navigation

Advanced detection systems in maritime navigation face several operational challenges that impact their effectiveness. These challenges include vast geographic variability, where maritime routes traverse diverse environmental conditions that require datasets covering a wide range of scenarios to maintain accuracy. In addition, small targets often exhibit dynamic behavior in maritime settings, complicating detection due to the lack of effective strategies for switching between identifying small, dynamic objects and more clearly defined targets within the same scene. Quantitatively analyzing the intrinsic relationship between distant and small targets is also challenging in complex maritime environments, as it is difficult to delineate logical changes between these types without a clear, objective framework. Moreover, conventional monocular and binocular ranging methods are ineffective in maritime contexts, preventing computer vision systems from objectively assessing target distances because “distant targets” are often defined subjectively rather than through empirical measures.

The practical implications of these challenges extend into non-commercial applications. For Vessel Traffic Services (VTS) monitoring, the enhanced detection framework can improve safety by accurately detecting small vessels and obstacles even in congested or visually challenging environments, thereby supporting more effective traffic management and early hazard detection. In marine conservation, the proposed strategy enables more precise monitoring of marine life and environmental conditions by accurately identifying small and often overlooked organisms or debris, which is essential for assessing ecosystem health. For surveillance against illegal fishing, improved detection capabilities can assist enforcement agencies by reliably identifying suspicious activities or unauthorized vessels in real time, thereby enhancing the ability to intervene and protect marine resources.

5.3. Building datasets and model improvement strategies

In this study, the research team employed specific methods to construct the ShipNav dataset and implement strategies for model improvement. The process began with the collection and annotation of images depicting small targets relevant to maritime navigation. The ShipNav dataset is distinctively curated to include images from the perspective of a ship’s bridge, capturing a variety of weather conditions and detailed shore-based backgrounds, which enhances the model’s practical applicability.

To improve the model’s detection accuracy and efficiency, the research team integrated advanced attention mechanisms, specifically CBAM and SimSPPF. Additionally, they introduced a novel loss function, the S-IoU, designed to refine the accuracy of target detection further. These enhancements are aimed at optimizing the model to handle the complexities of real-world maritime navigation scenarios more effectively.

While the proposed multi-scale object detection framework significantly enhances target detection capabilities in maritime navigation, it encounters limitations regarding scalability, adaptability to different ship types, and performance under varied environmental conditions. The framework’s optimization for specific scenarios raises questions about its scalability to different maritime operations and its adaptability across diverse ship environments, such as varying bridge designs and electronic interferences. Additionally, its efficacy in adverse weather conditions and integration with existing vessel systems presents challenges that need to be addressed. Future research should focus on enhancing scalability, ensuring robust performance across different ship types and environmental conditions, and achieving seamless integration with existing navigation systems to ensure the framework’s applicability in real-world maritime operations.

5.4. Selection of baseline model

Considering the balance between performance and efficiency, this paper selects YOLOv5m as the baseline model instead of YOLOv11 or other small object detection models for deployment on edge servers. Firstly, YOLOv5m offers a good balance between speed and accuracy. For edge computing applications, such as deployments on edge servers, processing speed and response time are crucial. YOLOv5m provides faster inference speeds without sacrificing too much detection accuracy. Secondly, the model size of YOLOv5m is moderate, and its computational requirements are suitable for the constraints of edge devices. Edge devices typically have limited processing power and storage space; the relatively smaller model of YOLOv5m can operate better on these devices without excessively consuming resources. Finally, YOLOv5 is a thoroughly validated model with numerous successful deployment cases. Compared to newly launched models such as YOLOv11, YOLOv5 has higher technical maturity and lower risk.

5.5. Edge service system based on computer vision

This research significantly advances maritime safety and autonomous navigation by enhancing the capability of detection systems to reliably identify small and distant objects in various environmental conditions. Integrating these improved detection frameworks into edge servers such as the Jetson Nano, combined with navigational aids like buoy lights and AIS, could substantially enhance maritime navigation by providing robust, real-time decision support directly on the vessel. The Jetson Nano, with its compact size and substantial processing power, is well-suited for deploying advanced object detection models that require real-time analysis and minimal latency. This setup would allow vessels to process complex visual and sensor data on-board, ensuring immediate response capabilities.

Utilizing buoy lights and AIS devices alongside the enhanced detection framework can significantly improve situational awareness. Buoy lights help in identifying safe waterways and navigation channels, while AIS provides vital information on nearby vessels, such as their identity, position, speed, and heading. By integrating these technologies with the detection framework on the Jetson Nano, vessels can achieve a higher level of navigational safety, effectively detect and avoid potential hazards, and comply with maritime traffic regulations more efficiently. This system could autonomously adjust to dynamic maritime environments, offering predictive insights and enhanced decision-making support, which is critical in avoiding collisions and navigating through complex maritime routes. Additionally, the integrated system can be programmed to alert crew members to potential hazards and automatically take preventive actions if needed, thereby increasing the overall safety and efficiency of maritime operations.

6. Conclusions

The research addresses the challenge of detecting small and distant targets in autonomous maritime navigation, which traditional systems like AIS and radar struggle with, especially in complex and congested maritime environments. It introduces a novel architecture that

integrates the Convolutional Block Attention Module (CBAM) and optimizes the Spatial Pyramid Pooling (SPP) module, achieving a 5.1 % improvement in detection accuracy and significant reductions in model complexity. These enhancements help enhance maritime safety by enabling more reliable detection of small objects, crucial for navigating safely in challenging conditions.

The advancements detailed in this study are particularly relevant for the development and operational enhancement of autonomous vessels. The ability of our model to precisely detect small, distant targets in complex maritime environments underlines its potential for integration into autonomous ship navigation systems, which are pivotal for enhancing maritime safety. Autonomous vessels rely heavily on accurate and timely information about their surroundings to navigate safely, especially in congested or challenging maritime corridors. By improving the detection capabilities of small objects, such as buoys, small boats, and debris, the model directly contributes to reducing the risk of collisions and navigational errors. Furthermore, the application of this vision-based method extends to a variety of maritime operations, including search and rescue missions where rapid and reliable detection of objects is crucial, and environmental monitoring where accurate detection of small objects can aid in tracking pollution sources or marine life. The enhanced detection capability also supports the implementation of geofencing and other regulatory compliance measures, ensuring vessels operate within safe and legally designated areas.

In summary, the technological advancements presented in this paper do not only push the boundaries of computer vision in maritime settings but also offer tangible benefits for the safety and efficiency of maritime operations, particularly in the context of increasingly autonomous maritime navigation systems. Future work will focus on further refining the model for deployment on edge devices, aiming to optimize its efficiency and applicability in real-world scenarios where computational resources are limited. This endeavor aligns with the ongoing need for advanced detection systems capable of operating in diverse and challenging maritime settings, thereby contributing to the advancement of autonomous maritime navigation technologies.

CRediT authorship contribution statement

Ruolan Zhang: Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Funding acquisition, Conceptualization. **Xingchen Ji:** Writing – original draft, Validation, Supervision, Data curation. **Sean Loughney:** Writing – review & editing, Writing – original draft, Investigation. **Jin Wang:** Writing – review & editing, Conceptualization. **Zaili Yang:** Writing – review & editing, Validation, Supervision.

Data availability

Data will be made available on request.

Declaration of competing interest

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

Acknowledgments

This work was partially funded by the Liaoning Provincial Department of Education project (Research project No. LJKMZ20220374), This work is also supported through the International Association of Maritime Universities (IAMU) and The Nippon Foundation in Japan. The authors would like to acknowledge the support of the International Association of Maritime Universities (research project number 20240201).

References

Adjabi, I., Ouahabi, A., Benzaoui, A., et al., 2020. Past, present, and future of face recognition: a review. Electronics 9 (8), 1188.
Akyon, F.C., Altinuc, S.O., Temizel, A., 2022. Slicing aided hyper inference and fine-tuning for small object detection. In: 2022 IEEE International Conference on Image Processing (ICIP), pp. 966–970. https://doi.org/10.1109/ICIP46576.2022.9897990.
AliAkbarpour, H., Moori, A., Khorramdel, J., et al., 2024. Emerging trends and applications of neuromorphic Dynamic vision sensors: a survey. IEEE Sens. Rev. 1, 14–63.
Benjumea, A., Teeti, I., Cuzzolin, F., et al., 2021. YOLO-Z: improving small object detection in YOLOv5 for autonomous vehicles. CoRR. abs/2112.11798.
Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M., 2020. Yolov4: optimal speed and accuracy of object detection. CoRR abs/2004.10934. https://doi.org/10.48550/arXiv.2004.10934.
Chen, L., Li, Y., Huang, C., et al., 2022. Milestones in autonomous driving and intelligent vehicles: survey of surveys. IEEE Transact. Intell. Vehicles 8 (2), 1046–1056.
Everingham, M., Van Gool, L., Williams, C.K.I., et al., 2010. The pascal visual object classes (voc) challenge. Int. J. Comput. Vis. 88, 303–338.
Ghazali, M.H.M., Satar, M.H.A., Rahiman, W., 2024. Unmanned surface vehicles: from a hull design perspective. Ocean Eng. 312, 118977.
Girshick, R., 2015. Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448.
Han, K., Wang, Y., Tian, Q., et al., 2020. Ghostnet: more features from cheap operations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1580–1589.
Han, K., Wang, Y., Chen, H., et al., 2022. A survey on vision transformer. IEEE Trans. Pattern Anal. Mach. Intell. 45 (1), 87–110.
Hirzel, M., Schneider, S., Tangwongsan, K., 2017. Sliding-window aggregation algorithms: tutorial. In: Proceedings of the 11th ACM International Conference on Distributed and Event-based Systems, pp. 11–14.
Jocher, G., Chaurasia, A., Qiu, J., 2023. YOLO by ultralytics. URL: https://github.com/ultralytics/ultralytics.
Jocher, G., Qiu, J., Chaurasia, A., 2023. Ultralytics YOLO [Computer software], Version 8.0.0. https://github.com/ultralytics/ultralytics.
Kheradmandi, N., Mehranfar, V., 2022. A critical review and comparative study on image segmentation-based techniques for pavement crack detection. Constr. Build. Mater. 321, 126162.
Kiefer, B., Kristan, M., Perš, J., et al., 2023. 1st workshop on maritime computer vision (macvi) 2023: challenge results. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 265–302.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 25.
Gevorgyan, Z., 2022. SIoU loss: More powerful learning for bounding box regression. arXiv preprint arXiv:2205.12740.
Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., 2022. YOLOv6: A single-stage object detection framework for industrial applications. arXiv preprint arXiv:2209.02976.
Lin, T.Y., Maire, M., Belongie, S., et al., 2014. Microsoft coco: common objects in context. In: Computer Vision–ECCV 2014: 13Th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer International Publishing, pp. 740–755.
Liu, W., Anguelov, D., Erhan, D., et al., 2016. Ssd: Single Shot Multibox detector[C]// Computer Vision–ECCV 2016: 14Th European Conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing, pp. 21–37.
Moosbauer, S., Konig, D., Jakel, J., et al., 2019. A benchmark for deep learning based object detection in maritime environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 0-0.
Pang, Y., Wang, T., Anwer, R.M., et al., 2019. Efficient featurized image pyramid network for single shot detector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7336–7344.
Qiao, H., Chen, J., Huang, X., 2021. A survey of brain-inspired intelligent robots: integration of vision, decision, motion control, and musculoskeletal systems. IEEE Trans. Cybern. 52 (10), 11267–11280.
Qin, Y., Yan, Y., Ji, H., et al., 2021. Recursive correlative statistical analysis method with sliding windows for incipient fault detection. IEEE Trans. Ind. Electron. 69 (4), 4185–4194.
Redmon, J., Farhadi, A., 2017. YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271.
Redmon, J. and Farhadi, A., 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
Redmon, J., Divvala, S., Girshick, R., et al., 2016. You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788.
Rezatofighi, H., Tsoi, N., Gwak, J.Y., et al., 2019. Generalized intersection over union: a metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 658–666.
Saleh, A., Sheaves, M., Rahimi Azghadi, M., 2022. Computer vision and deep learning for fish classification in underwater habitats: a survey. Fish Fish. 23 (4), 977–999.
Shao, Z., Wu, W., Wang, Z., et al., 2018. Seaships: a large-scale precisely annotated dataset for ship detection. IEEE Trans. Multimed. 20 (10), 2593–2604.
Shen, A., Zhu, Y., Angelov, P., Jiang, R., 2024. Marine debris detection in satellite surveillance using attention mechanisms. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 17, 4320–4330.

Sunkara, R., Luo, T., 2022. No more strided convolutions or pooling: a new CNN building block for low-resolution images and small objects. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer Nature Switzerland, Cham, pp. 443–459.
Tang, Y., Zhao, C., Wang, J., Zhang, C., Sun, Q., Zheng, W.X., Du, W., Qian, F., Kurths, J., 2022. Perception and navigation in autonomous systems in the era of learning: A survey. IEEE Transactions on Neural Networks and Learning Systems 34 (12), 9604–9624.
Vaswani, A., Shazeer, N., Parmar, N., et al., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.
Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M., 2023. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7464–7475.
Woo, S., Park, J., Lee, J.Y., et al., 2018. Cbam: convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19.
Xu, X., Jiang, Y., Chen, W., Huang, Y., Zhang, Y., Sun, X., 2022. Damo-yolo: A report on real-time object detection design. arXiv preprint arXiv:2211.15444.
Xu, S., Zhang, M., Song, W., et al., 2023. A systematic review and analysis of deep learning-based underwater object detection. Neurocomputing 527, 204–232.
Zhang, P., Huang, W., Chen, Y., et al., 2023. A novel deep-learning-based QoS prediction model for service recommendation utilizing multi-stage multi-scale feature fusion with individual evaluations. IEEE Trans. Autom. Sci. Eng. 21 (2), 1740–1753.
Zhang, H., Xiao, L., Cao, X., et al., 2022a. Multiple adverse weather conditions adaptation for object detection via causal intervention. IEEE Trans. Pattern Anal. Mach. Intell.vol. 46 (3), 1742–1756.
Zhang, Z., Yang, N.W., Yang, Y.J., 2024. Autonomous navigation and collision prediction of port channel based on computer vision and lidar. Sci. Rep. 14 (1), 11300.
Zhao, C., Liu, R.W., Qu, J., et al., 2024. Deep learning-based object detection in maritime unmanned aerial vehicle imagery: review and experimental comparisons. Eng. Appl. Artif. Intell. 128, 107513.
Zhao, Y., Lv, W., Xu, S., et al., 2024. Detrs beat yolos on real-time object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16965–16974.
Zheng, Z., Wang, P., Liu, W., et al., 2020. Distance-IoU loss: faster and better learning for bounding box regression. Proc. AAAI Conf. Artif. Intell. 34 (7), 12993–13000.
Zhou, X.Y., Liu, Z.J., Wang, F.W., et al., 2021. A system-theoretic approach to safety and security co-analysis of autonomous ships. Ocean Eng. 222, 108569.
Zhu, X., Lyu, S., Wang, X., et al., 2021. TPH-YOLOv5: improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2778–2788.
Zou, Z., Chen, K., Shi, Z., et al., 2023. Object detection in 20 years: a survey. Proc. IEEE. 111 (3), 257–276.

Glossary

F_n, F_s: The FLOPs of regular convolution and GhostConv
y_i': The i feature map among the m intrinsic feature map
c: The input channel
h': The height of the output feature map
h: The image height
w': The width of the output feature map
w: The image width
AIS: Automatic Identification Systems
CBAM: Convolutional Block Attention Module
CNN: Convolutional Neural Networks
CSP: Cross-Stage Partial
CSP-Net: Cross-Stage Partial Network
FECG: Feature extraction module based on CBAM with GhostConv moudles
FPN: Fusion Pyramid Network
GhostConv: Ghost Convolution
GIoU: Generalized Intersection over Union
MS-COCO: Microsoft Common Objects in COntext
NMS: Non-Maximum Suppression
PAN: Path Aggregation Network
S-IoU: Scaled Generalized Intersection over Union
SimPPF: Similarity-based Spatial Pyramid Pooling Fusion
SPP: Spatial Pyramid Pooling
SPPF: Spatial Pyramid Pooling Fusion
SSD: Single Shot Multibox Detector
UAVs: unmanned aerial vehicles
USV: Unmanned Surface Vehicles
WSODD: Water Surface Object Detection Dataset