*Article*

# A Lightweight Transmission Line Foreign Object Detection Algorithm Incorporating Adaptive Weight Pooling

Junbo Hao [1,*], Guangying Yan [2], Lidong Wang [2], Honglan Pei [2], Xu Xiao [3] and Baifu Zhang [4]

[1] State Grid Shanxi Integrated Energy Service Co., Ltd., Taiyuan 030031, China
[2] State Grid Yuncheng Electric Power Supply Company, Yuncheng 044099, China; yanguangying@sx.sgcc.com.cn (G.Y.); wanglidong@sx.sgcc.com.cn (L.W.); peihonglan@sx.sgcc.com.cn (H.P.)
[3] State Grid Gaoping Electric Power Supply Company, Gaoping 048499, China; xiaoxu@sx.sgcc.com.cn
[4] School of Electrical and Power Engineering, Taiyuan University of Technology, Taiyuan 030024, China; zhangbaifu92@163.com
[*] Correspondence: hjb00126@126.com

**Abstract:** Aerial photography using unmanned aerial vehicles (UAVs) to detect foreign objects is an important method to ensure the safety of transmission lines. However, existing detection algorithms often encounter challenges in complex environments, including limited recognition capability and high computational demands. To address these issues, this paper proposes YOLO-LAF, a lightweight foreign object detection algorithm that is based on YOLOv8n and incorporates an innovative adaptive weight pooling technique. The proposed method introduces a novel adaptive weight pooling module within the backbone network to enhance feature extraction for detecting foreign objects on transmission lines. Additionally, a multi-scale detection strategy is designed to integrate the FasterBlock and EMA modules. This combination enables the model to effectively capture both global and local image features through cross-channel interactions, thereby reducing misdetection and omission rates. Furthermore, a C2f-SCConv module is introduced in the neck network to streamline the model by eliminating redundant features, thus improving computational efficiency. Experimental results demonstrate that YOLO-LAF achieves average accuracies of 91.2% and 85.3% on the Southern Power Grid and RailFOD23 datasets, respectively, outperforming the original YOLOv8n algorithm by 2.6% and 1.8%. Moreover, YOLO-LAF reduces the number of parameters by 23.5% and 14.8% and the computational costs by 19.9% and 24.8%, respectively. These improvements demonstrate the superior detection performance of YOLO-LAF compared to other mainstream detection algorithms.

**Keywords:** transmission line; lightweight; adaptive weight; multi-scale

## 1. Introduction

Transmission lines are a critical component of the power grid, providing a continuous and stable supply of electricity for both industrial and residential use [1]. However, as the demand for electricity grows and the coverage of transmission lines expands, these lines have become increasingly vulnerable to the effects of adverse weather, natural disasters, and foreign object interference. Such factors can lead to faults that pose significant risks to the safety and reliability of the power grid. Among these, foreign object interference is a major cause of transmission line faults, particularly in regions with strong winds or densely populated urban areas. Objects such as bird nests, hanging debris, and floating materials are prone to becoming entangled with transmission lines. Such interference not only disrupts residents' daily lives but also increases the difficulty and cost of maintenance and cleaning [2]. Therefore, the timely detection and removal of foreign objects is crucial to ensuring the safe and reliable operation of transmission lines.

Traditional target detection methods generally consist of two main steps: feature extraction and recognition. In the feature extraction phase, these algorithms rely heavily on manually extracted features, such as target size, shape, and texture. Recognition is then

performed using conventional classification algorithms [3,4]. However, these traditional methods are vulnerable to interference from complex backgrounds, particularly in scenarios with significant lighting variations or high background noise, which can severely degrade detection accuracy. Moreover, traditional approaches tend to be computationally expensive and require high-performance hardware, making them challenging to deploy for real-time detection tasks.

In recent years, deep learning [5] technology has made significant advancements in the field of target detection. Unlike traditional methods, deep learning models have the ability to automatically learn key features from images, leading to enhanced recognition performance. Consequently, these models have been widely adopted for detection tasks in various complex environments. Currently, mainstream deep learning target detection algorithms are primarily categorized into two types: two-stage detection algorithms (e.g., the R-CNN series [6–8]) and single-stage detection algorithms (e.g., the YOLO series [9–15], SSD [16]). Extensive research in this area has been conducted by scholars and research institutions. For example, Zhang et al. [17] introduced a feature balancing network into the YOLOv5 model to better balance semantic and spatial information across features of different scales. However, its detection performance for small targets in complex scenes remains inadequate. Sun et al. [18] proposed an improved YOLOv7-tiny algorithm for foreign object detection on transmission lines, utilizing channel pruning and diverse branching blocks. While this approach improves model efficiency, it often sacrifices accuracy and may fail to fully capture detailed features. Hao et al. [19] enhanced feature extraction by incorporating a triple attention mechanism (TA) and an improved bidirectional feature pyramid network (BiFPN). However, there may be conflicts among features on different scales. Wang et al. [20] enhanced the network's ability to capture key target features by introducing a two-branch pooling module in the YOLOv8 neck network. However, this approach increases computational overhead and model complexity, limiting its efficiency for real-time inspection tasks.

Compared to earlier versions of the YOLO algorithm, YOLOv8 further optimizes the model structure and feature extraction techniques, utilizing a deeper feature fusion module to capture more detailed image features. These enhancements make YOLOv8 promising for foreign object detection on transmission lines. However, YOLOv8 encounters challenges in detecting small foreign objects due to the limitations of traditional pooling operations, which lack adaptive feature processing across different channels. Additionally, the feature fusion mechanism for handling multi-scale targets is inadequate in complex environments, leading to reduced recognition accuracy for targets on varying scales. To address these challenges, this study proposes a novel lightweight object detection method based on YOLOv8. The proposed method enhances the feature extraction capability for small targets, increases detection accuracy in complex scenes, and optimizes the model structure to lower computational costs. The main contributions of this study are as follows:

(1) Designing a lightweight adaptive weight pooling module that dynamically adjusts channel weights for adaptive feature processing. This approach minimizes the loss of critical features during pooling, allowing more effective capture and preservation of key feature information. Consequently, the quality of the pooled feature representation is enhanced, improving the model's ability to detect small objects.

(2) Constructing an efficient multi-scale fusion module by integrating the FasterBlock module with the EMA attention mechanism. This module effectively fuses features across different scales, seamlessly combining global and local features. It enhances the model's ability to comprehend complex scenes, resulting in better generalization and robustness.

(3) Introducing the C2f-SCConv module with partial connectivity to reduce redundant computations, which ensures that the model remains lightweight while retaining strong feature representation capabilities. The module also performs spatial convolutions across different channel features, capturing and expressing inter-feature

relationships more effectively. This enhances the model's understanding of input data features and boosts overall performance.

## 2. Improved Algorithm YOLO-LAF, Based on YOLOv8n

YOLOv8 makes several optimizations and improvements based on the YOLOv5 algorithm: (1) The C3 structure of YOLOv5 is replaced by the C2f structure, which provides a richer gradient flow, significantly improving model performance. (2) The head network adopts an anchor-free design, eliminating issues related to the mismatch between anchor boxes and actual targets, thus enhancing detection flexibility. (3) More efficient activation functions, such as SiLU or Mish, are introduced, boosting the model's convergence speed and overall performance.

In order to further improve the robustness and accuracy of YOLOv8 in the transmission line foreign object detection environment, this paper proposes a lightweight adaptive weighted pooling multi-scale foreign object detection algorithm. The network architecture is illustrated in Figure 1. Firstly, we replace the last three convolutional blocks of the original backbone network and the first convolutional block of the neck network with a lightweight adaptive weight pooling module. This module dynamically adjusts weights based on input feature differences, ensuring strong detection capability in complex scenes. Secondly, we replace the C2f module in the backbone network with an efficient multi-scale fusion module, which integrates features from multiple resolutions. This enhances the model's ability to detect foreign objects in challenging environments. Finally, we introduce the C2f-SCConv module after the Concat connection layer in the neck network. This module reduces model complexity and computational cost by minimizing redundant features, thus significantly improving overall performance.
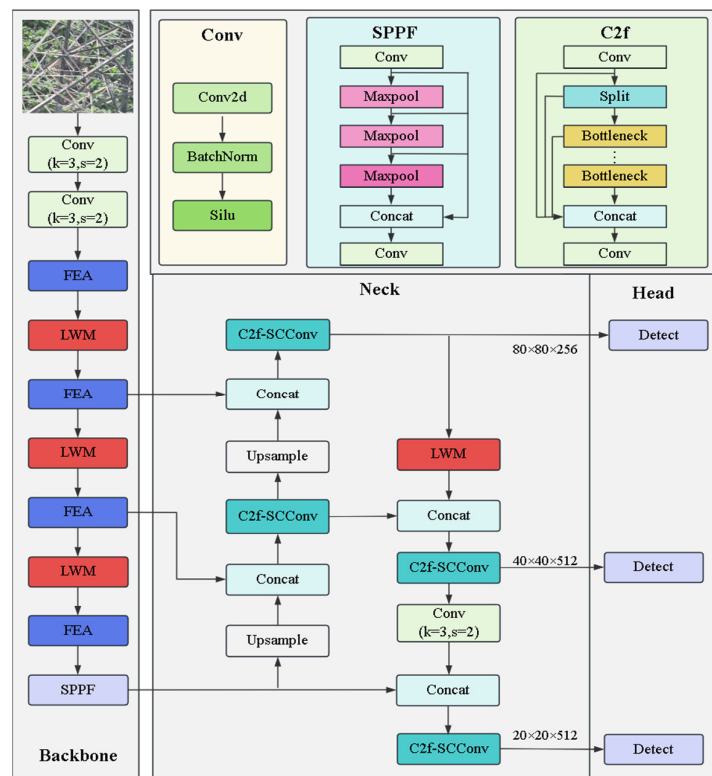


**Figure 1.** Network architecture of YOLO-LAF.

### 2.1. Lightweight Adaptive Weight Pooling Module

To address the challenge of feature extraction imbalance in foreign object detection within the complex environment of transmission lines, this paper proposes the Lightweight Adaptive Weighted Pooling Module (LWM for short). The structure of this module is

shown in Figure 2. The LWM specifically targets the issue of smaller objects being lost during feature pooling in the detection process. By dynamically adjusting the pooling weights, the module adaptively allocates feature extraction resources based on the size of the targets, ensuring that key features are effectively preserved.
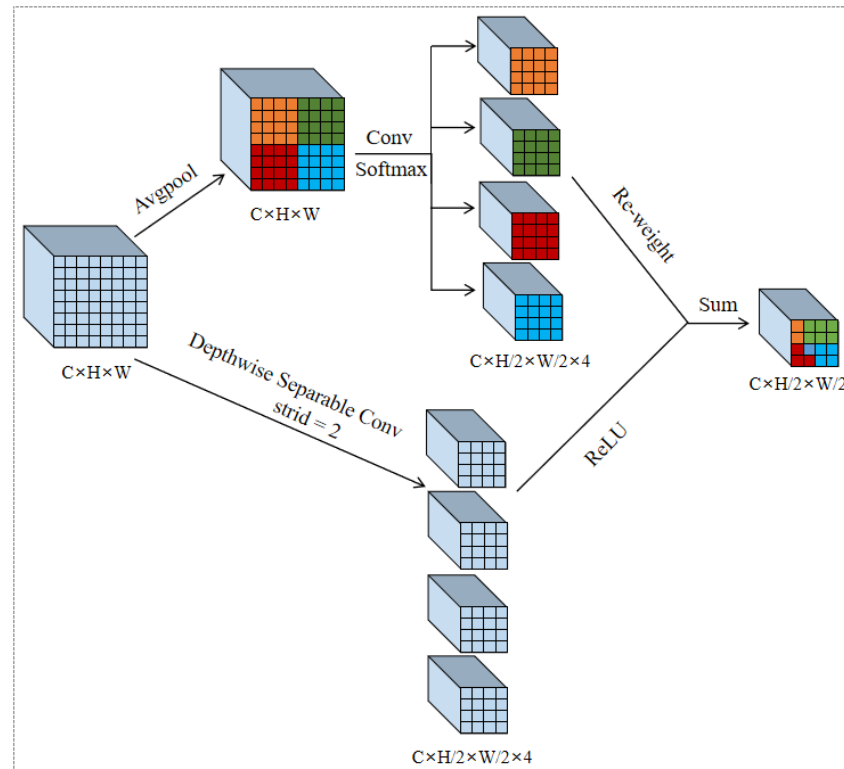


**Figure 2.** LWM module structure.

The LWM module contains two branches, of which the first branch generates a weight map through average pooling and $1 \times 1$ convolution. It calculates the importance of each position in the attention weight map by transforming the array dimensions, preserving key information and features as much as possible. The resulting weights are then normalized into a probability distribution using the Softmax activation function, ensuring that the weights across all regions sum to 1. The second branch draws inspiration from the Focus slicing operation, which reorganizes the spatial structure in the feature map. This operation redistributes pixel points initially in the spatial dimension into the channel dimension, effectively compressing spatial information into the channel and simplifying model processing. However, due to the high computational cost of this slicing operation, we propose an improvement in which it is replaced with a depth-separable convolution with a stride of 2. This modification significantly reduces computational overhead, enhancing the model's efficiency and making it more lightweight. Depthwise separable convolution decomposes the standard convolution into depthwise convolution and pointwise convolution. In depthwise convolution, each input channel is convolved independently using a separate kernel. In pointwise convolution, a $1 \times 1$ convolution kernel is used to process the output of depthwise convolution. This approach improves computational efficiency and model compression. Finally, the weight information extracted from both branches is fused by using a weighted summation operation, which ensures that the model maintains feature diversity while improving computational speed and detection performance. In summary, the computation of the *LWM* module can be expressed as follows:

$$LWM = Softmax(c^{1 \times 1}(AvgPool(X))) \times ReLU(Norm(d^{k \times k}(X))) \tag{1}$$

In the equation, $c^{1 \times 1}$ refers to a $1 \times 1$ convolution, while $d^{k \times k}$ denotes a depthwise separable convolution with a kernel size of k × k. The term *Norm* indicates normalization, *X* represents the input feature map, and *AvgPool* signifies average pooling. To minimize computational overhead and reduce the number of parameters, average pooling is used to aggregate global feature information from each receptive field. A $1 \times 1$ convolution is then applied to facilitate the exchange of information among the features. Finally, the Softmax activation function is applied to emphasize the importance of each feature within the receptive field. The dimensional transformation of the array is expressed in Equation (2).

$$DIM = \{bs, ch, h, w\} = \{bs, ch, h/2, w/2, S\} \tag{2}$$

In the equation, *bs* represents the batch size; *ch* denotes the number of channels in the input feature map; and *h* and *w* refer to the height and width of the original feature map, respectively. *S* represents the weight information, with a default size of 4. As indicated in Equation (2), after the pooling operation, the height and width of the original feature map are reduced to half of their original dimensions, while the number of channels remains unchanged. The feature information is preserved in the weight channel *S*, enabling the network to focus more effectively on detailed information and ensuring that key features are successfully captured, even in complex environments.

### 2.2. FasterBlock–EMA Module

The foreign object detection model for transmission lines requires substantial supplies of data and computational resources. Additionally, the imbalance between global and local feature representation in the training data causes the model to converge prematurely on certain features, leading to poor generalization in complex environments and increasing the risk of false detections or missed objects. To address this issue, this paper introduces the FasterBlock [21] module into the YOLOv8n algorithm; the module effectively reduces redundant convolutional computations and memory accesses, thus enhancing the model's operation speed and resource utilization efficiency [22]. In addition, the Efficient Multi-Scale Attention (EMA) [23] module is introduced into the FasterBlock module to construct the multi-scale module FasterBlock–EMA (FEA for short). This enhancement aims to further improve the performance and efficiency of the model across various scenarios.

The FasterBlock module consists of four parts: Partial Convolution (PConv), Conv, Batch Normalization (BN), and a Rectified Linear Unit (ReLU). PConv selectively applies standard convolution only to a subset of input channels for spatial feature extraction, while leaving the remaining channels unchanged. This reduces the computational burden and improves the processing speed of the model. In addition, during consecutive or regular memory accesses, PConv computes only the first or last contiguous channel as a representative of the whole feature map, ensuring an equal number of channels for input and output feature maps. Therefore, this module is well suited for vision tasks requiring fast processing.

Modelling cross-channel relationships through channel dimensionality reduction may have a negative impact on deep visual feature extraction; to address this problem, we introduce the EMA model without dimensionality reduction, thereby preserving the information of each channel while reducing computational overhead. Additionally, EMA introduces an information aggregation method across spatial dimensions, enabling richer feature fusion. When EMA is combined with the FasterBlock module, the resulting FEA module further reduces computational costs and selectively emphasizes key local features while maintaining attention on global features. This improves the detection performance of the models for multi-scale targets. The structure of FasterBlock–EMA is illustrated in Figure 3, where * represents the convolution operation.
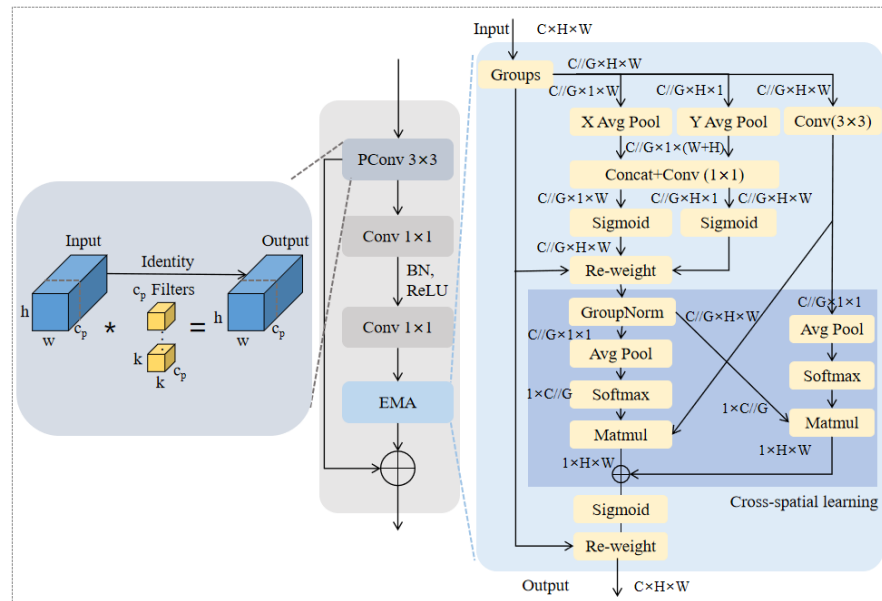
**Figure 3.** FasterBlock–EMA module structure.

### 2.3. C2f-SCConv Module

In target detection tasks, the extraction of redundant features by convolutional layers not only increases the computational burden and memory consumption but may also degrade model performance, particularly in complex scenes and for small targets. To address these challenges, this paper introduces the Spatial and Channel Reconstruction Convolution (SCConv) [24] to improve the Bottleneck module in the original C2f structure. The proposed C2f-SCConv module replaces standard convolution with SCConv, forming a new SCBlock module that is embedded in the C2f structure. The structure of this module is shown in Figure 4, where h and w represent the height and width of the original feature map, c is the number of channels, and n denotes the number of layers.
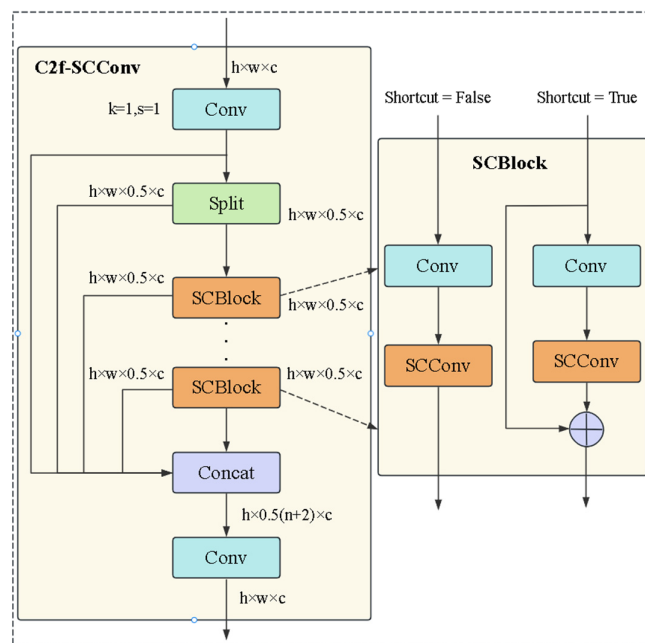


**Figure 4.** C2f-SC module structure.

The structure of SCConv is shown in Figure 5 and primarily consists of the Spatial Reconstruction Unit (SRU) and the Channel Reconstruction Unit (CRU). The SRU

addresses spatial redundancy by employing weight decomposition to separate and reconstruct redundant features, thereby suppressing redundancy in spatial dimensions and enhancing the expressiveness of the features. The CRU adopts a "split–transform–merge" strategy to effectively reduce channel redundancy, lowering computational and storage costs. By combining these two reconstruction units, SCConv accurately captures complex relationships within the input features. This not only controls feature redundancy but also reduces the number of model parameters and floating-point operations per second (FLOPs), significantly enhancing the model's feature extraction capability.
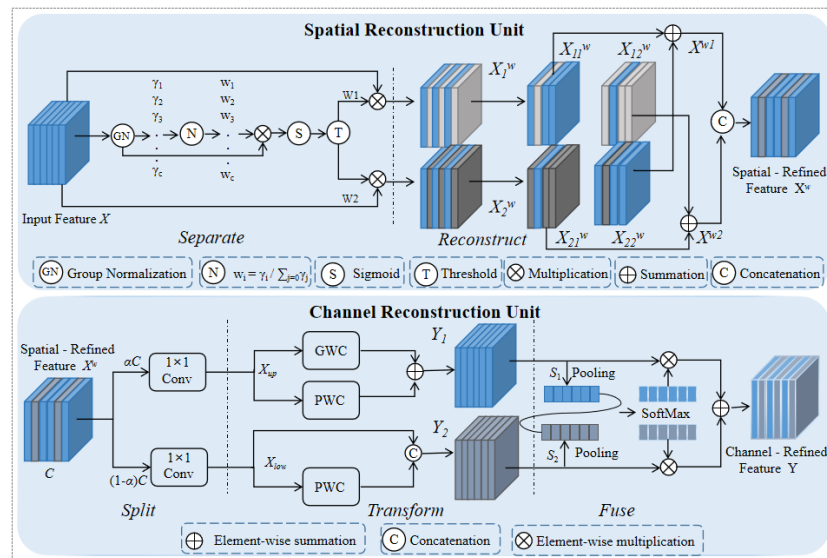


**Figure 5.** SCConv module structure.

## 3. Experiments

### 3.1. Experimental Environment and Parameter Configuration

In the model training process of this study, the Stochastic Gradient Descent (SGD) optimizer [25] was utilized to reduce the risk of the model converging to local optima. The training environment included PyTorch 2.0.1, CUDA 11.3, and Python 3.9.0, with the detailed server configurations provided in Table 1. The input for model training was a $640 \times 640$ three-channel image, with an initial learning rate of 0.01, a momentum factor of 0.937. The batch size was set to 16, and the model was trained for a total of 300 epochs.

**Table 1.** Server configuration environment.

| Configuration Name | Version/Parameter |
| --- | --- |
| Operating system | Ubuntu 20.04LTS |
| GPU | RTX4090ti * 2 |
| RAM | 48 GB |
| Memory | 2 TB SATA |

### 3.2. Experimental Dataset

To evaluate the performance of the target detection algorithm proposed in this paper, experiments were conducted on two public datasets: the Southern Power Grid dataset and RailFOD23 [26] dataset. The dataset splits are detailed in Table 2, while the label distribution is shown in Figure 6. The effect of these datasets on the performance of YOLO-LAF in detecting each target is summarized in Table 3. A brief description of the two datasets is provided below:

(1)   The Southern Power Grid dataset primarily consists of data collected by drones, totaling 2400 images that encompass four types of foreign objects: bird nests, kites,

balloons, and rubbish. The UAVs are equipped with multi-spectral cameras capable of capturing images in various spectral bands, including visible light and near-infrared. These bands enable the identification of foreign objects on transmission lines by analyzing reflectivity and texture characteristics. Potential threats, such as kites, bird nests, and rubbish, can be effectively distinguished through this analysis. Considering the limited number of images, data augmentation techniques such as flipping, scaling, and cropping were applied to expand the dataset. An example of data augmentation is shown in Figure 7. After augmentation, the dataset increased to 3200 images, which were then split into training, validation, and test sets in an 8:1:1 ratio.

(2) The RailFOD23 dataset leverages large models such as ChatGPT and text-to-image generation techniques to create foreign object detection data for railway power transmission lines. It includes four common types of foreign objects: plastic bags on power lines, objects fluttering or suspended on wires, bird nests on transmission towers, and balloons near transmission lines. This dataset contains 14,615 images and 40,541 annotated objects, divided into training, validation, and test sets in a 7:2:1 ratio.
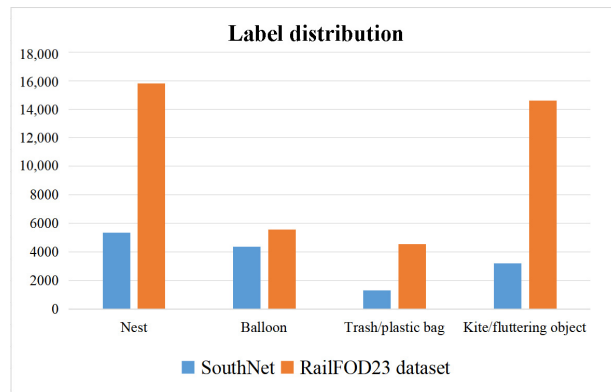


**Figure 6.** Label distribution.
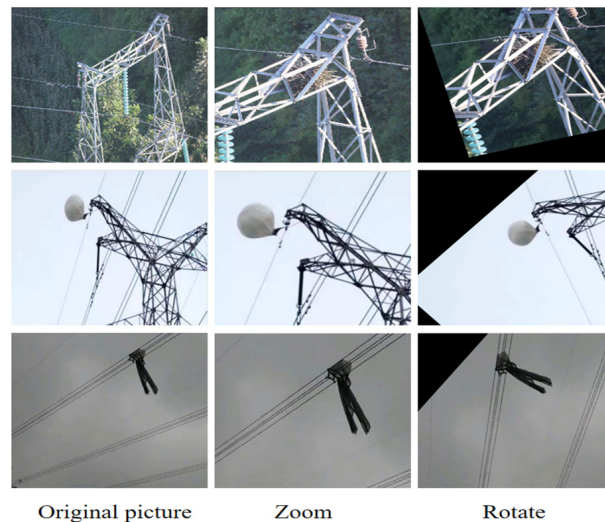


**Figure 7.** Data enhancement example diagram.

**Table 2.** Dataset distribution.

| Dataset | Train | Val | Test | Total |
|---|---|---|---|---|
| Southern Power Grid | 2560 | 320 | 320 | 3200 |
| RailFOD23 | 10,230 | 2923 | 1462 | 14,615 |

**Table 3.** YOLO-LAF models detect results in each target category.

| Category | Southern Power Grid | | | RailFOD23 | | |
|---|---|---|---|---|---|---|
| | **P %** | **R %** | **mAP50 %** | **P %** | **R %** | **mAP50 %** |
| Plastic bag | 92.9 | 94.6 | 96.9 | 91.4 | 87.6 | 91.6 |
| Fluttering object | 92.5 | 81.8 | 84 | 88.4 | 73.6 | 77.1 |
| Nest | 87.2 | 89.4 | 92.4 | 90.4 | 79.1 | 88.2 |
| Balloon | 85.3 | 82.6 | 91.5 | 90.2 | 72.5 | 84.3 |
| All | 89.5 | 87.1 | 91.2 | 90.1 | 78.2 | 85.3 |

### 3.3. Evaluation Index

When evaluating object detection algorithms, it is essential to consider key metrics such as detection accuracy, detection speed, and memory usage. Therefore, this paper utilizes precision, recall, mean average precision (*mAP*), giga-floating point operations per second (*GFLOPs*), and the number of parameters (*Params*) as evaluation metrics [27]. The specific calculation methods are detailed below.

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$AP = \int_0^1 Precision(t)dt \tag{5}$$

$$mAP = \frac{1}{N}\sum_{m=1}^{N} AP(n) \tag{6}$$

In the formulas, *T* and *F* represent the true positive and true negative classes, respectively, while P and N indicate the predicted positive and negative classes. *TP* refers to the number of samples that are both truly positive and predicted as positive, *FP* refers to the number of samples that are actually negative but predicted as positive, and *FN* refers to the number of samples that are actually positive but predicted as negative. *AP* represents the average precision for each category, while *mAP* is the mean of the *AP* values across all categories.

### 3.4. Experimental Results and Analysis

3.4.1. Experimental Analysis of Lightweight Adaptive Weight Pooling Module

To verify the effectiveness of the lightweight adaptive weight pooling module (LWM), we compared its performance when integrated at different positions within YOLOv8. Meanwhile, in order to retain more key information during the pooling process, we replaced the last three convolutional blocks in the backbone network and the two convolutional blocks in the neck network with the proposed LWM module. Comparative experiments were conducted on the SouthNet dataset, and the results are shown in Table 4.

In Table 4, YOLOv8n-LWM-i indicates that the i-th standard convolution has been replaced with the LWM module. Replacing standard convolutions with the LWM module at various positions improves detection accuracy while reducing the number of model parameters and computational requirements. Among the configurations, YOLOv8n-LWM-3,4,5,6 achieved the best performance, with mAP50 of 90.7%, which is 2.5% higher than that of the original YOLOv8n. Furthermore, the number of parameters was reduced by 0.87M, and the computational cost decreased by 0.9 GFLOPs.

**Table 4.** Validation experiment of adaptive weight pooling module on the SouthNet dataset.

| Model | Precision % | Recall % | Params M | GFLOPs | mAP50 % |
|---|---|---|---|---|---|
| YOLOv8n | 90.8 | 83.7 | $3.06 \times 10^6$ | 8.1 | 88.6 |
| YOLOv8n-LWM-3 | 91.0 | 84.2 | $3.01 \times 10^6$ | 8.1 | 89.7 |
| YOLOv8n-LWM-4 | 91.1 | 84.1 | $2.97 \times 10^6$ | 8.0 | 89.4 |
| YOLOv8n-LWM-5 | 90.6 | 84.5 | $2.63 \times 10^6$ | 7.8 | 89.4 |
| YOLOv8n-LWM-6 | 90.2 | 83.9 | $2.87 \times 10^6$ | 7.9 | 90.1 |
| YOLOv8n-LWM-7 | 89.3 | 84.2 | $2.45 \times 10^6$ | 8.1 | 88.5 |
| YOLOv8n-LWM-3,4,5 | 90.6 | 83.3 | $2.34 \times 10^6$ | 6.5 | 89.1 |
| YOLOv8n-LWM-3,4,5,6 | 91.6 | 85.8 | $2.19 \times 10^6$ | 7.2 | 90.7 |
| YOLOv8n-LWM-3,4,5,6,7 | 90.6 | 85.5 | $2.68 \times 10^6$ | 7.9 | 90.2 |

### 3.4.2. Attention Mechanism Selection Experiment

To evaluate the performance of the EMA model within the multi-scale fusion module (FEA), this paper conducts comparative experiments between the EMA model in the FEA module and other attention mechanisms, including SE (Squeeze-and-Excitation) [28], CBAM (Convolutional Block Attention Module) [29], ECA (Efficient Channel Attention) [30], CA (Coordinate Attention) [31], and SimAM (Simple Attention Module) [32], using the RailFOD23 dataset. The results presented in Table 5 show that the model incorporating the EMA model achieved the highest detection accuracy. Compared with the original YOLOv8n model, the accuracy increased by 1.7%, the computational cost was reduced by 2.2GFLOPs, and the overall detection performance improved by 2.5%. In summary, the EMA model outperformed all other tested mechanisms, demonstrating its significant advantages within the FEA module.

**Table 5.** Attention mechanism selection experiment on the RailFOD23 dataset.

| Model | Precision % | Recall % | Params M | GFLOPs | mAP50 % |
|---|---|---|---|---|---|
| YOLOv8n | 87.8 | 81.7 | $3.06 \times 10^6$ | 9.3 | 83.5 |
| YOLOv8n+FasterBlock–SE | 87.5 | 82.2 | $2.73 \times 10^6$ | 10.9 | 84.7 |
| YOLOv8n+FasterBlock–CBAM | 87.3 | 84.3 | $2.55 \times 10^6$ | 9.4 | 85.1 |
| YOLOv8n+FasterBlock–ECA | 88.7 | 83.5 | $2.32 \times 10^6$ | 8.6 | 84.5 |
| YOLOv8n+FasterBlock–CA | 88.4 | 83.9 | $2.56 \times 10^6$ | 10.8 | 85.7 |
| YOLOv8n+FasterBlock–SimAM | 88.3 | 82.9 | $2.48 \times 10^6$ | 8.5 | 85.3 |
| YOLOv8n+FasterBlock–EMA | 89.5 | 83.6 | $2.35 \times 10^6$ | 7.1 | 86.0 |

### 3.4.3. Ablation Experiment

In this paper, YOLOv8n is selected as the baseline model, and ablation experiments are conducted on the Southern Power Grid and RailFOD23 datasets. The detection results are shown in Tables 6 and 7, respectively. From the tables, it can be seen that the accuracy of the LWM module on the Southern Power Grid and RailFOD23 datasets improved by 1.6% and 0.8%, respectively. This improvement is attributable to the LWM's enhancement of small target feature extraction through the adaptive weighting module, which, in turn, boosted the model's detection performance. The introduction of the FasterBlock–EMA module improved detection accuracy by 2% and 1.1%, respectively, demonstrating that the FEA module significantly enhances the model's feature extraction ability and improves detection in complex environments. Replacing the original C2f module with the C2f-SCConv module not only reduces parameters and computational cost but also further improves detection accuracy, proving its effectiveness in lightweight design. The performance improvement of the YOLO-LAF algorithm proposed in this paper was the most significant, with the mAP increasing by 2.6% and 1.8%, respectively, while both computational costs and parameter counts were significantly reduced. In summary, the modules proposed in this paper show obvious advantages in feature extraction, accuracy improvement, and computational efficiency optimization.

**Table 6.** Results of ablation experiments on the SouthNet dataset.

| Index | LWM | FasterBlock–EMA | C2f-SCConv | Params M | GFLOPs | mAP50 % |
|-------|-----|-----------------|------------|----------|--------|---------|
| A | | | | $3.06 \times 10^6$ | 8.1 | 88.6 |
| B | √ | | | $2.81 \times 10^6$ | 7.9 | 90.2 |
| C | | √ | | $2.65 \times 10^6$ | 7.1 | 90.6 |
| D | | | √ | $2.82 \times 10^6$ | 7.7 | 90.8 |
| E | √ | √ | | $2.29 \times 10^6$ | 7.5 | 90.5 |
| F | √ | | √ | $2.49 \times 10^6$ | 7.6 | 90.2 |
| G | | √ | √ | $2.47 \times 10^6$ | 7.8 | 90.4 |
| H | √ | √ | √ | $2.35 \times 10^6$ | 6.9 | 91.2 |

"√" indicates the corresponding method added.

**Table 7.** Results of ablation experiments on the RailFOD23 dataset.

| Index | LWM | FasterBlock–EMA | C2f-SCConv | Params M | GFLOPs | mAP50 % |
|-------|-----|-----------------|------------|----------|--------|---------|
| A | | | | $3.06 \times 10^6$ | 9.3 | 83.5 |
| B | √ | | | $2.97 \times 10^6$ | 9.4 | 84.3 |
| C | | √ | | $2.69 \times 10^6$ | 8.7 | 84.6 |
| D | | | √ | $2.75 \times 10^6$ | 8.9 | 83.9 |
| E | √ | √ | | $2.52 \times 10^6$ | 8.4 | 84.5 |
| F | √ | | √ | $2.64 \times 10^6$ | 8.6 | 83.7 |
| G | | √ | √ | $2.67 \times 10^6$ | 8.3 | 84.8 |
| H | √ | √ | √ | $2.45 \times 10^6$ | 8.5 | 85.3 |

"√" indicates the corresponding method added.

### 3.4.4. Comparative Experiment

To further validate the performance of the YOLO-LAF detection model, comparative experiments were conducted with current mainstream target detection algorithms on the Southern Power Grid and RailFOD23 datasets. The experimental results are shown in Tables 8 and 9.

**Table 8.** Results of comparison experiments on the Southern Power Grid dataset.

| Model | Precision % | Recall % | Params M | GFLOPs | mAP50 % |
|-------|-------------|----------|----------|--------|---------|
| Faster R-CNN | 83.2 | 81.9 | $52.7 \times 10^7$ | 295.7 | 82.6 |
| YOLOv3 | 84.6 | 84.6 | $78.5 \times 10^6$ | 134.6 | 67.7 |
| YOLOv5s | 83.1 | 78.8 | $20.8 \times 10^6$ | 48.3 | 84.5 |
| Reference [17] | 85.3 | 81.4 | $51.4 \times 10^6$ | 27.5 | 86.9 |
| YOLOX | 90.6 | 83.5 | $4.23 \times 10^6$ | 11.8 | 90.1 |
| YOLOv7-tiny | 90.2 | 88.9 | $18.3 \times 10^6$ | 41.26 | 88.2 |
| Reference [18] | 88.3 | 81.0 | $5.53 \times 10^6$ | 10.2 | 89.7 |
| YOLOv8n | 90.8 | 83.7 | $3.06 \times 10^6$ | 8.1 | 88.6 |
| YOLOv8s | 90.1 | 85.2 | $11.1 \times 10^6$ | 28.4 | 90.3 |
| YOLOv9-t | 92.7 | 87.2 | $2.61 \times 10^6$ | 10.7 | 90.9 |
| YOLOv10 | 91.4 | 88.3 | $2.69 \times 10^6$ | 8.2 | 89.5 |
| YOLO-LAF | 91.6 | 89.7 | $2.35 \times 10^6$ | 6.9 | 91.2 |

Tables 8 and 9 show the performance comparison of different detection algorithms. Faster R-CNN is more computationally intensive and slower due to its complexity. The YOLO family of algorithms (YOLOv3, YOLOv5s, YOLOX, YOLOv7, etc.) significantly reduces the number of parameters and the amount of computation through iterative versions. However, there is still room for optimization in terms of feature fusion and minimizing information loss. Although YOLOv9 and YOLOv10 outperform YOLOv8n in terms of accuracy and number of parameters, their generalization ability is weaker in transmission line foreign object detection, especially when detecting small targets or partially occluded objects, where accuracy decreases significantly. The YOLO-LAF model

proposed in this paper not only reduces the number of parameters to 2.35M and 2.45M and the computational cost to 6.9 GFLOPs and 8.5 GFLOPs but also improves detection accuracy to 91.2% and 85.3%, respectively. Meanwhile, YOLO-LAF exhibits less fluctuation on the precision-recall curve (e.g., Figure 8), demonstrating higher stability and generalization.

**Table 9.** RailFOD23 dataset compared experimental results.

| Model | Precision % | Recall % | Params M | GFLOPs | mAP50 % |
|---|---|---|---|---|---|
| Faster R-CNN | 86.3 | 78.8 | $89.6 \times 10^7$ | 248.1 | 83.9 |
| YOLOv3 | 81.5 | 82.7 | $97.4 \times 10^6$ | 154.7 | 70.8 |
| YOLOv5s | 82.1 | 80.6 | $34.2 \times 10^6$ | 67.2 | 82.7 |
| Reference [17] | 83.9 | 80.7 | $68.1 \times 10^6$ | 38.4 | 81.9 |
| YOLOX | 87.2 | 78.9 | $10.52 \times 10^6$ | 13.9 | 83.4 |
| YOLOv7-tiny | 88.5 | 82.3 | $23.9 \times 10^6$ | 60.7 | 82.5 |
| Reference [18] | 84.1 | 78.4 | $7.69 \times 10^6$ | 13.4 | 81.6 |
| YOLOv8n | 85.1 | 78.6 | $3.06 \times 10^6$ | 9.3 | 83.5 |
| YOLOv8s | 87.6 | 81.4 | $17.9 \times 10^6$ | 22.1 | 84.6 |
| YOLOv9-t | 91.3 | 84.5 | $3.87 \times 10^6$ | 13.1 | 83.2 |
| YOLOv10 | 92.2 | 80.4 | $2.72 \times 10^6$ | 7.4 | 84.7 |
| YOLO-LAF | 90.1 | 78.2 | $2.45 \times 10^6$ | 8.5 | 85.3 |
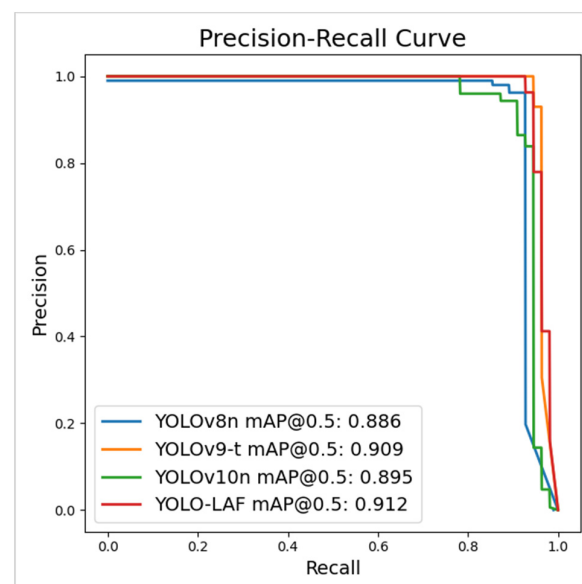


**Figure 8.** Comparison of P-R curves.

### 3.4.5. Visualization and Analysis

To more intuitively compare the performance of the transmission line foreign object detection model before and after the improvement, this paper selects representative images for visual analysis, with some of the detection results shown in Figure 9. From the figure, it can be observed that the Faster R-CNN algorithm exhibited serious misdetection and omission, as indicated by the red circles in the figure. Although the YOLOv5s, YOLOv7-tiny, YOLOv8n, YOLOv9t, and YOLOv10n algorithms showed improved detection results, misdetections and false detections still occurred due to the foreign object targets occupying fewer pixels in the image. Additionally, the detection results of references [17,18] showed even more pronounced cases of misdetection and false detection. In contrast, the improved YOLO-LAF algorithm proposed in this paper can effectively solve the leakage and misdetection problems existing in other algorithms. The detection accuracy was significantly improved while meeting the speed requirements for real-time detection, making it more suitable for actual transmission line foreign object detection tasks.
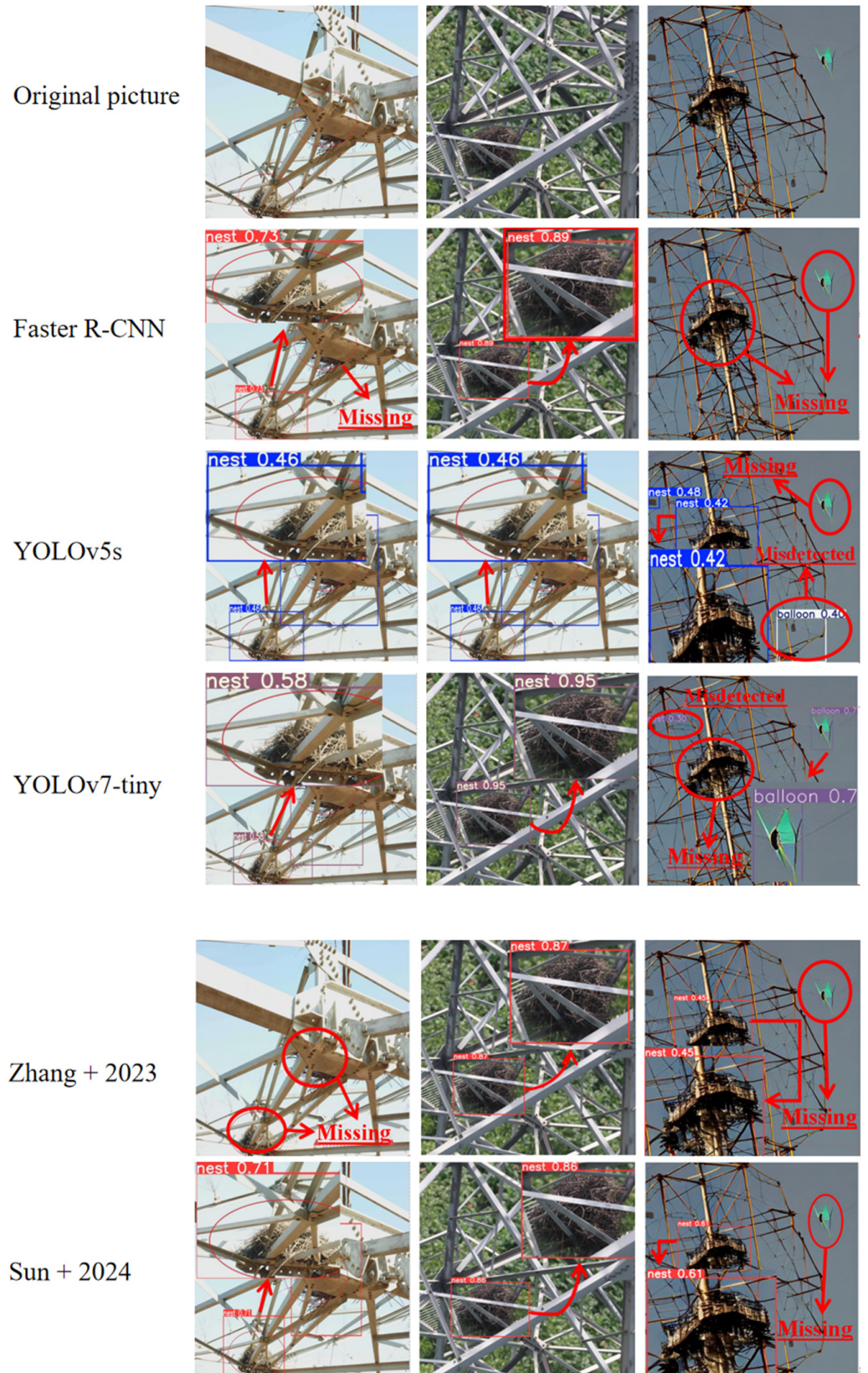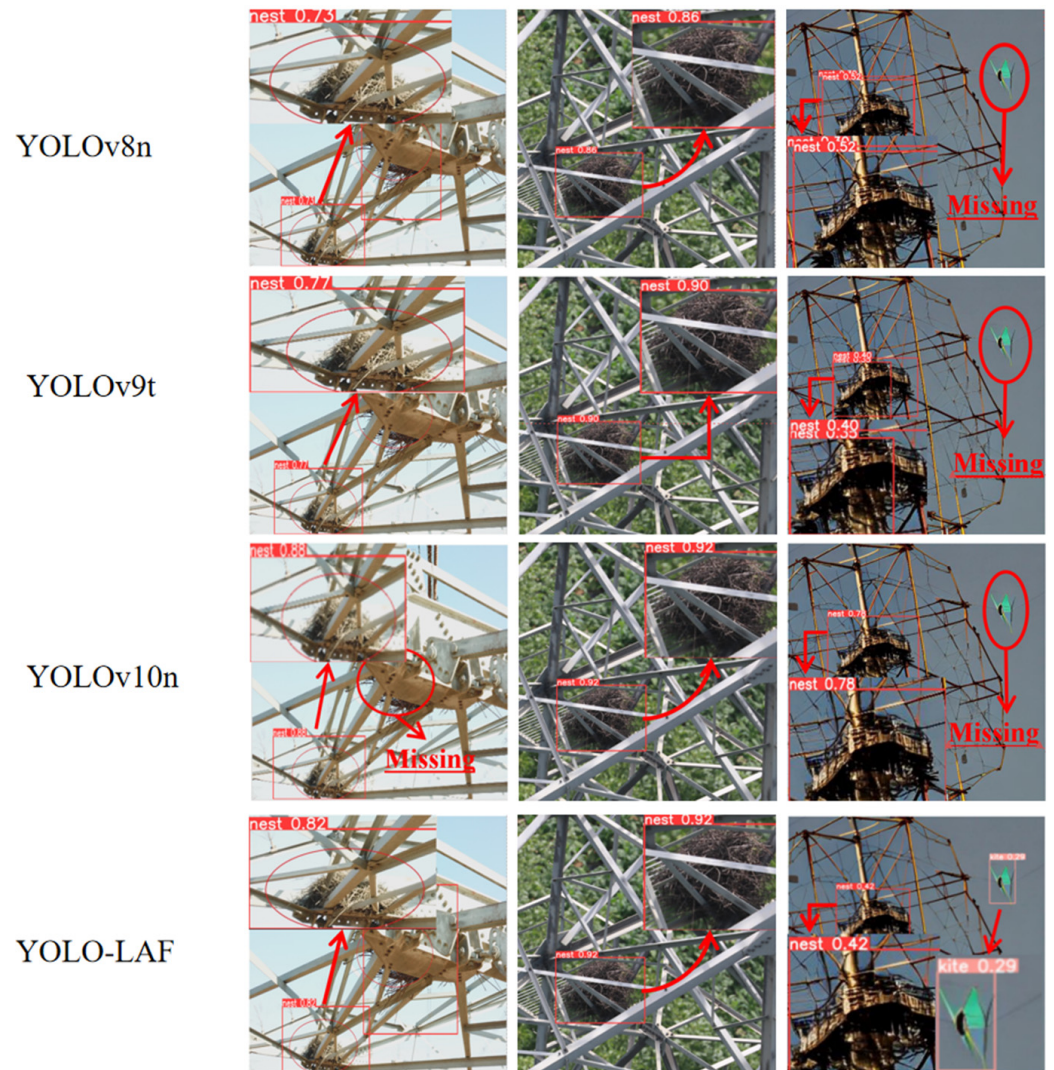
**Figure 9.** *Cont.*

**Figure 9.** Visualization results of YOLO series and References [17,18].

### 3.4.6. Thermal Map Visualization Analysis

To clearly demonstrate the effectiveness of the proposed method in regional image quality assessment, this paper employs Gradient-weighted Class Activation Mapping (Grad-CAM [33]) for heatmap visualization analysis. Grad-CAM generates heatmaps by computing the gradients of the feature maps from convolutional neural networks, highlighting the model's focus on different areas during image quality detection. Figure 10 illustrates the heatmaps before and after the model improvements, where red indicates the regions of highest attention, yellow represents areas with moderate attention, and blue signifies areas with minimal impact on image recognition. As shown in Figure 10, the contours and shapes of the target regions of the improved YOLOv8-LAF model's heat map are much clearer, revealing more high-confidence regions. Especially in scenes with complex backgrounds or dense targets, the enhanced feature extraction of the target makes the demarcation between the target region and the background more obvious, while also providing stronger noise suppression. Furthermore, through the optimization of the feature extraction module, the model effectively reduces the high rate of responses to the background region, allowing the heatmap to focus more on the target.
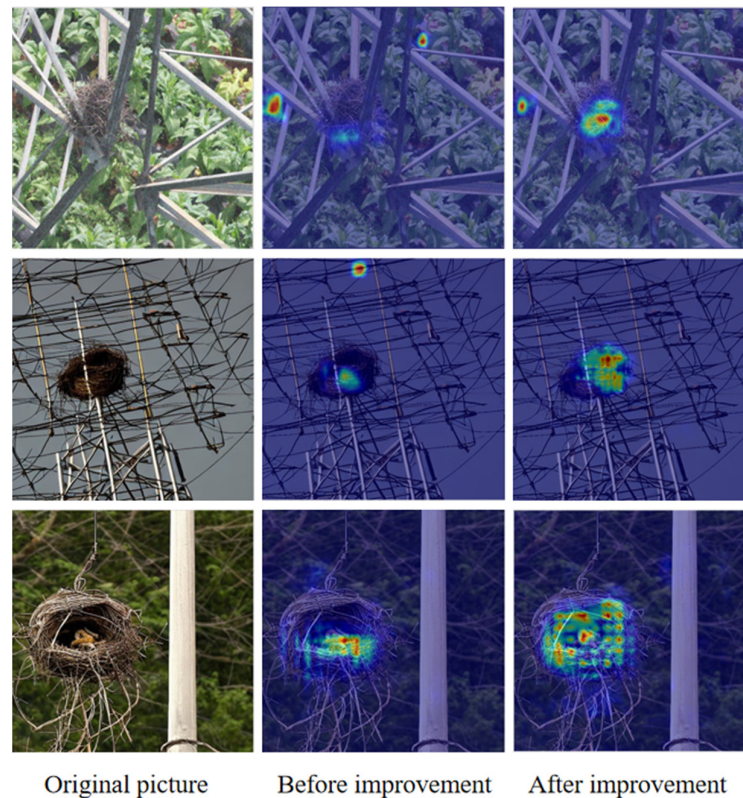
Original picture      Before improvement      After improvement

**Figure 10.** Comparison of heat maps before and after model improvement.

## 4. Discussion and Conclusions

This paper proposes an improved YOLO-LAF model based on the YOLOv8 algorithm, which is innovatively tailored to the demands of foreign object detection in transmission line inspection tasks. By incorporating practical application scenarios, the model demonstrates that the adaptive weighting module plays an important role in enhancing feature extraction and reducing the loss of information during pooling. However, the excessive use of adaptive weighting modules may result in the loss of detailed information, negatively impacting model performance. Therefore, a better balance between computational efficiency and detection accuracy can be achieved by reasonably configuring the number and placement of these modules.

To improve the accuracy and efficiency of foreign object detection on transmission lines, this paper makes improvements in three aspects: (1) A lightweight adaptive weight pooling module (LWM) is designed to enhance the model's ability to effectively capture foreign object target information during the pooling process. (2) An efficient multi-scale fusion module (FEA) is constructed to improve the fusion of global and local information for foreign object targets in complex environments. (3) The C2f-SCConv module is integrated into the neck network layer to boost the real-time detection efficiency of the model. Experimental results showed that the proposed algorithm outperformed existing YOLO series models on two publicly available datasets, Southern Power Grid and RailFOD23, with detection accuracies of 91.2% and 85.3%, respectively, showing improvements of 2.6% and 1.8% over the original YOLOv8 model. Additionally, the number of model parameters was reduced by 23.5% and 14.8%, respectively, while the computation volume decreased by 19.9% and 24.8%, respectively, resulting in significantly improved detection performance in transmission line foreign object detection.

Although the YOLO-LAF algorithm has achieved improvement in detection accuracy and efficiency, its robustness still needs to be further validated in highly complex scenarios, such as detecting transmission lines under severe weather conditions. Future work will focus more on how to improve the robustness of the model in extremely complex envi-

ronments and explore additional lightweight techniques to further optimize the model structure; the model will be deployed in industrial settings, and more comprehensive data will be collected simultaneously to improve its performance in complex scenarios.

**Author Contributions:** Conceptualization: J.H.; methodology: J.H. and L.W.; software: H.P.; validation: G.Y. and X.X.; formal analysis: G.Y.; investigation: B.Z.; resources: L.W.; data curation: L.W.; writing—original draft preparation: J.H.; writing—review and editing: G.Y.; supervision: X.X.; project administration: J.H.; funding acquisition: B.Z. All authors have read and agreed to the published version of the manuscript.

**Data Availability Statement:** RailFOD23 public dataset URL: https://figshare.com/articles/figure/RailFOD23_zip/24180738?file=43616139 (accessed date: 15 November 2024).

**Conflicts of Interest:** The authors declare no conflicts of interest. Author J.H. was employed by the company State Grid Shanxi Integrated Energy Service Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## Abbreviations

| | |
|---|---|
| UAV | unmanned aerial vehicle |
| EMA | Efficient Multi-Scale Attention |
| SCConv | Spatial and Channel Reconstruction Convolution |
| SE | Squeeze-and-Excitation |
| CBAM | Convolutional Block Attention Module |
| ECA | Efficient Channel Attention |
| CA | Coordinate Attention |
| SimAM | Simple Attention Module |

## References

1. Huang, X.; Wu, Y.; Zhang, Y.; Li, B. Structural Defect Detection Technology of Transmission Line Damper Based on UAV Image. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–14. [CrossRef]
2. Ji, C.; Jia, X.; Huang, X.; Zhou, S.; Chen, G.; Zhu, Y. FusionNet: Detection of Foreign Objects in Transmission Lines During Inclement Weather. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 1–18. [CrossRef]
3. Tavara, S. Email Author; Parallel computing of support vector machines: A survey (Review). *ACM Comput. Surv.* **2019**, *51*, 123. [CrossRef]
4. Shakiba, F.M.; Azizi, S.M.; Zhou, M.; Abusorrah, A. Application of machine learning methods in fault detection and classification of power transmission lines: A survey. *Artif. Intell. Rev.* **2023**, *56*, 5799–5836. [CrossRef]
5. Zhu, J.; Guo, Y.; Yue, F.; Yuan, H.; Yang, A.; Wang, X.; Rong, M. A Deep Learning Method to Detect Foreign Objects for Inspecting Power Transmission Lines. *IEEE Access* **2020**, *8*, 94065–94075. [CrossRef]
6. Yang, Q.; Ma, S.; Guo, D.; Wang, P.; Lin, M.; Hu, Y. A small object detection method for oil leakage defects in substations based on improved faster-rcnn. *Sensors* **2023**, *23*, 7390. [CrossRef]
7. Yin, L.; Zainudin, M.; Saad, W.; Sulaiman, N.; Idris, M.; Kamarudin, M.; Mohamed, R.; Razak, M. Analysis recognition of ghost pepper and cili-padi using mask rcnn and yolo. *Prz. Elektrotech.* **2023**, *2023*, 92. [CrossRef]
8. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
9. Redmon, J.; Farhadi, A. Yolov3: An incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
10. Bochkovskiy, A.; Wang, C.; Liao, H. Yolov4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
11. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
12. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. Yolov6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976.
13. Wang, C.; Bochkovskiy, A.; Liao, H. 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, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.

14. Wang, C.; Yeh, I.; Liao, H. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.

15. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.

16. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision-ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.

17. Huan, Z.; Qi, Q.; Jie, Z. Research on Bird nest detection method of transmission lines based on improved YOLOv5. *Power Syst. Prot. Control* **2023**, *51*, 151–159.

18. Sun, Y.; Li, J. Foreign body Detection Algorithm of YOLOv7-tiny Transmission Lines based on channel pruning. *J. Comput. Eng. Appl.* **2024**, *60*, 319–328.

19. Hao, Q.; Tao, Z.; Bo, Y.; Yang, R.; Xu, W. Transmission Line FaultDetection and Classification Based on Improved YOLOv8s. *Electronics* **2023**, *12*, 4537. [CrossRef]

20. Wang, Y.; Feng, L.; Song, X.; Qu, Z.; Yang, K.; Wang, Q.; Zhai, Y. TFD-YOLOv8: A Foreign body detection method for transmission lines. *J. Graph.* **2024**, *45*, 91.

21. Chen, J.; Kao, S.; He, H.; Zhuo, W.; Wen, S.; Lee, C.-H.; Chan, S.-H.G. Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023.

22. Wang, J.; Zhang, F.; Zhang, Y.; Liu, Y.; Cheng, T. Lightweight Object Detection Algorithm for UAV Aerial Imagery. *Sensors* **2023**, *23*, 5786. [CrossRef]

23. Daliang, O.; Su, H.; Zhang, G.; Luo, M.; Guo, H.; Zhan, J.; Huang, Z. Efficient Multi-Scale Attention Module with Cross-Spatial Learning. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.

24. Li, J.; Wen, Y.; He, L. SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023.

25. Jin, H.; Liu, Q.; Chen, D. A comprehensive stochastic gradient descent Q-learning method with Adaptive learning rate. *J. Comput. Sci.* **2019**, *42*, 2203–2215.

26. Chen, Z.; Yang, J.; Feng, Z.; Zhu, H. RailFOD23: A dataset for foreign object detection on railroad transmission lines. *Sci. Data* **2024**, *11*, 72. [CrossRef]

27. Lin, T.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. *Comput. Vis.* **2014**, *8693*, 740–755.

28. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation net-works. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.

29. Woo, S.; Park, J.; Lee, J. CBAM: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.

30. Wang, Q.; Wu, B.; Zhu, P. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.

31. Hou, Q.; Zhou, D.; Feng, J. Coordinate Atte ention for EfficientMobile Network Design. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.

32. Yang, L.; Zhang, R.; Li, L.; Xie, X. SimAM: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks. In Proceedings of the International Conference on Machine Learning Research (PMLR), Virtual, 18–24 July 2021; pp. 11863–11874.

33. Selvaraju, R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual explanations from deep networks via gradient-based localization. *Int. J. Comput. Vis.* **2020**, *128*, 336–359. [CrossRef]