Improved Lightweight Marine Oil Spill Detection Algorithm of YOLOv8[1]

Jianting Shi[a,b], Tianyu Jiao[a], Daniel P. Ames[b,*], Zhijun Li[c]

[a] College of Computer and Information Engineering, Hei Longjiang University of Science and Technology,Harbin 150022, China

[b] College of Engineering - Civil and Construction Engineering, Brigham Young University, Provo, UT, 84602, United States

[c] The Guangxi key Laboratory of Machine Vision and Intelligent Control, Wuzhou University, Wuzhou, 543002, China

Abstract: Marine oil spill detection using Synthetic Aperture Radar (SAR) is crucial but challenged by dynamic marine conditions, diverse spill scales, and limitations in existing algorithms regarding model size and real-time performance. We introduce LSFE-YOLO, a lightweight marine oil spill detection model built upon an optimized YOLOv8s architecture. Our approach integrates FasterNet into the backbone and adjusts the neck network width to reduce memory overhead and parameters, significantly boosting detection speed. We also propose a novel GroupNorm and Lightweight Shared Convolution (GN-LSC) Head detection module that features a shared convolutional structure. This structure not only decreases the number of parameters in the detection head but also minimizes redundant computations. Further enhancements include incorporating the Squeeze-and-Excitation (SE) attention mechanism to improve feature extraction and a C2f_MBE module for enhanced multi-scale feature representation. Experimental results show LSFE-YOLO improves mAP by 1.3% and F1 score by 1.7% over YOLOv8s, while achieving substantial reductions in model size (81.9%), parameter count (82.9%), and computational cost (84.2%), alongside a 20 FPS increase in detection speed. LSFE-YOLO offers an efficient and effective solution for real-time marine oil spill detection.

Keywords: SAR images 1; YOLOv8 2; deep learning 3; marine oil spill detection 4; lightweight 5

1. Introduction

The marine environment faces significant threats from oil spills caused by maritime transportation and offshore oil drilling platforms, exacerbating the problem of oil pollution (Leifer et al., 2012). Due to the ocean's limited self-purification capacity, oil spills can cause severe damage to marine ecosystems, severely impacting ecological balance (Brekke and Solberg, 2005). With advancements in satellite remote sensing technology, Synthetic Aperture Radar (SAR) technology has proven to be an effective means for monitoring marine oil spills due to its all-weather, all-day capabilities (Wan and Cheng, 2013). When short gravity waves and capillary waves affect the sea surface, the microwave Bragg scattering on the sea surface

[1] ∗ Corresponding author.

E-mail addresses: jianting@byu.edu (Jianting Shi), 2959162543@qq.com(Tianyu Jiao), dan.ames@byu.edu (Daniel P. Ames), zhijun_dl@163.com (Zhijun Li) significantly weakens. This makes the oil film appear as prominent black spots in SAR images (Yin et al., 2014). This renders SAR data a valuable source for detecting marineoil spills (Ma et al., 2021). However, the sea surface is subject to several natural phenomena, including waves, ocean currents, and low winds, which give rise to uneven intensity and high noise levels in oil spill images (Zhu et al., 2021), which makes oil spill detection challenging. Accurately identifying oil spill areas can help us reduce the false alarm rate in marine oil spill monitoring.In addition, failure to promptly detect oil spill areas allows the oil to disperse through ocean currents, resulting in widespread marine pollution and causing significant adverse effects on the marine environment (Simecek-Beatty & Lehr, 2017). Consequently, there is a need for high standards of real-time accuracy in marine oil spill detection. How to effectively reduce computational resource consumption and accelerate detection speed while ensuring the accuracy of oil spill detection has become a key research focus in this field.

There are three primary research directions for detecting marine oil spills using SAR remote sensing images: traditional threshold segmentation methods, machine learning methods, and deep learning methods (Keramea et al., 2021). The traditional threshold segmentation method focuses on determining an appropriate gray threshold, usually based on the gray histogram of the image. Yu et al. (2017) proposed an adaptive mechanism using the Otsu method, which combines edge detection with threshold segmentation to identify and extract the location and extent of oil spills in remote sensing images. Li et al. (2019) utilized a sliding window method that selects the optimal segmentation threshold based on the maximum entropy within the window. This approach effectively filters out small regions generated during segmentation and merges adjacent areas according to their distance. The correct detection rate for this method was 86.61%. With the increasing popularity of traditional neural networks and machine learning algorithms, Kim et al. (2018) leveraged Terra-SAR dual-polarization data to extract polarization parameters and optimize the input layer of an artificial neural network for analyzing oil spill regions in SAR images. Meanwhile, Magrì et al. (2021) selected the most relevant features and used them in a support vector machine (SVM) classifier to successfully identify and classify marine oil spill areas. However, these methods often require manual feature recognition design, which can be time-consuming and lacks scalability. They also heavily rely on threshold settings or model hyperparameter configurations, which can be subjective and uncertain (Dong et al., 2023). Therefore, we currently focus our research on deep learning methods. Recent advances in imaging equipment, computer hardware, and software indicate that deep learning techniques have considerable potential to improve the efficiency and accuracy of marine oil spill detection. The object detection method based on deep learning can automatically extract and learn complex high-level features (Xu et al., 2023). Researchers have applied some deep learning methods to detect oil spills at sea. XIONG & Meng (2019) designed a densely connected network model using DenseNet convolutional neural networks. This model extracts multi-scale features from images, improving the ability to capture fine features and enhancing image recognition accuracy. We input the oil spill SAR image into the CNN network. The CNN model extracts the SAR image's features, which improves the model's detection accuracy but requires high memory and computing resources. Huang et al. (2022) proposed a SAR oil spill detection algorithm based on Faster R-CNN. The CNN model was used to extract SAR image features, input the feature information into the Region Proposal Network (RPN), and generate a prediction boundary box, which could realize fast end-to-end oil spill detection. However, the detection accuracy of this method is relatively low. Zhu et al. (2023) proposed CBD-Net, an oil spill detection network based on scenario and boundary supervision, to extract oil spill areas by integrating multi-scale features and to introduce the attention mechanism module scSE to enhance the internal consistency of oil spill areas. However, introducing an additional boundary supervision module reduces the detection speed.

Although the research methods discussed above have improved either accuracy or speed, they cannot simultaneously achieve both the required detection speed and accuracy for the model. For marine oil spill detection, in order to prevent further spread of the oil, detection accuracy must be high and meet real-time detection requirements. Based on the above problems, this paper proposes a lightweight marine oil spill detection model LSFE-YOLO based on the YOLOv8s algorithm. By introducing the lightweight FasterNet as the backbone feature extraction network and reducing the neck network width, we effectively reduce the model size and computation amount, accelerating the model's detection speed. We designed a new GN-LSC Head detection module and introduced GroupNorm normalization technology to obtain a more lightweight network while maintaining the model's detection performance. Adding an SE attention mechanism to the backbone network better solves the challenge of background information interference and improves the model's detection performance. We used EffectiveSE to improve MBConv convolution, integrating the original C2f module in the YOLOv8 model to form the C2f_MBE module. This strengthens the extraction of marine oil spill features. In addition, since there are few available public data sets for marine oil spill detection, this paper creates and marks a SAR image marine oil spill detection data set for evaluating the model's performance, which solves the problem of insufficient public data sets. The improved model is more lightweight, easy to deploy, and effectively solves the problem of balancing detection speed and accuracy.

2. Materials and Methods

2.1. Image acquisition

The SAR images we used were obtained from the Alaska Satellite Facility (ASF) platform. Publicly available oil spill data were obtained from the National Oceanic and Atmospheric Administration (NOAA), including information such as the geographic coordinates, date of occurrence, and image source of the oil spill. NOAA (www.ospo.noaa.gov/Products/ocean/) has published verified oil spill data from the Gulf of Mexico, the Pacific Ocean, the Atlantic Ocean, the Great Lakes, and international waters, and it uses Sentinel-1 satellites to gather information on various types of oil spills. The corresponding Sentinel-1 SAR images can be obtained from the ASF (https://search.asf.alaska.edu/) platform using information such as the geographical coordinates and occurrence dates of oil spill incidents. Sentinel-1 monitors the surface in all weather conditions and provides high-quality SAR images, making it an ideal data source for our research
(http://www.hydroshare.org/resource/ 7c54d14f1f2b48a4a2fdfdab12aa233f).

Because the Interferometric Wide Swath (IW) acquisition mode generates SAR images with high quality, positioning accuracy, and radiation resolution, we used Sentinel-1's Ground Range Detected (GRD) product to collect data in IW mode. The GRD product contains data with two polarization modes: VV and VH. On the ocean surface, oil films are relatively smooth and have low radar wave echo reflectivity. The VV polarization has higher echo reflectivity, and the oil film forms more apparent surface scattering characteristics. Therefore, VV polarization data is selected as the raw marine oil spill detection data. Figure 1a shows the VV polarization SAR image of the Gulf of Mexico on January 27, 2023, downloaded from the ASF platform in the Sentinel-1 GRD product IW acquisition mode.

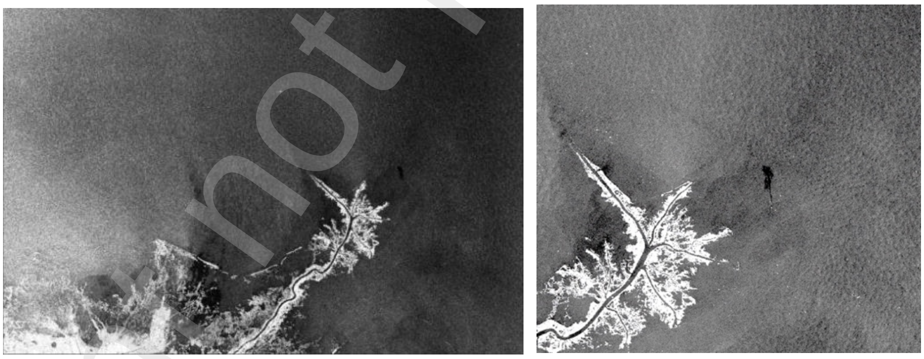
Existing oil spill detection datasets are complex to obtain and lack publicly available datasets. Therefore, this paper establishes an oil spill detection dataset to evaluate the proposed method's accuracy and address the scarcity of data sources for oil spill detection. A total of 306 raw SAR images of marine oil spills were collected.

2.2. Image preprocessing

We downloaded the GRD format file from the ASF platform to acquire comprehensive Sentinel-1 Synthetic Aperture Radar (SAR) data related to the NOAA oil spill incident. We collected a total of 1,335 SAR images depicting oil spills on the sea surface. Our initial phase involves preprocessing the acquired SAR data, transforming the raw data into high-quality inputs suitable for target detection tasks, thereby ensuring data reliability. We used SNAP (Sentinel Application Platform), a free and open-source software specifically designed for processing and analyzing Sentinel satellite data, to handle the collected SAR data.

Our preprocessing procedures included orbit correction, radiation calibration, geocoding, image filtering, decibelization, and cropping of images Oil spill scenarios are inherently complex and variable, necessitating a larger number of representative samples to enhance the effectiveness of target detection algorithms for identifying oil spills on the sea surface. To address this need, we employed a data augmentation technique to increase the size of our training dataset. To preserve the feature information contained within the synthetic aperture radar (SAR) images, we implemented only horizontal and vertical flipping, random contrast adjustment, and random scaling as methods to enhance the original dataset.

Following the data enhancement process, we successfully acquired a total of 2,670 Synthetic Aperture Radar (SAR) images. We used the LabelImg data annotation tool for manual calibration, and we subsequently divided the dataset into training, validation, and test sets in an 8:1:1 ratio. We documented the annotated labels in text format, establishing a robust foundation for our subsequent oil spill identification tasks. To illustrate this process, we present the oil spill in the Gulf of Mexico in Figure 1a, alongside a preprocessed image in Figure 1b. Image preprocessing significantly enhanced the brightness and textural characteristics of the oil spill region in the SAR image.



(a) (b)

Figure 1. SAR image. (**a**) Sentinel-1 SAR image; (**b**) The preprocessed SAR image

2.3. The YOLOv8 network architecture

The YOLO network has a simple structure and fast detection speed, making it a representative single-stage target detection algorithm with broad application prospects. Glenn Jocher proposed the YOLOv8 algorithm,which follows the same principles as the YOLOv3 and YOLOv5 algorithms. The YOLOv8 network structure is shown in Figure 2. Compared to previous versions, YOLOv8 introduces a new architecture that further improves the model's performance and flexibility. The YOLOv8 network structure comprises three main components: the backbone, neck, and head networks.

The data preprocessing strategy of YOLOv8 continues that of YOLOv5, mainly using four enhancement techniques: Mosaic augmentation, mixup, spatial perturbation, and color perturbation(Chen et al., 2023). These methods improve the model's generalization ability and robustness. The backbone network primarily extracts information from images and includes the C2f module, SPPF module, and Conv_BN_SiLU module. Compared to the C3 module in YOLOv5, the C2f module in YOLOv8 incorporates additional branches. This enables richer gradient flow information while maintaining a lightweight structure.The SPPF module pools input feature maps at different scales, extracting and fusing multi-scale features to enhance the model's feature extraction capability. The Conv_BN_SiLU module, which consists of convolutional operations, batch normalization, and the SiLU activation function, reduces the model's parameters and accelerates convergence. The Neck part primarily functions in feature fusion, constructing YOLO's feature pyramid through FPN and PAN structures to fully integrate multi-scale information. The head part mainly retrieves category and location information for the detection targets. Decoupled head and anchor-free strategies separate the classification and detection tasks, improving model performance and flexibility. YOLOv8 uses BCE Loss for classification, DFL Loss, and CloU Loss for regression. This enables a more precise determination of the target object's actual bounding box. Compared to previous YOLO algorithms, YOLOv8 achieves higher accuracy and enhanced scalability.

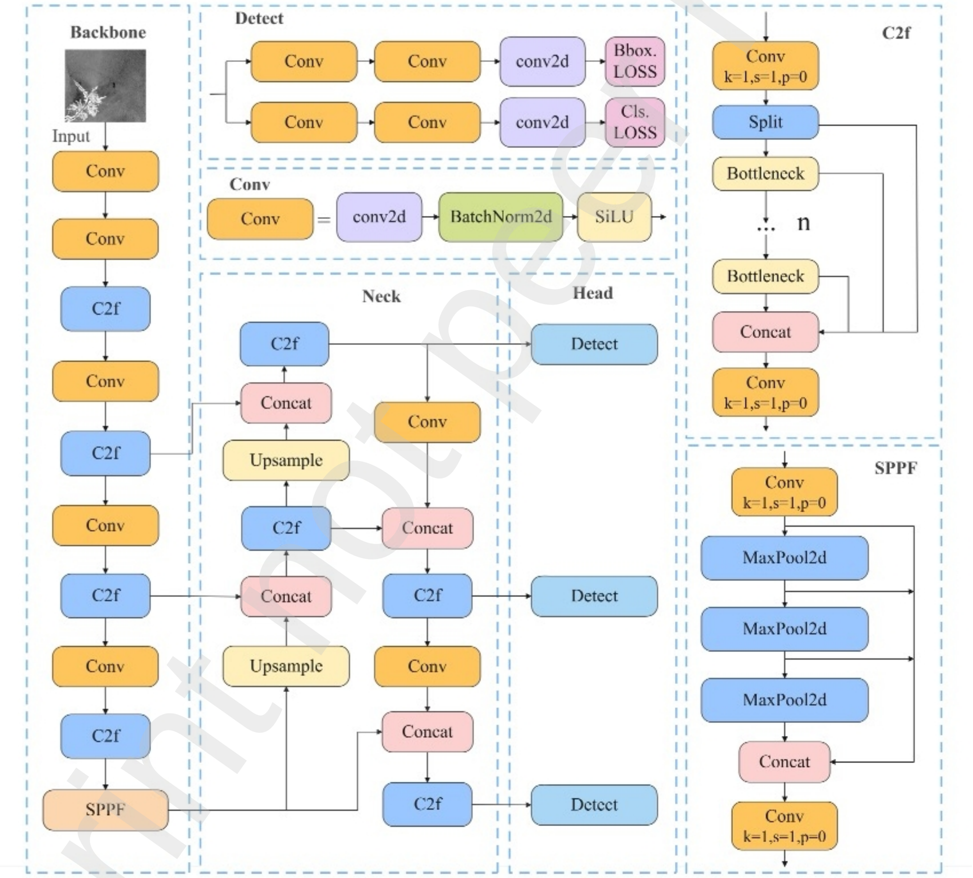


Figure 2. Network structure of YOLOv8 (Yang et al. 2023)

2.4. The proposed LSFE-YOLO network architecture

In response to the practical demand for marine oil spills and to cope with the challenges of high background interference, significant changes in target scale, and high requirements for real-time performance, as well as to reduce the number of model parameters, calculation volume and improve detection speed while maintaining model detection accuracy, we propose a model based on YOLOv8s improvements for marine oil spill, referred to as LSFE-YOLO. This model aims to enhance the performance and efficiency of marine oil spills. We present an improvement to the YOLOv8 model in the following aspects. The network structure of the improved LSFE-YOLO is illustrated in Figure 3.

2.4.1. FasterNet

FasterNet represents an innovative neural network architecture that demonstrates superior processing speeds compared to other neural networks across various devices, while ensuring high accuracy in numerous visual tasks. This architecture exhibits a reduction in latency and an enhancement in throughput.In evaluations conducted on ImageNet-1k, FasterNet outperformed MobileViT-XXS by factors of 3.1 times on both GPUs and CPUs, and 2.5 times on ARM processors, all while achieving a 2.9% accuracy improvement (Lu 2024). Consequently, we selected FasterNet as the backbone network for YOLOv8 to enhance the model's detection performance.

The most widely utilized lightweight networks, such as MobileNet, ShuffleNet, and GhostNet, utilize depthwise convolution (DWConv), group convolution (GConv), and depthwise separable convolution (DSC) to extract spatial features, thereby reducing model complexity and computational demands (Wang et al. 2023). However, minimizing floating point operations (FLOPs) often leads to increased memory access. Furthermore, these networks typically incorporate additional data processing operations, including concatenation, shuffling, and pooling, which extend runtime. As a result, these neural networks frequently do not achieve sufficient speed for certain applications.

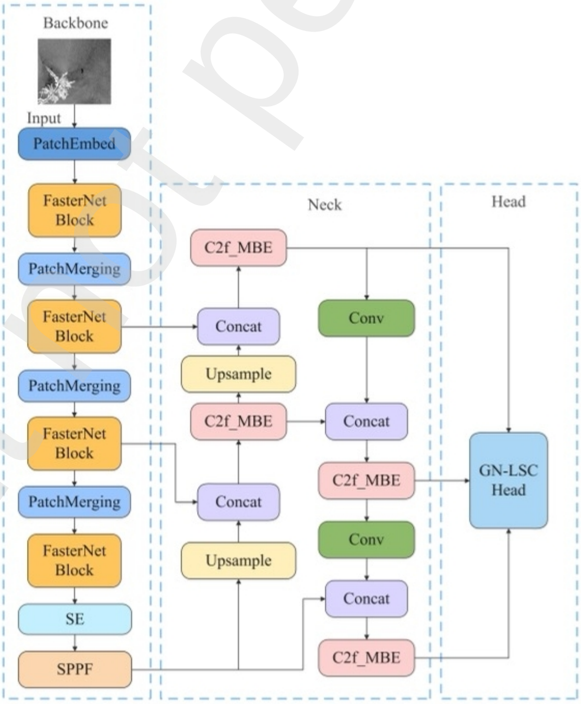


Figure 3. Network structure of LSFE-YOLO

The FasterNet framework utilizes a technique known as partial convolution (PConv), which significantly reduces computational redundancy and memory access while improving the efficiency of spatial feature extraction. As depicted in Figure 4, PConv leverages the inherent redundancy within the feature map by applying regular convolution only to a subset of the input channels, leaving the remaining channels unaffected. Typically, for continuous or regular memory access, either the first or last consecutive Cp channels are chosen as the computational representation of the entire feature map. As a result, PConv markedly decreases both the number of parameters and computational load, thereby improving the accuracy and speed of detection in convolutional neural network models.

The PConv calculation method is illustrated in Figure 3. A total of Cp consecutive channels are selected as the computational representation of the entire feature map. When Cp is taken as 1/4 of the total, the PConv FLOPs amount to only 1/16 of those required for regular convolution. Additionally, PConv entails relatively minor memory access, utilizing approximately 1/4 of the memory needed for regular convolution. The formulas for calculating the PConv FLOPs and memory access are as follows:

$$F = h \times w \times k^2 \times c_p^2 \tag{1}$$

$$M = h \times w \times 2c_p + k^2 \times c_p^2 \approx h \times w \times 2c_p \tag{2}$$

Where h represents the height of the feature map, w represents the width of the feature map, and k represents the kernel size. Furthermore, F denotes the floating point operations (FLOPs) of PConv, while M represents the memory access of PConv.

Furthermore, we incorporated Pointwise Convolution (PWConv) on top of PConv to ensure the comprehensive and effective use of information from all channels. We used the FasterNet block as the main feature extraction module for each stage of the FasterNet network. Each FasterNet block comprises a PConv layer and two subsequent PWConv layers, forming an inverted residual structure. We added normalization and activation layers after the intermediate layer to maintain feature diversity and reduce latency. The FasterNet block structure is illustrated in Figure 5. Consequently, compared with the original backbone network of YOLOv8, the FasterNet network exhibits reduced computational complexity and enhanced detection accuracy and speed.

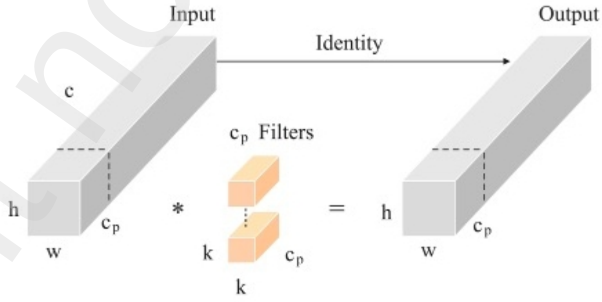


Figure 4. Partial Convolution

Considering the points mentioned above, we observe that the YOLOv8s neck network often uses C2fand Conv modules, which significantly affect memory and cache usage. These two modules feature convolutional kernels that produce numerous feature map channels, resulting in many parameters and computations. However, the network's depth is closely tied to the model's expressive power, and further reducing its depth is not feasible. As a solution, we decreased the network's width by 25% by minimizing the number of convolutional kernels in both the C2f and Conv modules within the neck network. This adjustment greatly lowers the number of network parameters and enhances the model's operational speed. These two modules feature convolutional kernels that produce numerous feature map channels, resulting in many parameters and computations. However, the network's depth is closely tied to the model's expressive power, and we find further reducing its depth is not feasible. As a solution, the width of the network was decreased by 25% by minimizing the number of convolutional kernels in both the C2f and Conv modules within the neck network. This adjustment greatly lowers the number of network parameters and enhances the model's operational speed.

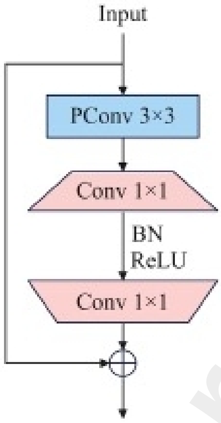


Figure 5. FasterNet Block structure diagram

2.4.2. GN-LSC Head

The YOLOv8 algorithm is designed to identify and localize target objects using a specialized detection head. This detection head features a decoupled structure that separates the classification and bounding box regression tasks into two independent network branches. Although this model offers enhanced flexibility and detection accuracy, each branch of the decoupled head consists of two one-dimensional convolutions and one two-dimensional convolution. This results in an excessive number of parameters and significant redundant calculations in the detection head, which makes it less suitable for embedded devices with limited computing resources. To address this issue, we propose a detection head with a shared convolutional structure, enhanced by GroupNorm normalization technology. This approach aims to reduce computational burdens while improving the positioning and classification performance of the detection head.

Normalization techniques play a crucial role in enhancing model performance and stability. Figure 6 illustrates the normalization methods BatchNorm and GroupNorm, where N represents the batch dimension, H, W represents the feature map size, and C denotes the number of channels. BatchNorm normalizes based on the batch dimension and depends highly on batch size;as the batch size decreases, the error associated with BatchNorm increases rapidly.In contrast, GroupNorm divides the channels into groups and normalizes the features within each group. GroupNorm's calculations are independent of the batch size, resulting in stable accuracy across different batch sizes. Therefore, we propose using GroupNorm in place of BatchNorm to enhance the model's detection performance.

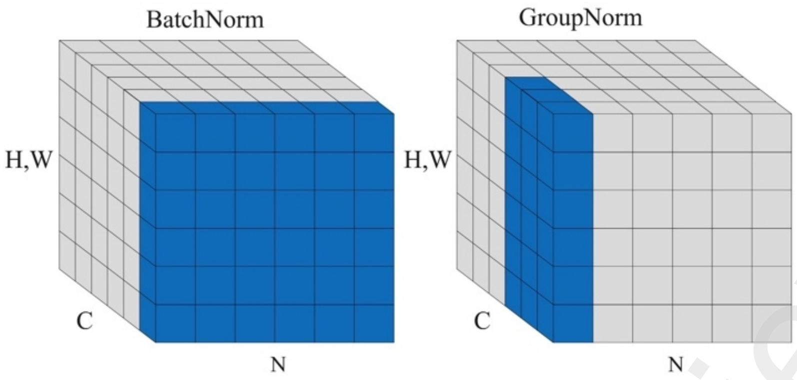


Figure 6. FasterNet Block structure diagram

Figure 6 illustrates the GroupNorm calculation process, where the same mean and standard deviation normalize the blue-marked pixels. Designing the detection head with a shared convolutional structure can significantly reduce the number of parameters and computational redundancy, making the head structure lighter. In order to enhance feature extraction capabilities, two 3x3 convolutions are connected in series. In addition, while using the shared convolutional structure, to deal with the problem that the target scales detected by each detection Head are inconsistent, the Scale layer is used to scale the features, and the final GN-LSC HEAD detection module is obtained, thus improving the computational efficiency. The structure of the detection head is shown in Figure 7. Additionally, while using the shared convolutional structure, we use the Scale layer to address inconsistent target scales detected by each detection head and to scale the features. This obtains the final GN-LSC HEAD detection module, thus improving computational efficiency.The structure of the detection head is shown in Figure 7.

The design of the detection head for a shared convolutional structure can significantly reduce the number of parameters and computational redundancy, resulting in a lighter head structure. Two 3x3 convolutional layers are connected in series to enhance feature extraction capabilities. Furthermore, a Scale layer is implemented to adjust the features to address the issue of inconsistent target scales detected by each detection head while using the shared convolutional structure. This leads to the creation of the final GN-LSC Head detection module, thereby improving computational efficiency. The structure of the detection head is illustrated in Figure 7.

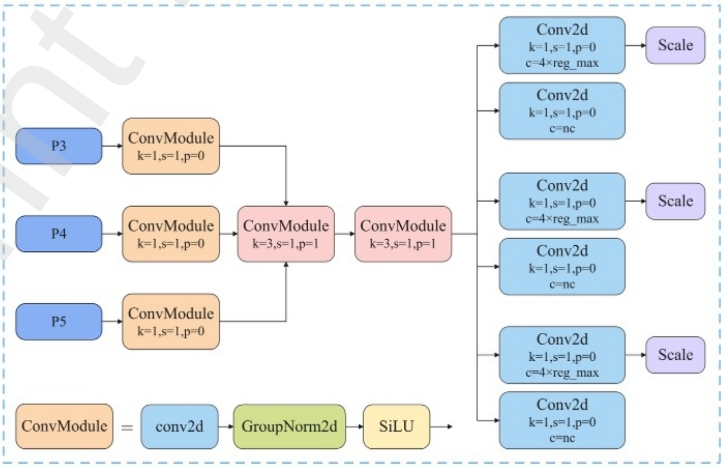


Figure 7. GN-LSC Head Structure Diagram

The GN-LSC Head detection module demonstrates a significant reduction in both the number of parameters and computational redundancy within the model. To enhance the performance of this model further, the GroupNorm normalization method has been implemented in place of BatchNorm.To evaluate GroupNorm's efficacy when integrated into the detection head, we conducted comparative experiments. Table 1 presents these, contrasting the GN-LSC Head module's performance using both normalization techniques. The data in the table indicate that the GN-LSC Head detection module, using GroupNorm, improves three critical performance metrics: precision (P), recall (R), and mean Average Precision (mAP). These increased by 0.4%, 5.1%, and 1.9%, respectively, compared to the conventional BatchNorm approach. Furthermore, incorporating the improved detection head into the YOLOv8 model not only reduces computational costs relative to the original model but also demonstrates superior detection accuracy.

Table 1. GN-LSC Head module validity experiments

Model	P/%	R/%	mAP/%	Model size/MB	GFLOPs
YOLOv8s	93.1	91.5	95.1	22.6	28.4
BatchNorm	93.4	87.5	94.2	18.6	21.4
GroupNorm	93.8	92.6	96.1	18.6	21.4

2.4.3. SENet

Complex background information in images of oil spills on the sea surface can decline target detection performance. We introduce the SE module to enhance target detection performance in images of oil spills on the sea surface. The SE module effectively improves the recognition ability of targets in images by adaptively recalibrating the feature response between channels, enhancing the weight of essential features, and suppressing irrelevant features.

Embedding the SE module before the SPPF9 (Spatial Pyramid Pooling-Fast) module in the YOLOv8 backbone network means that we screened and optimized the feature map input to the SPPF module. The SPPF module is a component of the YOLOv8 backbone that enhances the receptive field of the network by applying multiple pooling operations with different kernel sizes to the same input feature map. The SE module prioritizes key features by adaptively recalibrating channel-wise feature responses, thus strengthening informative features while suppressing less relevant ones. The structure of the SE module is shown in Figure 8. This is achieved through two operations: the squeeze and excitation operations. In the squeeze operation, we use the global average pooling operation to aggregate the spatial information of each channel of the input feature map into a scalar, effectively reducing the spatial dimension of the features and capturing global features. In the excitation operation, the aggregated scalar generates each channel's weights through a fully connected layer and an activation function. We use these weights to re-calibrate the channel features so that the model can focus more on essential features.

This reduces the interference of irrelevant features and enables the subsequent spatial pyramid pooling process to focus more on essential features, thereby improving the model's accuracy in detecting targets in complex backgrounds. Therefore, the SE module effectively enhances the model's overall performance at a relatively small additional computational cost.

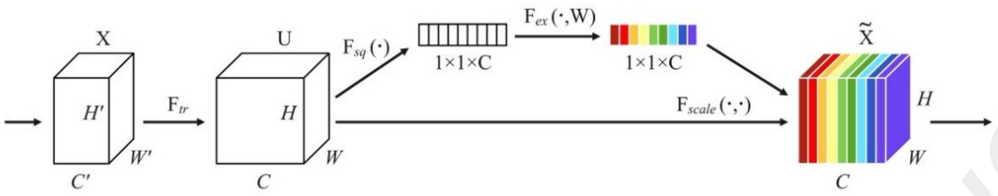


Figure 8. A Squeeze-and-Excitation block

Where F_{tr} indicates that the input feature graph X is generated by convolution operation to generate feature graph U, and the feature graph U is globally average pooled by extrusion operation F_{sq} to generate channel statistic z, where the c element of z is calculated as follows:

$$z_c = \mathbf{F}_{sq}(\mathbf{u}_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \tag{7}$$

F_{ex} represents the excitation operation. Statistic z passes the first fully connected layer and ReLU activation function and then passes the second fully connected layer and Sigmoid activation function to generate the weight s for recalculating the channel. The calculation formula is as follows:

$$\mathbf{s} = \mathbf{F}_{ex}(\mathbf{z}, \mathbf{W}) = \sigma(g(\mathbf{z}, \mathbf{W})) = \sigma\left(\mathbf{W}_2 \delta\left(\mathbf{W}_1 \mathbf{z}\right)\right) \tag{8}$$

F_{scale} represents the scaling operation, which multiplies the generated weight s with the corresponding channel of the feature graph U to carry out feature fusion and generate the final feature graph $\tilde{X}$.

2.4.4. C2f_MBE

The C2f module is an innovative design in YOLOv8, which refers to the design ideas of ELAN in YOLOv7 (Yu 2024). The C2f structure has more residual connections and a richer gradient flow but still lacks attention to important target information during feature extraction (Lee 2020). In addition, the scale of the oil spill area in the sea surface oil spill image varies greatly, which poses a challenge for accurate detection. Large-scale oil spills may dominate the image, while smaller spills can be easily overlooked, especially in complex marine backgrounds. Therefore,we introduce the Mobile Inverted Bottleneck Convolution (MBConv) to improve the C2f module. The MBConv inverted residual structure better preserves information integrity while capturing more feature information at different scales and learning more complex and richer features. This improves the detection performance of objects at various scales, which is particularly beneficial for identifying oil spills of varying sizes. The SE module integrated into MBConv allows the network to pay more attention to important feature information, better use and strengthen valuable features, and enhance the network's expressive power, which help the model more effectively distinguish between small, localized spills and larger, diffuse ones. Moreover, the lightweight nature of MBConv ensures that these improvements come with minimal overhead, making it well-suited for real-time or resource-constrained oil spill monitoring systems. The processing of MBConv is shown below:

Step1, uses a 1x1 convolution to upsample.

Step2, a Depthwise Convolution (DWConv) extracts features, then introduces the SE module.

Step3, uses a 1x1 convolution to downsample and output at the Dropout layer, as Figure 9a shows.

The proposed LSFE-YOLO model replaces the Bottleneck of the C2f module in the neck network with MBConv. The improved C2f module can better capture the interrelationships between features and enhance the extraction of features in the oil spill area, enabling the model to obtain more comprehensive oil spill feature information. After improving the neck network, the detection ability of the model has also been significantly improved. Still, at the same time, the number of parameters of the model has increased, and the detection speed has been reduced. The speed and accuracy of oil spill detection are equally crucial for the real-time oil spill detection task. Therefore, to maintain the model's accuracy while minimizing the loss of model speed, the MBConv original SE module is replaced with the EffectiveSE (Liu 2020) module. MBEConv is the im-proved MBConv, and the structure is shown in Figure 9b.

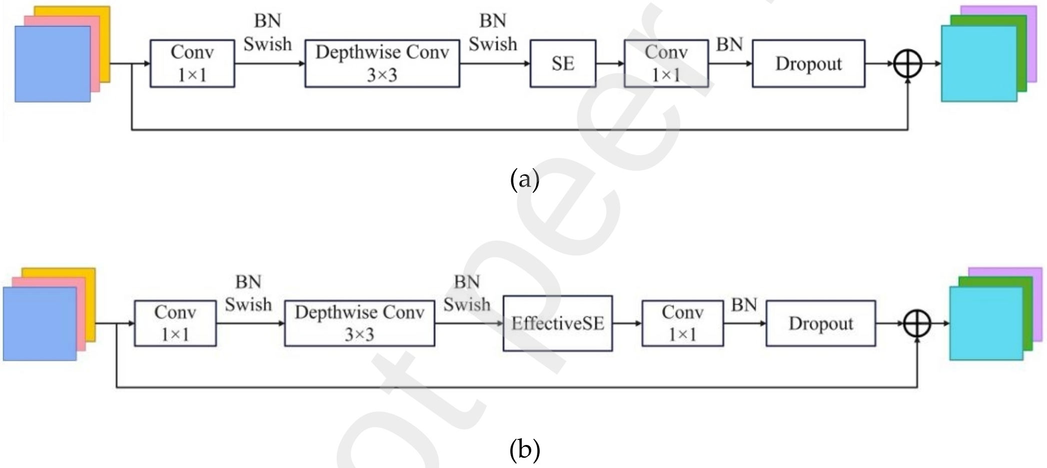


Figure 9. MBConv structure diagram. (a) The unimproved MBConv structure; (b) The MBConv structure of the EffectiveSE module is introduced

The EffectiveSE module improves the SE module; its core components are the same as those of the SE module: extrusion and excitation operations. The primary distinction between the EffectiveSE module and the SE module is that the SE module comprises two fully connected layers. In contrast, the EffectiveSE module employs a single fully connected layer, thereby reducing the number of parameters in the model. The two fully connected layers of the SE module require reducing and expanding the channel dimension. However, the EffectiveSE module circumvents the necessity for channel dimension reduction, thereby retaining a greater quantity of channel information. This results in an enhancement of the model's capacity to represent features, allowing the network to preserve and emphasize more relevant features without incurring extra computational cost. Consequently, the EffectiveSE module enhances the efficiency of model detection while maintaining the advantages of the SE module.These improvements make it especially suitable for real-time oil spill detection tasks and deployment in edge devices or low-power environments, where computational resources are limited and inference speed is critical. The novel C2f_MBE module structure is illustrated in Figure 10.

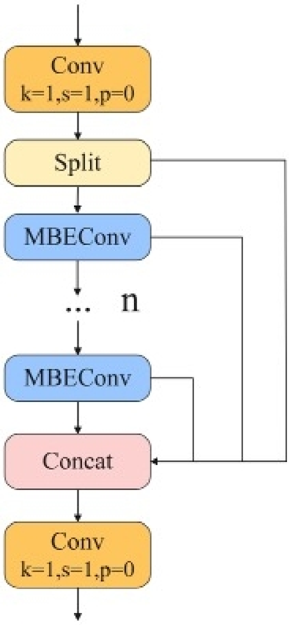


Figure 10. C2f_MBE structure diagram

To mitigate the impact on detection speed, we implemented improvements in MBConv by replacing the original Squeeze-and-Excitation (SE) module with an EffectiveSE variant. To evaluate the proposed module's effectiveness, this study conducts three sets of comparative experiments involving the C2f module in the YOLOv8 model. These experiments compare the original module (M0), the C2f module using MBConv (M1), and the C2f module employing the improved version of MBConv (M2), followed by a thorough analysis of the experimental results from the three configurations.

As illustrated in Table 2, the detection speed of the M1 and M2 models is marginally lower than that of the original M0 model. However, both the F1 and mAP values for the M1 and M2 models surpass those of the M0 model, demonstrating that the implementation of MBConv effectively enhances the model's detection accuracy concerning sea surface oil spills. Notably, the M2 model not only sustains the detection accuracy achieved by the M1 model but also demonstrates a slight improvement, all while increasing the detection speed by 2.9 FPS. This outcome substantiates our assertion that the EffectiveSE module is more efficient than the SE module.

Table 2. C2f_MBE module validity experiments

Model	F1/%	mAP/%	FPS
M0	92.3	95.1	96
M1	93.3	95.9	90.1
M2	93.7	96.2	93

2.5. Environmental configuration

We conducted all experiments in an identical experimental setting to guarantee their reliability in this paper. We used Ubuntu 20.04 as the operating system in the experiment, and we trained the model using a GPU based on the PyTorch deep learning framework.

We used an RTX 3090 (24GB) as the graphics processing unit (GPU) in the experiment. The PyTorch version is 1.10.0, Python version is 3.8, CUDA version is 11.3, and the central processing unit (CPU) is an Intel (R) Xeon (R) Platinum 8358P @ 2.60GHz. Table 3 presents the parameter settings for experiment training.

Table 3. Experimental parameter setting.

parameters	value
epoch	300
batch size	32
patience	50
image_size	640
learning rate	0.01
optimizer	SGD
momentum	0.937
weight_decay	0.0005

2.6. Performance evaluation

We used multiple metrics to evaluate the algorithm's effectiveness. Precision (P), recall (R), and mean average precision (mAP) are commonly used metrics for evaluating model performance (Zhao et al., 2024). In addition, the F1 score is used to assess the improvement in the detection accuracy of the model. The larger the value, the better the detection accuracy of the model. Precision defines the proportion of true positives among all samples classified as positive. The recall rate defines the proportion of all actual positive samples correctly classified as positive.The F1 score is the harmonic mean of the precision and recall rates. The mAP represents the mean detection accuracy of all categories and reflects the model's overall detection performance. The P, R, F1 score and mAP are calculated as follows:

$$P = \frac{TP}{TP + FP} \tag{9}$$

$$R = \frac{TP}{TP + FN} \tag{10}$$

$$F1 = \frac{2 \times P \times R}{P + R} \tag{11}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{12}$$

TP indicates the number of positive samples correctly predicted by the model, *FP* suggests the number of positive samples incorrectly predicted by the model, *FN* shows the number of negative samples incorrectly predicted by the model, and *AP* is the detection accuracy of each category. We also measured the model's performance in terms of model size, model parameter quantity, Frames Per Second (FPS), and Giga Floating Point Operations per Second (GFLOPs). The smaller the model size, parameter quantity, and GFLOPs value, the higher the model's lightweight, the lower the complexity and computing power, and the lower the hardware performance requirements. A higher FPS value indicates better real-time performance and suggests faster detection speed for the model.

3. Results

We conducted the following experiments to verify the LSFE-YOLO model's superiority: comparing different lightweight networks based on YOLOv8s, comparative experiments between the SE attention mechanism and other spatial and channel attention mechanisms, ablation studies on the LSFE-YOLO model's improvements, and comparative experiments before and after the upgrades. Additionally, the enhanced model is compared with current mainstream object detection algorithms.

3.1. Comparisons of lightweight networks

To highlight the advantages of the proposed lightweight LSFE-YOLO network in maintaining high accuracy and fast detection speed, this section compares various lightweight network models based on the YOLOv8s network. We tested the original YOLOv8s model, the YOLOv8s model with EfficientViT as the backbone, the YOLOv8s model with StarNet as the backbone, the YOLOv8s model with ShuffleNetV2 as the backbone, and the improved LSFE-YOLO model. As shown in Fig. 11, the LSFE-YOLO model achieves the highest detection accuracy and the smallest number of parameters.

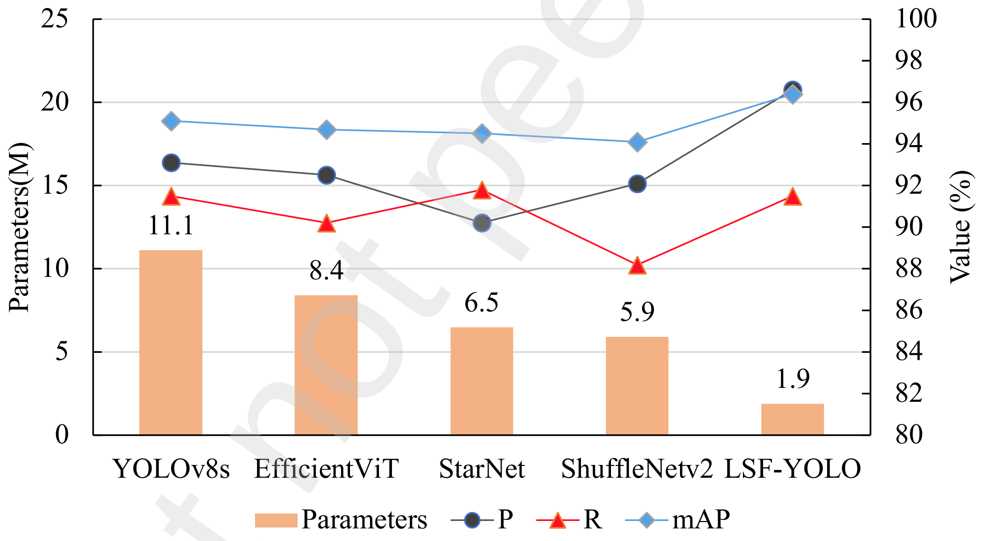


Figure 11. Comparison of accuracy and parameter number of different lightweight networks

As shown in Table 4, the detection results indicate that the proposed lightweight network significantly reduces the original model's size by 81.9%. ShuffleNetv2 follows with a 46.5% reduction, StarNet with 40.7%, and EfficientViT with 22.6%. The LSFE-YOLO model uses FasterNet as its backbone network, which reduces model size and accelerates detection speed. Compared to the original model, the LSFE-YOLO model decreases GFLOPs by 23.9 and achieves 116 FPS, the highest among the models we tested. ShuffleNetv2, StarNet, and EfficientViT reduce GFLOPs by 12.5, 11.1, and 8, respectively, compared to the original model. This illustrates that the LSFE-YOLO model has a lower computational load and complexity, with faster detection speed. Furthermore, the LSFE-YOLO model achieves a mAP of 96.4%, which is 1.3% higher than the original model, while the mAP for the other three networks falls below that of the original model. This validates that the LSFE-YOLO model is more lightweight and effectively balances detection accuracy and speed, meeting the demands of real-time detection.

Table 4. Detection results of different lightweight networks.

Model	mAP/%	Model size/MB	GFLOPs	FPS
YOLOv8s	95.1	22.6	28.4	96
YOLOv8s-EfficientViT	94.7	17.5	20.4	48
YOLOv8s-StarNet	94.5	13.4	17.3	102
YOLOv8s-ShuffleNetv2	94.1	12.1	15.9	108
LSFE-YOLO	96.4	4.1	4.5	116

3.2. *Ablation experiments*

To evaluate the impact of each improvement step on detection performance, we conducted ablation experiments on the innovative modules. Table 5 shows six schemes from S0 to SP representing different combinations of five improvement strategies. S0 is the baseline network without improvement strategies, which is equivalent to the YOLOv8s model, S1 builds on the S0 framework by incorporating the FasterNet structure to enhance the backbone network, S2 further modifies the network's width over S1, S3 introduces a lightweight detection head, known as the GN-LSC head, based on S2, and S4 integrates the Squeeze-and-Excitation (SE) attention mechanism into the backbone of S3, SP substitutes all original C2f modules within the S4 framework with the newly proposed C2f_MBE modules, thereby establishing the comprehensive LSFE-YOLO mode. The YOLOv8s model is the baseline, and we reconstructed the backbone network based on the FasterNet network. We adjusted the network width and replaced the original detection head with the newly designed lightweight GN-LSC head. These improvements significantly enhance the lightweight nature of the S0 model, effectively reducing model complexity and computational load. When we incorporate the FasterNet network and reduce network width, the resulting S2 model can reduce parameters by 77.5%, model size by 76.5%, and GFLOPs by 75%, along with a 26% increase in detection speed. By further introducing the GN-LSC head, the S3 model, compared to the S2 model, decreases parameters by 32%, model size by 32.1%, GFLOPs by 25.4%, and also achieves a slight increase in detection speed. The experimental results demonstrate that the S3 model exhibits the lowest parameters, smallest model size, and fastest detection speed.

Although the above improvement methods make the model more lightweight, they impact the model's detection accuracy. To enhance the model's detection accuracy, we introduce the SE module and replace the original C2f module in YOLOv8s with the C2f_MBE module. Incorporating the efficient channel attention mechanism SE enables the network to focus better on essential features and suppress unimportant ones.The SE module is relatively lightweight, with fewer parameters, maintaining model size and computational load while increasing the mAP from 94.7% in the S3 model to 95.3% in the S4 model and finally, replacing the C2f module with the C2f_MBE module results in the SP model, which achieves a 1.1% increase in mAP and a 15.1% reduction in GFLOPs, with a minimal sacrifice in model parameters and FPS compared to the S4 model. This demonstrates that the SP model provides more accurate oil spill localization, making sea surface oil spill detection more efficient.

Table 5. Detection results of the ablation experiment.

Model	FasterNet	network width	GN-LSC head	SE	C2f_MBE	mAP/%	Model size/MB	Parameters/M	GFLOPs	FPS
S0						95.1	22.6	11.1	28.4	96
S1	√					95.5	12.4	6.1	16.1	115
S2	√	√				94.9	5.3	2.5	7.1	121
S3	√	√	√			94.7	3.6	1.7	5.3	123
S4	√	√	√	√		95.3	3.6	1.7	5.3	120
SP	√	√	√	√	√	96.4	4.1	1.9	4.5	116

3.3. Experimental comparison of different attention mechanisms

To further verify the SE attention mechanism's effectiveness in the LSFE-YOLO model, this experiment compares it with the SimAM, ECA, and GlobalContext attention mechanisms. We integrated all four attention mechanisms into the LSFE-YOLO model's backbone network for experimental analysis. Table 6 summarizes the results. The precision (P), recall (R), and mean Average Precision (mAP) values for the SE attention mechanism in this study are the highest, recorded at 96.6%, 91.5%, and 96.4%, respectively. This evidence demonstrates that the model using the SE attention mechanism performs best in detection. While the SimAM and ECA attention mechanisms show low parameter counts, their detection accuracies are subpar. Therefore, we advise sacrificing some model parameters to enhance detection accuracy. We incorporate the SE attention mechanism into the model, achieving a balance between model parameters and detection accuracy.

Table 6. Comparison of different attention mechanisms.

Attention Mechanism	P/%	R/%	mAP/%	Parameters
SimAM	96.1	90	95.6	0
ECA	95.9	90	95.7	3
GC	95	91.5	96	2533
SE	96.6	91.5	96.4	1152

3.4. Comparison of different advanced detection algorithms

We assess the LSFE-YOLO detection model's effectiveness by comparing it to several of the most popular and advanced object detection methods of recent years, including Faster R-CNN, SSD, RT-DETR (Peng and Ji, 2023), and the YOLO series. Meanwhile, to ensure the evaluation process's integrity, we performed all experiments within a consistent experimental environment, using identical data partitioning, hyperparameter configurations, and training iterations,as shown in Table 7. Given the intricate nature of the sea surface environment, characterized by factors such as low wind conditions, ocean currents, and the presence of biological oil films, the potential for misidentification of oil spills at sea may increase, resulting in a heightened rate of false detections. This phenomenon adversely affects the accuracy of oil spill detection efforts. As the accompanying table illustrates, the enhancedLSFE-YOLO model presented in this study achieved a Precision (P) index of 96.6%, representing a 3.5% improvement over the original YOLOv8s model. A higher Precision value correlates with a reduced rate of erroneous predictions, particularly concerning categories such as low wind, leading to a greater proportion of accurately identified positive samples. This indicates that the improved model exhibits a notable decrease in the false detection rate. Furthermore, when compared to other leading one-stage and two-stage algorithms, the model proposed in this study exhibits the highest Precision value, thereby demonstrating the effectiveness of the enhancements implemented.

As the table presents, the two-stage target detection model, Faster R-CNN, achieves a detection accuracy of only 89.8%, with an FPS rate of 30. This performance is inadequate for the necessary detection accuracy and, more importantly, does not satisfy the real-time requirements for detecting oil spills on the surface of the sea. Although the SSD and YOLOv3 models demonstrate improved detection accuracy compared to the Faster R-CNN model, the overall detection rate for oil spills remains suboptimal, and the model sizes are substantial. In contrast to the two-stage algorithm, the RT-DETR-L, YOLOv5m, YOLOv8m, YOLOv10m, and YOLOv8s-spd-eca-ad models exhibit enhanced detection outcomes. Nonetheless, these models impose a significant computational burden, and their detection speeds do not fulfill the criteria for real-time detection.In contrast to the two-stage algorithm, the RT-DETR-L, YOLOv5m, YOLOv8m, YOLOv10m, and YOLOv8s-spd-eca-ad models exhibit enhanced detection outcomes. Nonetheless, these models impose a significant computational burden, and their detection speeds do not fulfill the criteria for real-time detection.

In comparison, the YOLOv5s, YOLOv8s,YOLOv10s, and YOLOv8-YP(Lv, 2020) models exhibit improved detection speeds, reduced model sizes, and lower computational complexity. However, there is still a need for further enhancements in detection accuracy. In contrast, our proposed LSFE-YOLO model demonstrates superior performance in both mean Average Precision (mAP) and F1 metrics, achieving scores of 96.4% and 94%, respectively. Additionally, this model outperforms others in four lightweight metrics: parameters, model size, GFLOPs, and FPS. When compared to the original YOLOv8s model, LSFE-YOLO shows a significant reduction in complexity, with a model size of 4.1 MB, a calculation amount of 4.5 MB, and the number of parameters reduced to 1.9 M. These figures indicate decreases of 81.9%, 84.2%, and 82.9%, respectively, compared to the original model. The detection speed reaches 116 frames per second, marking a 20.8% increase. The lightweight sea surface oil spill detection model, LSFE-YOLO, meets real-time detection requirements while achieving optimal accuracy in complex environments. It successfully balances model efficiency, detection accuracy, and speed, providing an effective solution.

Table 7. Comparison of different advanced detection algorithms.

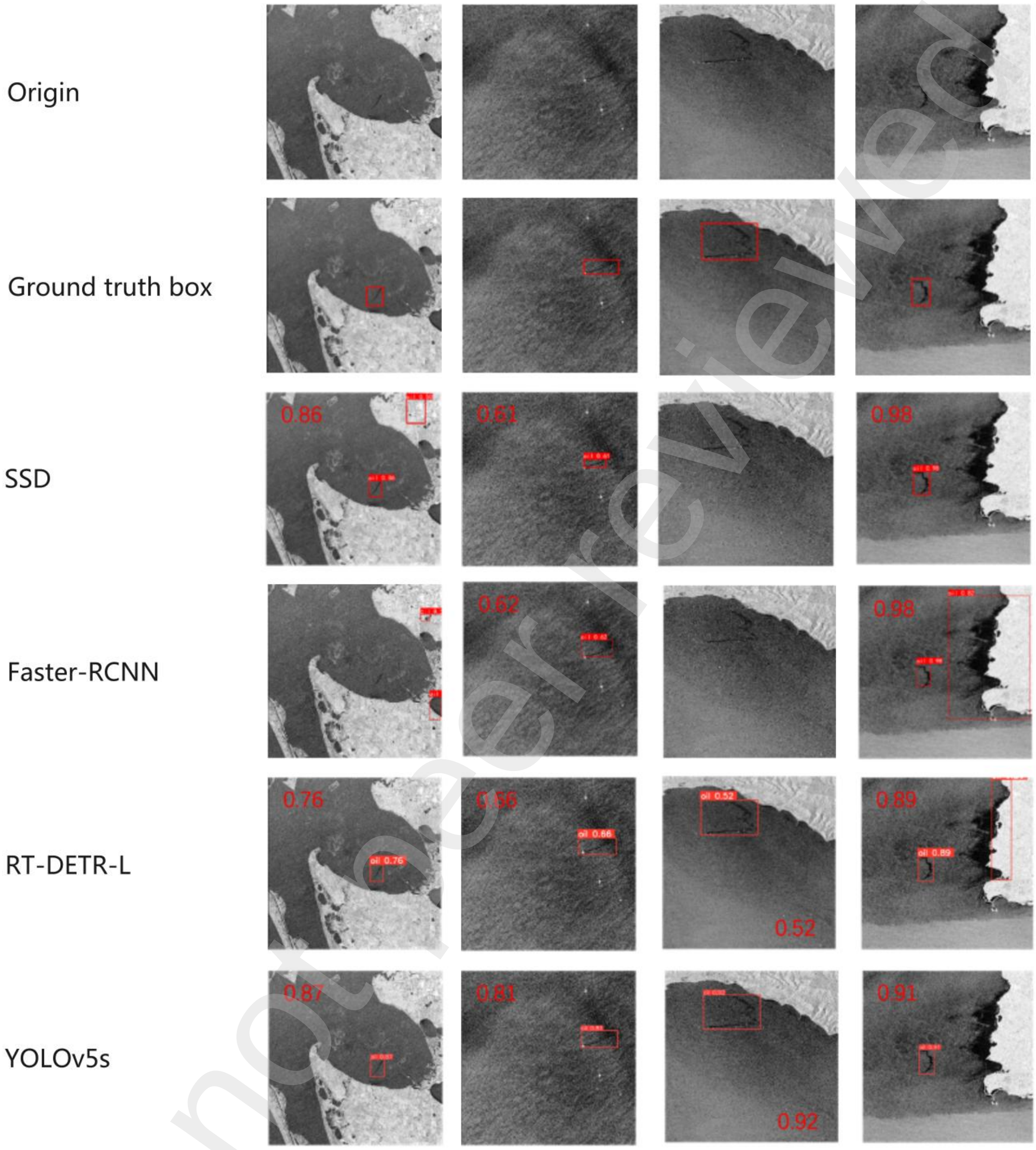
Model	P/%	R/%	mAP/%	F1/%	Model size/MB	Parameters/M	GFLOPs	FPS
Faster RCNN	55.1	92.4	89.8	69	108.2	136.7	369.7	30
SSD	92.1	88.2	93.5	90	90.6	23.6	174.8	91
RT-DETR-L	92.8	90.5	94.5	91.6	59.1	28.4	100.6	48
YOLOv3	94.6	85.1	93.8	89.6	207.8	103.6	282.2	49
YOLOv5s	94.7	91.8	94.4	93.2	18.6	9.1	23.8	103
YOLOv5m	93.4	91	95.4	92.2	50.5	25.1	64.2	80
YOLOv8s	93.1	91.5	95.1	92.3	22.6	11.1	28.4	96
YOLOv8m	93.2	91	95.5	92.1	52.1	25.8	78.7	80
YOLOv10s	90	89.1	93.7	89.5	16.6	7.2	24.1	104
YOLOv10m	92.2	89.1	93.6	90.6	33.5	15.3	58.9	83

YOLOv8s-spd-eca-ad[20]	94.3	90.3	95.4	92.3	21.2	10.3	52.1	76
YOLOv8-YP[25]	93.5	89.4	93.9	91.4	10.9	5.3	12.7	88
LSFE-YOLO	96.6	91.5	**96.4**	**94**	**4.1**	**1.9**	**4.5**	**116**

3.5. Detection effect display

To provide a more intuitive demonstration of the detection effect of the improved algorithm presented, four randomly selected images with oil stains are compared. As Figure 12 illustrates, we present the detection outcomes of LSFE-YOLO and the other comparative models.

The efficacy of oil stain detection varies among different models. Notably, only the YOLOv5s, YOLOv8s-spd-eca-ad, and the LSFE-YOLO model which we have proposed can detect oil stains in all four images without false positives. However, the proposed model exhibits superior detection accuracy and faster detection speed. Among the other comparison models, the Faster-RCNN model only detects two oil stains, and the SSD model detects three oil stains but also exhibits false positives. The remaining network models, RT-DETR-L, YOLOv8s, YOLOv8m, and YOLOv10s, identified four oil slicks. However, the detection results indicate that the efficacy of oil slick detection in scenes with elevated background noise is suboptimal, and we observed a prevalence of false positives. In conclusion, the LSFE-YOLO model we have presented exhibits enhanced detection precision while maintaining a commendable detection speed, facilitating more efficient marine oil spill detection in complex aquatic environments.



Origin
Ground truth box
SSD
0.86
0.61
0.98
Faster-RCNN
0.62
0.98
RT-DETR-L
0.76
oil 0.76
0.66
oil 0.66
oil 0.52
0.52
0.89
oil 0.89
YOLOv5s
0.87
0.81
0.92
0.91

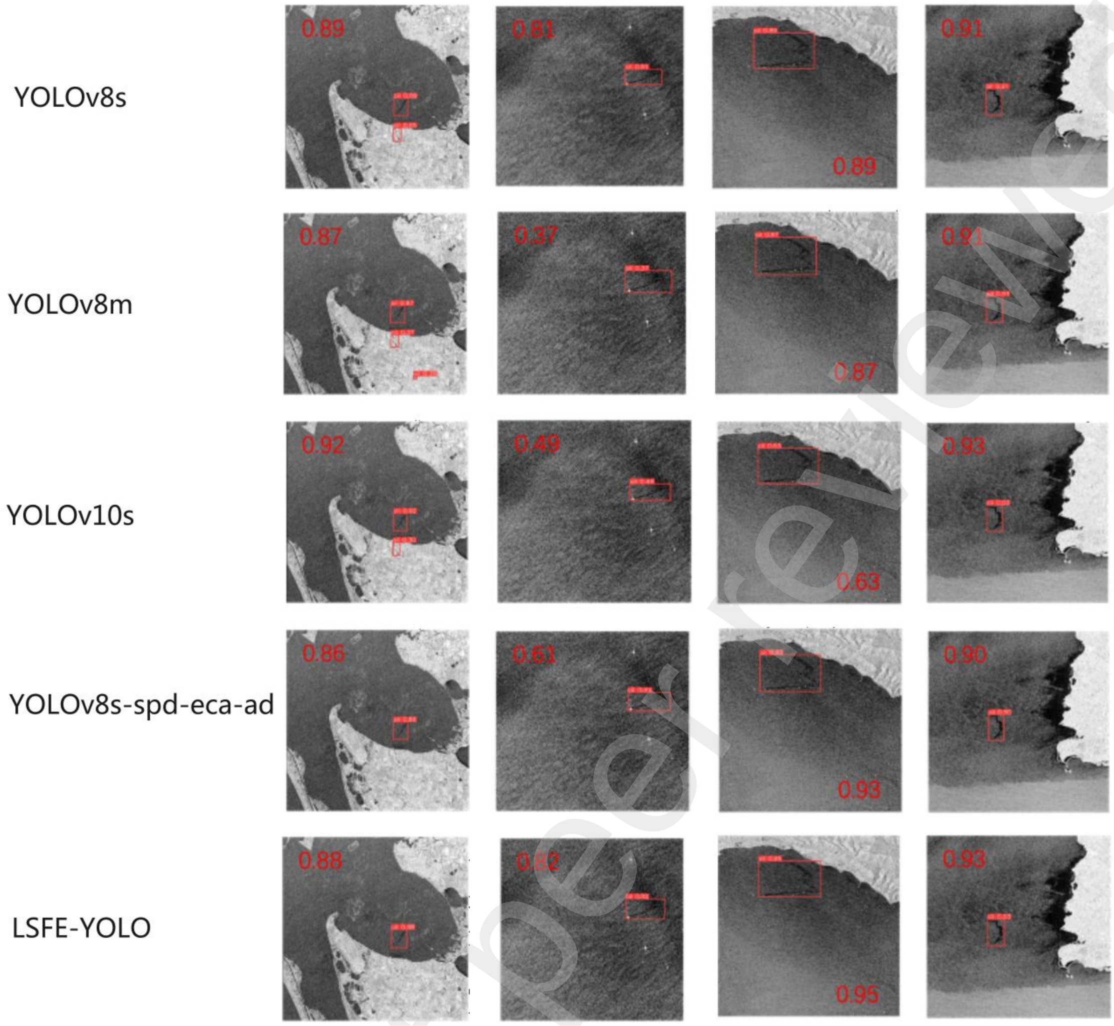


Figure 12. Comparison of detection effects of different models.

3.6. Generalization experiment

To further verify the model's generalization ability and analyze its detection results for multi-scale targets in complex backgrounds, we employ the public dataset GC10-DET[26] for testing. The dataset comprises steel plate surface defects collected by Tianjin University in an actual industrial scenario. It includes ten types of surface defects: punching (Pu), weld (Wl), crescent moon crescent bend (Cg), water spot (Ws), oil spot (Os), indentation (Rp), fold (Cr), silk spot (Ss), waist fold (Wf), and foreign body indentation (In). The ten surface defects are as follows: weld (Wl), oil spot (Os), silk spot (Ss), foreign body indentation (In), indentation (Rp), fold (Cr), and waist fold (Wf). These are the overall detection results for all defect types.

Table 8 presents the experimental results. The test results once again demonstrate the superiority of the LSFE-YOLO model. While reducing the number of model parameters and improving model detection speed, the R, F1 scores, and mAP evaluation metrics are superior to those of YOLOv8s,verifying that our proposed model has enhanced generalization capabilities.

Table 8. GC10-DET dataset generalization ability test

Model	Parameters/M	FPS	Defect type	P/%	R/%	F1/%	mAP/%
YOLOv8s	11.1	96	Pu	95.4	100	97.6	99.5
			WI	76.5	100	86.7	94.3
			Cg	74.3	92.3	82.3	89.4
			Ws	74	73.8	73.9	75.9
			Os	70.9	59	64.4	66.6
			Ss	61.6	52.9	56.9	56.6
			In	60.1	25.1	35.4	34.5
			Rp	61	33.3	43.1	41.9
			Cr	63.6	30.3	41	44
			Wf	76	90.6	82.7	90.5
			all	71.3	65.7	68.4	69.3
LSFE-YOLO	1.9	116	Pu	82.8	100	90.6	99.5
			WI	60.2	96.6	74.2	84.1
			Cg	73.5	92.3	81.8	90.4
			Ws	64.9	74.1	69.2	79.3
			Os	74	69.4	71.6	80.4
			Ss	62.8	63.8	63.3	63.2
			In	51.8	28.6	36.9	29.3
			Rp	57.6	30.7	40.1	37.9
			Cr	100	29	45	42.3
			Wf	78.8	92.9	85.3	87.5
			all	70.6（-0.7）	67.7（+2）	69.1（+0.7）	69.4（+0.1）

4. Conclusion

The transportation of marine vessels and the operation of offshore oil rigs inevitably lead to oil spills, which pose significant threats to the aquatic environment. Consequently, prompt and accurate detection of surface oil spills is critical. However, distinguishing surface oil spill characteristics from background elements remains challenging, particularly due to the substantial variations in the scale of oil spills, which can easily result in misdetections.

To address these challenges, we propose a novel lightweight LSFE-YOLO model. Incorporating FasterNet for backbone network reconstruction, along with reducing the neck network's width, significantly decreases model size while enhancing detection speed. Furthermore, we introduce a more efficient GN-LSC Head detection module. This module leverages a shared convolutional structure and GroupNorm normalization to maintain detection accuracy while reducing both the number of parameters and computational redundancy. We integrated the SE attention mechanism into the model to further improve detection performance, enhancing its ability to suppress irrelevant features and better distinguish oil spills from complex background noise. Additionally, we designed a C2f_MBE module to enhance the detection of multi-scale targets, improving feature representation and mitigating the issue of missed detection caused by substantial variations in oil spill dimensions.

Experimental results demonstrate that the LSFE-YOLO model achieves improved detection accuracy while maintaining relatively low computational complexity and memory consumption. By striking a balance between detection precision and real-time performance, the proposed model meets the demanding requirements of high-accuracy, real-time marine oil spill detection. Nevertheless, we still observe instances of false detections.To further mitigate these issues, we will focus on enhancing the model's recognition capabilities by continuously accumulating sea surface oil spill data and refining its feature extraction mechanisms.

Statement: During the preparation of this work, the author(s) used Grammarly to assist in drafting and refining the text. After using this tool, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the published article.

CRediT authorship contribution statement

Jianting Shi: Writing - review & editing, Writing - original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Tianyu Jiao:** Writing - original draft, Visualization, Validation, Methodology, Formal analysis, Data curation, Conceptualization. **Daniel P. Ames:** Writing - review & editing, Supervision, Project administration, Conceptualization. **Zhijun Li:** Supervision, Conceptualization.

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.

Data availability

Data will be made available on request.

References

Ira Leifer, William J. Lehr, Debra Simecek-Beatty, Eliza Bradley et al., 2012. State of the art satellite and airborne marine oil spill remote sensing: Application to the BP Deepwater Horizon oil spill. Remote Sensing of Environment 124, 185-209. https://doi.org/10.1016/ j.rse.2012.03.024.

Camilla Brekke, Anne H.S. Solberg, 2005. Oil spill detection by satellite remote sensing 95 (1), 1-13. https://doi.org/10.1016/j.rse.2004.11.015.

Wan, Jianhua, Cheng, Yang, 2013. Remote sensing monitoring of Gulf of Mexico oil spill using. ENVISAT ASAR images. 2013 21st International Conference on Geoinformatics, 1-5. https: // doi.org /10.1109/ Geoinformatics. 2013. 6626165.

Yin, Junjun, Jian,Yang, Zhou, Zheng-Shu, Song, Jianshe,2014. The extended Bragg scattering model-based method for ship and oil-spill observation using compact polarimetric SAR. IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens. 8, 3760-3772. https: // doi.org /10.1109/ JSTARS. 2014.2359141.

Ma, Xiaoshuang, Wu, Penghai, Kong, Peng, 2021. Oil spill detection based on deep convolutional neural networks using polarimetric scattering information from Sentinel-1 SAR images. IEEE Trans. Geosci. Remote Sens. 60, 1-13. https://doi.org/10.1109/TGRS. 2021.3126175.

Zhu, Qiqi, Zhang, Yanan, Li, Ziqi, Yan, Xiaorui, Guan, Qingfeng, Zhong,Yanfei, 2021. Oil spill contextual and boundary-supervised detection network based on marine SAR images. J. IEEE Trans. Geosci. Remote Sens. 60, 1-10. https: // doi.org /10.1109/TGRS.2021.3115492.

Debra Simecek-Beatty, William J. Lehr, 2017. Extended oil spill spreading with Langmuir circulation. Mar. Pollut. Bull. 122(1-2), 226-235. https://doi.org/10.1016/j.marpolbul.2017.06.047.

Panagiota Keramea, Katerina Spanoudak, George Zodiatis, Georgios Gikas, Georgios Sylaios, 2021. Oil spill modeling: A critical review on current trends, perspectives, and challenges. Mar. Sci. Eng. 2021, 9(2), 181. https://doi.org/10.3390/jmse9020181.

Yu, Fangjie, Sun, Wuzi, Li, Jiaojiao, Zhao, Yang, Zhang, Yanmin, Chen, Ge, 2017. An improved Otsu method for oil spill detection from SAR images. Oceanologia. 59(3), 11-317. https://doi.org/10.1016/j.oceano.2017.03.005.

Li, Zhiheng, Chen, Liang, Zhang, Bocheng, Shi Hao, Long Teng, 2019. SAR image oil spill detection based on maximum entropy threshold segmentation. Signal Process. 35, 1111-1117. https://doi.org/10.16798/j.issn.1003-0530.2019.06.024.

Daeseong Kim, Hyung-Sup Jung, 2018. Mapping oil spills from dual-polarized SAR images using an artificial neural network: Application to oil spill in the Kerch Strait in November 2007. Sensors 2018, 18(7), 2237. https://doi.org/10.3390/s18072237.

Stefania Magrì,Tomaso Vairo, Andrea P. Reverber, Bruno Fabiano, 2021. Oil Spill Identification and Monitoring from Sentinel-1 SAR satellite earth observations: A machine learning approach. Chem. Eng. Trans. 86, 379-384.https://doi.org/10.3303/CET2186064.

Dong, Xiaorui, Li, Jiansheng, Li, Bing, Jin, Yueqin, Miao, Shufeng, 2023. Marine oil spill detection from low-quality SAR remote sensing images. Mar. Sci. Eng. 2023, 11(8), 1552. https://doi.org/10.3390/jmse11081552.

Xu, Weiyue, Xu Tao, Thomasson, Alex, Chen, Wei, Karthikeyan, Raghupathy,Tian, Guangzhao, Shi, Yeyin, Ji, Changying, Su, Qiong, 2023. A lightweight SSV2-YOLO based model for detection of sugarcane aphids in unstructured natural environments. Comput. Electrin. Agric. 211,107961. https://doi.org/10.1016/j.compag.2023.107961.

Xiong, Yaohua, Meng Xudong, 2019. A SAR oil spill image recognition method based on densenet convolutional neural network. C. ICRIS 2019. 78-81. https:// doi.org /10.1109/ ICRIS.2019.00028.

Huang, Xudong, Zhang, Biao, Perrie, William, Lu Yingcheng, Wang Chen, 2022. A novel deep learning method for marine oil spill detection from satellite synthetic aperture radar imagery. Mar. Pollut. Bull. 179,11366. https://doi.org/10.1016/j.marpolbul.2022.113666.

Zhu, Qiqi, Zhang,Yanan, Li, Ziqi, Yan Xiaorui, Guan,Qingfeng, Zhong, Yanfei, 2021. Oil spill contextual and boundary-supervised detection network based on marine SAR images. IEEE Transactions on Geoscience and Remote Sensing, 60. 1-10. https://doi.org/10.1109/ TGRS.2021. 3115492.

Chen, Jierun, Kao Shiu-hong, He, Hao, Zhuo, Weipeng, Wen, Song, Lee, Chui-ho, Chan, S.-H, Gary, 2023. Run, don't walk: chasing higher FLOPS for faster neural networks. C. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 2023, 12021-12031. https://doi.org/10.1109/CVPR52729.2023.01157

Wang, Guoliang, Wang Jixiang, Nie Ziling, Yang Hao, Yu Shuaiying, 2023. A Lightweight YOLOv8 Tomato Detection Algorithm Combining Feature Enhancement and Attention. Agronomy. 13(7), 1824, https://doi.org/10.3390/agronomy13071824.

Lu, Yunting, Kang, Shaopeng, Wu, Shuang, He, Chuan, 2024. Nonwoven defect detection by fusing selection kernel attention. Computer Engineering and Applications. 24 331-339, https://doi.org/10.3778/j.issn.1002-8331.2308-0182.

Wang, Chien-Yao, Bochkovskiy, Alexey, Liao, Hong-Yuan, Mark, 2023. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 7464-7475. https://doi.org/10.1109/CVPR52729.2023.00721.

Yu, Mei, Li, Yuxin, Yan, Peng, Li, Xiutong, Tian, Qin, Xie, Benliang, 2024. Dense detection algorithm for ceramic tile defects based on improved YOLOv8. J. Intell. Manuf..1-16. https:// doi.org/ 10.1007/s10845-024-02523-y.

Lee, Youngwan , Park Jongyoul, 2020 Centermask: Real-time anchor-free instance segmentation. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 3906-13915. https://doi.org/10.1109/CVPR42600.2020.01392.

Liu, Jian, Xu, Youshuan, Li, Henghui, Guo, Jiao, 2020. Soil Moisture Retrieval in Farmland Areas with Sentinel Multi-Source Data Based on Regression Convolutional Neural Network. Sensors 2021, 21(3), 877. https:// doi.org/10.3390/ s21030877.

Zhao,Yian, Lv, Wenyu, Xu, Shangliang, Wei, Jinman, Wang, Guanzhong, Dang, Qingqing, Liu, Yi, Chen, Jie 2024. Detrs beat yolos on real-time object detection. C. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 16965-16974. https:// doi.org/10.1109/CVPR52733.2024.01605.

Peng, Yanfei, Ji, Yue, 2023. Road Crack Detection Algorithm Based on Improved YOLOv8. ICAICA 2023. 28-32. https:// doi.org/10.1109/ICAICA58456.2023.10405428.

Lv, Xiaoming, Duan, Fajie, Jiang, Jiajia, Fu, Xiao, Gan, Lin, 2020. Deep metallic surface defect detection: The new benchmark and detection network. Sensors 20(6), 1562. https:// doi.org/ 10.3390/s20061562.