



Research article

Improved insulator location and defect detection method based on GhostNet and YOLOv5s networks

Jianjun Huang^{1,2}, Xuhong Huang^{1,2,3,*}, Ronghao Kang^{1,2}, Zhihong Chen^{1,2} and Junhan Peng^{1,2}

¹ School of Electronic, Electrical Engineering and Physics, Fujian University of Technology, Fuzhou 350118, China

² Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fuzhou 350011, China

³ National Demonstration Center for Experimental Electronic Information and Electrical Technology Education, Fujian University of Technology, Fuzhou 350108, China

* **Correspondence:** Email: huangxuhong@fjut.edu.cn.

Abstract: Outdoor, real-time, and accurate detection of insulator defect locations can effectively avoid the occurrence of power grid security accidents. This paper proposes an improved GhostNet-YOLOv5s algorithm based on GhostNet and YOLOv5 models. First, the backbone feature extraction network of YOLOv5 was reconstructed with the lightweight GhostNet module to reduce the number of parameters and floating point operations of the model, so as to achieve the purpose of being lightweight. Then, a 160×160 feature layer was added to the YOLOv5 network to extract more feature information of small targets and fuzzy targets. In addition, the introduction of lightweight GSConv convolution in the neck network further reduced the computing cost of the entire network. Finally, Focal-EIoU was introduced to optimize the CIoU bounding box regression loss function in the original algorithm to improve the convergence speed and target location accuracy of the model. The experimental results show that the parameter number, computation amount, and model size of the GhostNet-YOLOv5s model are reduced by 40%, 25%, and 36%, respectively, compared with the unimproved YOLOv5s model. The proposed method not only ensures the precision of insulator defect detection, but also greatly decreases the complexity of the model. Therefore, the GhostNet-YOLOv5s algorithm can meet the requirements of real-time detection in complex outdoor environments.

Keywords: insulator; YOLOv5; defect detection; lightweight; GhostNet

1. Introduction

In recent years, China's new energy vehicle industry has developed at a very fast speed under the promotion of government policies, and people's demand for electricity has become increasingly large. Therefore, in the case of huge power supply, it is very important to ensure the safe, stable, and reliable operation of the power grid for the whole power system. The complex power system will inevitably have various failures, and the safety accidents caused by insulator failure account for the highest proportion. This is due to the fact that insulators are exposed to harsh natural environments such as rain, snow, and intense sunlight over long periods, which can lead to the deterioration of their external structure and subsequently shorten their service life [1, 2]. Insulators play a critical role in power transmission lines, and research on the localization and identification of defects in insulators has garnered significant attention [3]. Currently, drone aerial photography technology is widely used in the inspection of transmission lines, which allows for the rapid acquisition of a large number of insulator images in a short period. However, manually inspecting and screening these images for defects would be extremely time-consuming [4]. Therefore, it is essential to develop an efficient detection method to process the images captured by drones, enabling real-time localization of insulator defects.

At present, as the application scenarios of deep learning technology become more and more extensive, many researchers also apply this technology in insulator defect detection and location [5]. Insulator target detection algorithms based on deep learning can be divided into two categories. The first category consists of two-stage object detection algorithms based on candidate regions, with representative algorithms including the fast region-based convolutional neural network (Fast R-CNN) [6] and Faster R-CNN [7], etc. The literature [8] proposed a method that combines the region partitioning network (RPN) with Faster R-CNN to form an attention mechanism, which further improves detection accuracy. However, due to the increased network complexity, this approach results in slower inference speeds. The work in [9] introduced an improved Faster R-CNN model for the classification and detection of catenary insulators. This model employed the Inceptionv2 network for feature extraction and utilizes both softmax and Src for cascading, thereby enhancing the accuracy of insulator detection. In the literature [10], the random forest classification method was used for image segmentation, and then a convolutional neural network was used to classify normal and defective insulators. Finally, Faster R-CNN was used to recognize self-detonation defects of insulators. The work in [11] replaced the feature extraction network of Faster R-CNN with Resnet101 to improve detection accuracy, but its ability to locate small objects was limited. The work in [12] replaced the backbone network of Faster R-CNN with the lightweight EfficientNet and constructed feature pyramids with different resolutions to achieve feature fusion, thereby enhancing the detection ability for objects of different scales, especially for small objects.

The second type is a one-stage object detection algorithm based on regression. One-stage detection algorithms usually use neural networks to extract advanced features in data and fuse feature maps to achieve target positioning [13]. Examples include the single shot detector (SSD) [14] and You Only Look Once (YOLO) [15] neural networks. In the literature [16], a lightweight SSD target recognition network was designed. A lightweight MobileNet [17] was used to replace the backbone feature extraction network in the original network, and then a multi-model fusion algorithm was used to realize insulator self-explosion fault detection. The work in [18] proposed an improved YOLOv3 [19] algorithm, which improves the speed and accuracy of insulator defect detection by introducing the

non-parametric attention mechanism SimAM into the backbone network. In the literature [20], a lightweight MobileNet is combined with the backbone network of CenterNet [21], and the convolutional block attention module (CBAM) is introduced to enhance the accuracy of predicting the location of small insulator targets. In the literature [22], the GhostNet [23] module was used to reconstruct the backbone network of the original YOLOv4 [24], and an ECA-Net channel attention mechanism was introduced to improve the detection ability of the model on small targets. The literature [25] introduced the ShuffleNetv2 [26] module into YOLOv5 [27] as the backbone feature extraction network, and then introduced a multi-scale feature fusion network and a fourth prediction head to enhance the network's ability to sense small targets with insulator defects. In the literature [28], to reduce the impact of uneven outdoor lighting on insulator detection, a method based on YOLOv5 was proposed. This method employs image enhancement techniques, including illumination correction and compensation, to improve the contrast and detail of the images. In the literature [29], a CBAM attention mechanism is introduced into YOLOv5 to obtain the space and channel weight coefficients, and the dimension of input feature mapping is transformed to enhance the model's ability to extract and fuse insulator defect features. The aforementioned improvement methods have optimized detection models for various application scenarios, but most of them are difficult to achieve the balance between the detection accuracy of insulator defects and the model's lightweight design. Furthermore, the background environment surrounding insulators is often complex, and most defects are small in size, which significantly increases the difficulty of detecting insulator defects.

To further address the challenges mentioned above, we propose an improved algorithm called GhostNet-YOLOv5s, which is based on the integration of GhostNet and the YOLOv5 model. The specific advantages and contributions of the proposed algorithm are as follows:

- 1) Reconstructs the lightweight backbone feature extraction network. A lightweight GhostNet feature extraction network is bottleneck combined with the Ghost bottleneck and GhostConv module, which reduces the number of parameters, computation, and model size of the network models. This optimization of reducing network complexity greatly facilitates the edge deployment of mobile embedded devices, improves the real-time detection capability of the mobile devices, and saves the computing memory consumption of the devices.

- 2) Adding a 160×160 feature layer is conducive to detecting small target objects, and extracting more fine-grained feature information of small targets. However, in order to avoid too much redundant information, the 160×160 feature layer of the feature pyramid network output no longer performs YOLO Head prediction output. In addition, GSConv [30] is introduced again to optimize the traditional convolution and save computing costs.

- 3) In order to further optimize the original bounding box loss function, a new Focal-EIoU loss function is constructed by combining the idea of Focal to detect difficult targets with EIoU, which is sensitive to positioning accuracy. Compared with the original loss function, Focal-EIoU has higher positioning accuracy and faster convergence speed.

2. Original YOLOv5s and improved YOLOv5s

2.1. Original YOLOv5s model

YOLOv5 is a single-stage object detection network, primarily composed of four key structures: input, backbone, neck, and head. According to the network model from large to small, YOLOv5 can

be subdivided into YOLOv5x, YOLOv5l, YOLOv5m, YOLOv5s, and YOLOv5n. The overall network structure of each version is the same, but the difference lies in the depth and width of the modules used in the configuration file. Although larger network models have higher detection accuracy, the overall network structure will be more complex, the detection speed will be slower, and the device memory overhead will be larger. Considering that the model needs to be deployed to mobile devices and can meet the application requirements of real-time detection, we choose the YOLOv5s version with a relatively small number of parameters and model size as the improved basic model. The structure of the YOLOv5 network is shown in Figure 1.

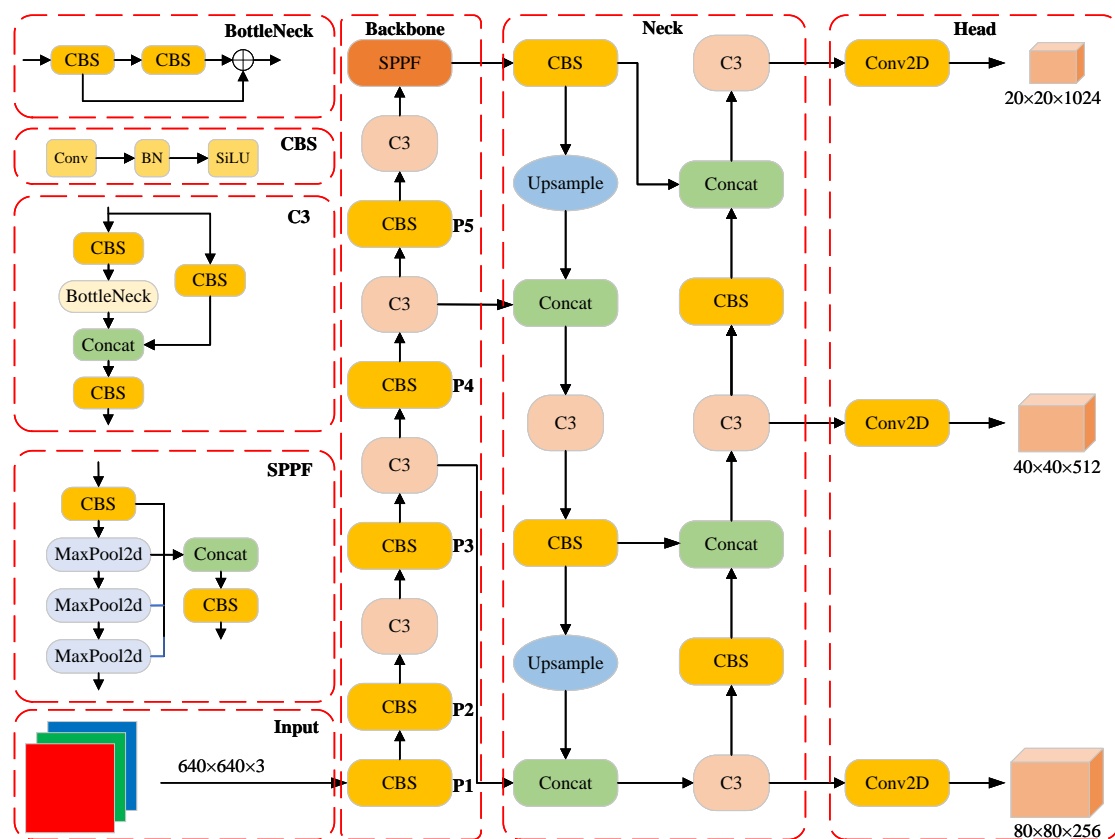


Figure 1. The structure of the YOLOv5 network.

The backbone network of YOLOv5 adopts CSPDarknet53 as the feature extraction network, which consists of Conv, C3, and SPPF. The front and back connection of Conv and C3 enhance the feature extraction ability of the network. The C3 module not only transmits the feature information to the lower layer, but also to the Concat operation with other layer structures of the neck network, which strengthens the information exchange between different feature layers. SPPF serializes inputs through multiple 5×5 sized maximum pooling layers, which can enhance the receptive field of the network and obtain some significant features. The neck network mainly fuses extracted feature information, which is composed of the feature pyramid network (FPN) and path aggregation network (PAN). FPN fuses feature information from top to bottom, while PAN transfers feature information from the bottom layer

to the high-level network. The introduction of FPN and PAN enables network models to detect targets more accurately on the feature maps of different scale. The head network will predict and evaluate the feature maps of three different scales, and obtain the specific location and category information of the target in the detection image. Finally, when the target to be detected is predicted, the candidate box with the highest confidence is reserved by using the non-maximum suppression (NMS) method.

2.2. Improved YOLOv5s network model

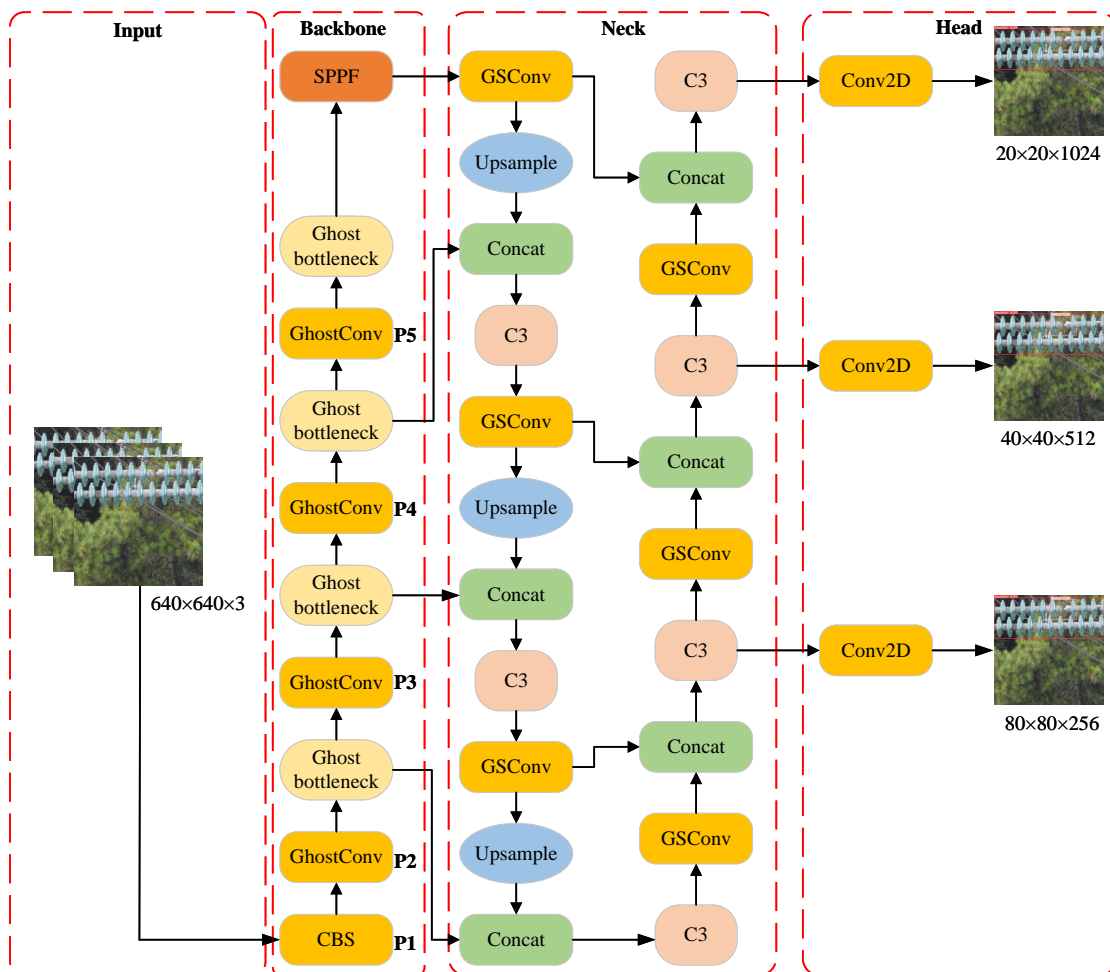


Figure 2. Network structure of GhostNet-YOLOv5s.

To realize the real-time detection of insulator defects, this paper proposes a lightweight network suitable for this study, which is called GhostNet-YOLOv5s. First, the backbone feature extraction network of YOLOv5s is reconstructed using the GhostNet lightweight network, which replaces C3 and ordinary convolution with Ghost bottleneck and GhostConv modules, respectively, which reduces the number of parameters and computation amount of the network. Second, a 160×160 small target detection layer is constructed in the neck network, which is conducive to improving the detection effect of small targets. At the same time, we also introduce the GSConv lightweight convolution in

the neck network. Finally, to cluster more high-quality anchor frames, the Focal-EIoU loss function is used to improve the positioning accuracy of detection targets. Figure 2 shows the network structure of GhostNet-YOLOv5s.

Due to factors such as the limited pixel size of small insulator defects, varying distances in UAV aerial photography, and the complexity of the image background, directly using YOLOv5s for insulator defect detection may result in a significant number of missed detections and false positives. Therefore, this paper introduces a 160×160 scale feature layer that is conducive to small target detection to achieve fine-grained image detection and extract more details of small targets. That is, the 160×160 scale feature layer is obtained by upsampling again behind the 80×80 scale feature layer of the neck network. The maximum sensitivity field of P5 is suitable for large target detection, and P4 is suitable for medium target detection. Based on P3, upsampling is carried out, and the P2 feature layer is fused to realize the defect detection of small targets.

However, adding a 160×160 feature layer responsible for small target detection in the network model will inevitably increase the number of network parameters and computational complexity. To further compress the parameters and FLOPs of the network, GSConv lightweight convolution is introduced into the PAN and FPN structure of the neck network, that is, GSConv is used instead of the complex traditional convolution. Finally, due to the problem of sample imbalance in the process of bounding box regression, the CIoU loss function used in the original YOLOv5s is improved into Focal-EIoU, which improves the convergence speed of model training without increasing the computational cost.

2.2.1. Reconstructing the backbone network

The YOLOv5s model primarily extracts feature information through traditional convolution operations within the CSPDarknet53 network. The working principle of traditional convolution is to use convolutional kernels of different sizes and apply operations such as upsampling and pooling on the feature map with specified strides to extract effective features, as shown in Figure 3(a). However, when traditional convolution works on feature maps, adjacent convolution kernels may extract a large amount of redundant information, which will affect the processing speed of feature information in the whole network. Therefore, the traditional convolutional network model has a complex structure, a large number of parameters and FLOPs, and high performance requirements for hardware devices.

To address the issues associated with traditional convolution, Huawei's Noah's Ark Lab proposed GhostNet [23]. GhostNet introduces the Ghost module, which generates a portion of the feature maps using a small number of traditional convolutions and then produces additional feature maps through linear transformations. This approach reduces both computational cost and the number of parameters. Compared to other lightweight networks like MobileNet [17] and ShuffleNetv2 [26], GhostNet typically achieves higher accuracy with the same level of resource consumption. Therefore, this paper reconstructs the backbone network using the GhostNet module, which can extract redundant feature information and simplify the computation process of the network with minimal computational resources, without compromising the overall performance of the model. As shown in Figure 3(b), GhostConv adopted in the Ghost module implements feature extraction in two steps. First, the traditional convolution operation is performed on the input image to obtain m ($m < n$) traditional feature maps with compressed channels. Then $s-1$ new feature maps are obtained by linear transformation Φ_i . Finally, the different feature maps are spliced to get n feature outputs.

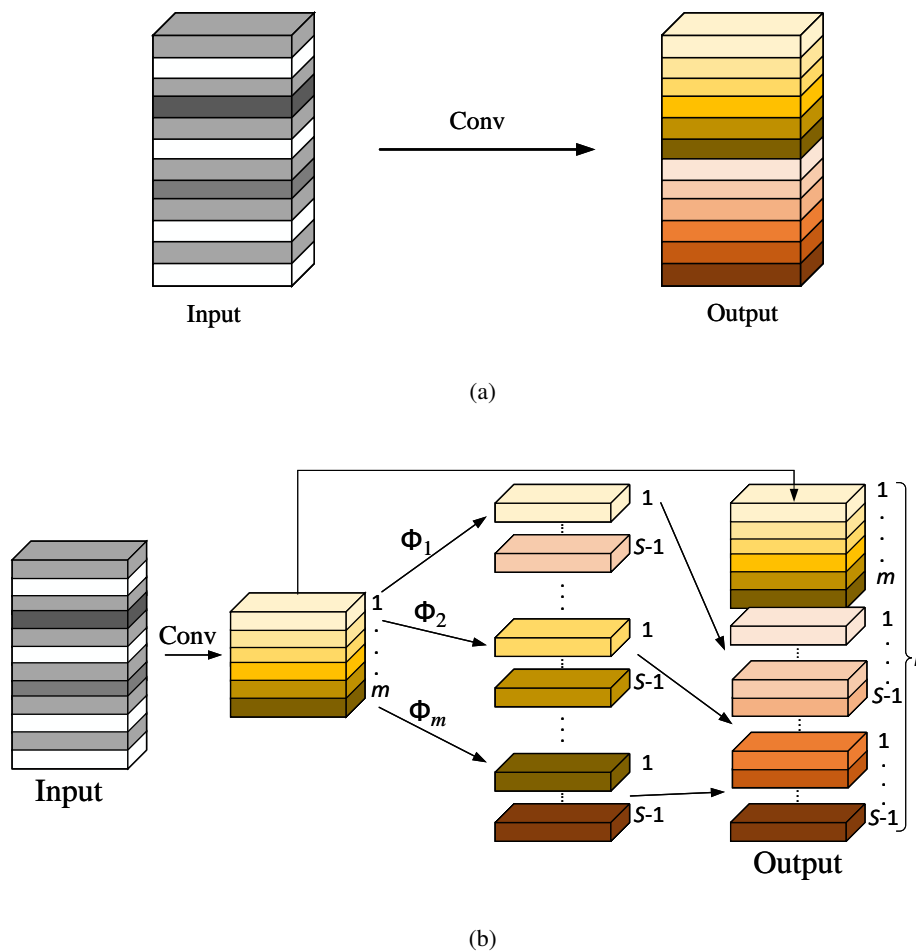


Figure 3. Structure diagram of different convolution modes. (a) Traditional convolution; (b) Ghost module.

For a clearer comparison of the computational magnitude of traditional convolutions with GhostConv, we assume that a convolution kernel with a size of $k \times k$ is adopted, the size of the input image is $C \times H \times W$, the size of the output feature map is $c \times h \times w$, and the number of output channels is $c = m + m(s - 1)$. Therefore, the computational amount required by traditional convolution is $C \times k \times k \times c \times h \times w$. However, in the Ghost convolution process, the computational amount required to generate m traditional feature graphs is $C \times k \times k \times m \times h \times w$ and the computational amount required to generate $m \times (s - 1)$ new feature graphs is $m \times k \times k \times (s - 1) \times h \times w$, so the computational amount of GhostConv is $C \times k \times k \times m \times h \times w + m \times k \times k \times (s - 1) \times h \times w$. Since $C \gg s$ is satisfied in the convolution calculation process, theoretically, the computational amount of GhostConv is only $1/s$ times that of traditional convolution. This reduces the amount of storage required for inference, helping to run on low-memory devices. The computational amount ratio of the two convolutions can be expressed as Eq (2.1).

$$\frac{F_1}{F_2} = \frac{C \times k \times k \times c \times h \times w}{C \times k \times k \times m \times h \times w + m \times k \times k \times (s - 1) \times h \times w} = \frac{C \times s}{C + s - 1} \approx s \quad (2.1)$$

where, F_1 and F_2 are the floating point computation amount of traditional convolution and GhostConv,

respectively; w and h are the width and height of the output feature graph, respectively; C is the number of input image channels; c is the number of output feature graph channels; $k \times k$ is the size of the convolution kernel; s is the number of new feature graphs generated by GhostConv; and m is the number of feature graphs generated by traditional convolution.

2.2.2. GSConv convolution

In object detection tasks, lightweight networks are typically constructed using a large number of depthwise separable convolutions to reduce the number of parameters and computational complexity. However, depthwise separable convolutions process each channel's information independently, which results in the loss of many hidden connections, leading to lower feature extraction and fusion capabilities compared to traditional convolutions. To address the shortcomings of depthwise separable convolutions, researchers have introduced the GSConv [30] convolutional module. The GSConv module consists of three parts: traditional convolution, depthwise separable convolution, and shuffle, as shown in Figure 4.

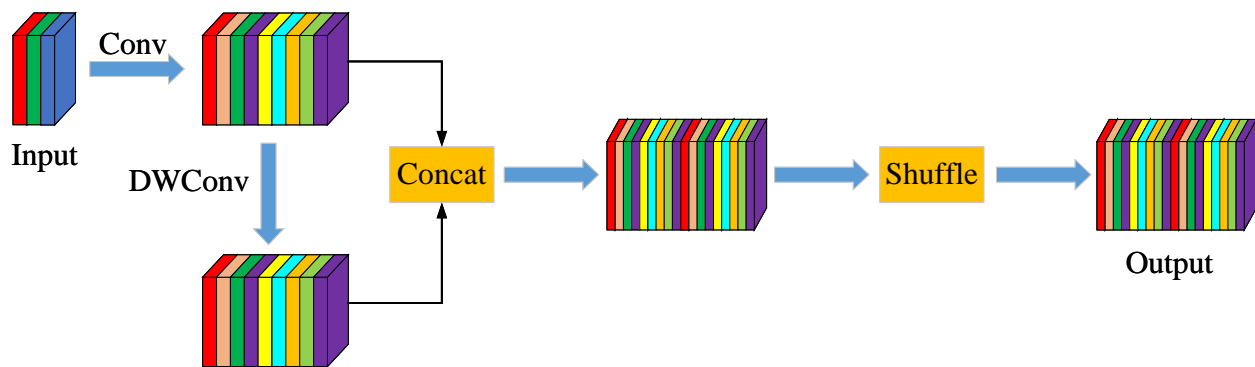


Figure 4. The structure of the GSConv module.

The GSConv module first generates a feature map with the number of parameters halved through traditional convolution, and then uses depthwise separable convolution to generate a new feature map. Then, these two sets of different feature layers are Concat combined, and finally, this feature information is infiltrated into the depthwise separable convolution through the shuffle operation, so that the information between channels can be exchanged. The convolution calculation of this operation method is close to the output of traditional convolution, but the calculation cost is reduced. The deep separable convolution layer and shuffle layer structure of the GSConv module enhance the nonlinear expression ability of feature information, making the GSConv module more suitable for lightweight detection models. The floating-point computation of traditional convolution is Eq (2.2), the floating-point computation of depth-separable convolution is Eq (2.3), and the floating-point computation of GSConv convolution is Eq (2.4).

$$FLOPs_1 = W \times H \times k \times k \times C \times c \quad (2.2)$$

$$FLOPs_2 = W \times H \times k \times k \times 1 \times c \quad (2.3)$$

$$FLOPs_3 = W \times H \times k \times k \times 1 \times \frac{c}{2}(C + 1) \quad (2.4)$$

where, W and H are the width and height of the feature map respectively; $k \times k$ is the size of the convolution kernel; and C and c are the number of input and output feature channels, respectively.

According to the analysis of the above three formulas, as the number of feature channels gradually increases, the computational amount of GSConv convolution is approximately half of that of ordinary convolution, but its feature extraction ability is the same as that of ordinary convolution. Furthermore, compared to depthwise separable convolutions, GSConv offers superior computational efficiency and flexibility. It performs exceptionally well in tasks that require maintaining high accuracy while reducing computational overhead. Therefore, the introduction of GSConv convolution in the model reduces the computation and parameter number, and improves the network running speed.

2.2.3. Improvement of loss function

The object detection algorithm usually adopts IoU to evaluate the overlap degree between the predicted box and the ground-truth box, while YOLOv5 uses the optimized CIoU [31] loss function to calculate the loss value. CIoU takes into account such variables as the aspect ratio between the ground-truth box and the predicted box, the center point distance, and the overlap area, which effectively improves the detection capability of the model. The specific calculation formula is shown in Eq (2.5).

$$\begin{cases} Loss_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{C^2} + \alpha v \\ \alpha = \frac{v}{(1-IoU)+v} \\ v = \frac{4}{\pi^2} (\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h})^2 \end{cases} \quad (2.5)$$

where, b and b^{gt} are the central points of the prediction box and the ground-truth box, respectively; ρ is the Euclidean distance between two central points; C is the diagonal distance of the minimum closure area that can contain both the prediction box and the ground-truth box; α is the trade-off parameter; and v indicates the length-width ratio of the target box.

The structure diagram of CIoU is shown in Figure 5, which mainly makes the model converge faster by optimizing the distance between the two target boxes. However, since the difference in aspect ratio is reflected by the parameters in Eq (2.5), rather than the actual variation in confidence, effective optimization of similarity is sometimes hindered.

In summary, traditional loss functions like IoU and CIoU have the following limitations: 1) they require combination with other loss functions to handle class imbalance effectively; 2) they exhibit slow convergence when dealing with targets that have little overlap; and 3) they are prone to inaccurate localization in complex scenarios. To address these issues, this paper adopts the more advanced Focal-EIoU [32] to optimize the YOLOv5 loss function, as it demonstrates superior performance in handling class imbalance, improving localization accuracy, and accelerating training convergence. Focal-EIoU separates the aspect ratio and introduces the Focal idea to cluster more high-quality anchor frames. Its mathematical expression is shown as Eq (2.6). The Focal-EIoU loss function mainly includes overlap loss, center distance loss, and width and height loss. The difference between it and CIoU is that width and height loss is considered. The purpose of introducing width and height loss is to lessen the gap between the width and height of the two target boxes, so the convergence speed of the target detection algorithm can be further improved.

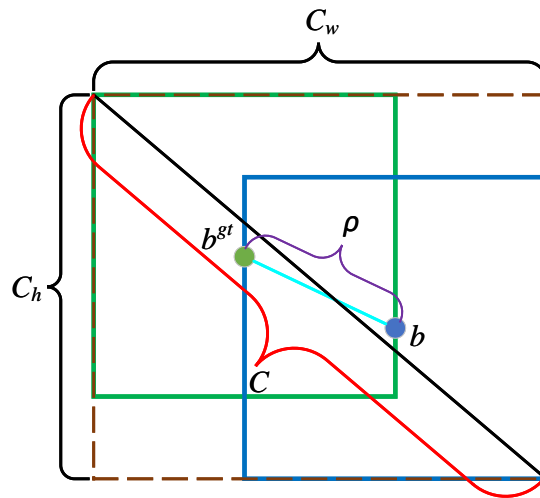


Figure 5. CIoU calculation structure diagram.

$$\begin{cases} Loss_{EIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{C^2} + \frac{\rho^2(w, w^{gt})}{C_w^2} + \frac{\rho^2(h, h^{gt})}{C_h^2} \\ Loss_{Focal-EIoU} = IoU^\gamma \times Loss_{EIoU} \end{cases} \quad (2.6)$$

In Eq (2.6), γ is the parameter that controls the suppression degree of the outliers, and the general value is 3. C_w and C_h are the width and height of the minimum external rectangular box, respectively.

3. Experiment and analysis

3.1. Experimental environment and data collection

The experimental operating platform of this paper is the Ubuntu 18.04 system, using Pytorch1.9.0 as the deep learning framework, Python 3.8, and Intel(R) Xeon(R) Platinum 8255C CPU @2.50 GHz. The GPU model was NVidia GeForce RTX 3080, and the graphics card memory was 10 GB. The initial learning rate of the experiment was set as 0.01, the SGD optimizer was used to update the network parameters, the learning momentum was 0.937, the weight decay rate was 0.0005, the batch size was 16, and the number of training epochs was 200.

The experimental dataset in this paper comes from two parts: the Chinese Power Line Insulator Dataset (CPLID) and the glass insulator defect picture provided by Question B of the 8th ‘‘Teddy Cup’’ (<https://www.tipdm.org/>). The CPLID dataset consists of 600 images of normal insulators and 248 images of defective insulators, all captured by UAVs, with each image having a resolution of 1152×864 . The dataset from the 8th ‘‘Teddy Cup’’ Problem B contains 40 high-resolution images of glass insulators. These high-resolution images may reduce detection speed, so the solution adopted in this paper is to divide each image into a 4×4 grid and remove images that do not contain insulators, resulting in 286 usable images. Because the number of defective insulators in the dataset is too small, in order to avoid the imbalance of categories affecting the insulator defect detection effect, data enhancement is used to

expand the data. By means of horizontal flip, rotation, Gaussian blur and random pixel removal, the image is expanded to 5228, which effectively increases the size of the training set and improves the generalization ability of the model. The effect diagram after image enhancement is shown in Figure 6. In this paper, the self-built dataset is randomly divided into a training set, validation set, and test set according to the ratio of 6:2:2, and two detection labels are set, namely “insulator” and “defect”.

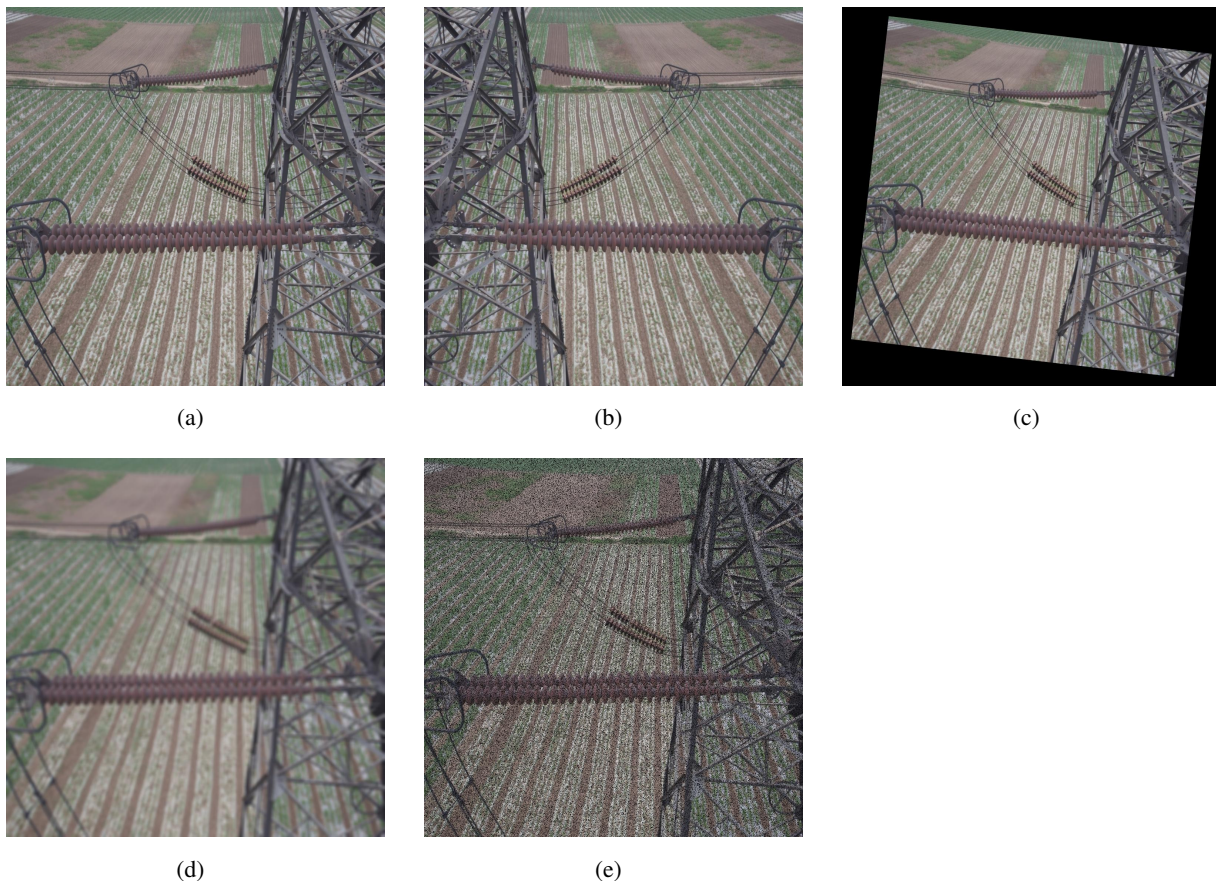


Figure 6. Image enhancement effect. (a) Original; (b) Horizontal flip; (c) Rotation; (d) Gaussian blur; (e) Random pixel removal.

3.2. Evaluation metrics

It is necessary to evaluate the experimental results objectively and accurately. This paper mainly adopts average precision (AP), mean average precision (mAP), number of parameters, FLOPs, model size, and frames per second (FPS) to appraise the detection performance of the trained model. The average precision is determined by recall (R) and precision (P) and is an intuitive measure of performance results for a single category. The calculation formulas of R and P are, respectively, Eqs (3.1) and (3.2). With the R as the horizontal coordinate and the maximum P rate corresponding to each recall rate as the vertical coordinate, a P - R curve is drawn, and the AP value is the integral area under the curve. The calculation formula is Eq (3.3). After obtaining multiple AP values of a single category, the mAP

value is obtained by averaging them. The calculation formula of mAP value is Eq (3.4).

$$R = \frac{T_P}{T_P + F_N} \times 100\% \quad (3.1)$$

$$P = \frac{T_P}{T_P + F_P} \times 100\% \quad (3.2)$$

$$AP = \int_0^1 P(R) dR \quad (3.3)$$

$$mAP = \frac{\sum_1^N AP_i}{N} \quad (3.4)$$

where, T_P indicates that the detection result of the positive sample is a positive sample, F_N indicates that the detection result of a positive sample is a negative sample, and F_P indicates that the detection result of a negative sample is a positive sample. N is the number of categories of targets to be detected in the dataset. In this experiment, $N = 2$, that is, normal insulators and defective insulators.

3.3. Comparative experiment of different lightweight methods

In order to verify the optimal performance of the selected lightweight backbone network on the self-built insulator dataset, this paper introduced three mainstream lightweight backbone networks to replace the original YOLOv5 backbone network, namely the MobileNetv3, ShuffleNetv2, and GhostNet networks. The test results of different backbone networks are shown in Table 1. As can be seen from Table 1, compared with the original CSPDarknet53 backbone network, the mAP@0.5, parameter number, FLOPs, and model size of the ShuffleNetv2 lightweight backbone network decreased by 2.1%, 3.03 M, 8.3 G, and 5.8 MB, respectively. The mAP@0.5, parameter number, FLOPs, and model size of the MobileNetv3 lightweight backbone network decreased by 3.5%, 2.55 M, 8.8 G, and 4.9 MB, respectively. The mAP@0.5, parameter number, FLOPs, and model size of the GhostNet lightweight backbone network decreased by 1.1%, 2.43 M, 6.3 G, and 4.6 MB, respectively. The change curves of mAP@0.5 and boundary box regression loss value for different lightweight improvement methods are shown in Figure 7. Through the comprehensive analysis of this experiment, it can be found that mAP@0.5, parameter number, FLOPs, model size, and FPS of the GhostNet lightweight backbone network are the best balanced among the three networks, and the bounding box regression loss is the least. In order to enable the backbone network to obtain more useful information when extracting features, and avoid the serious loss of accuracy caused by the excessive use of a lightweight backbone network, in this paper, GhostNet was used to reconstruct the backbone network of YOLOv5s, which ensured the accuracy of model detection while reducing the complex construction of the model, and met the detection requirements of embedded mobile devices.

Table 1. Comparative experiments of different lightweight methods.

Backbone	Precision (%)	Recall (%)	mAP@0.5 (%)	Parameters (M)	Flops (G)	Model size (MB)	FPS (f/s)
CSPDarknet53	97.0	88.2	93.0	7.02	15.8	13.8	107.5
ShuffleNetv2	94.8	85.6	90.9	3.99	7.5	8.0	99.0
MobileNetv3	95.1	83.0	89.5	4.47	7.0	8.9	86.9
GhostNet	96.5	86.7	91.9	4.59	9.5	9.2	101.1

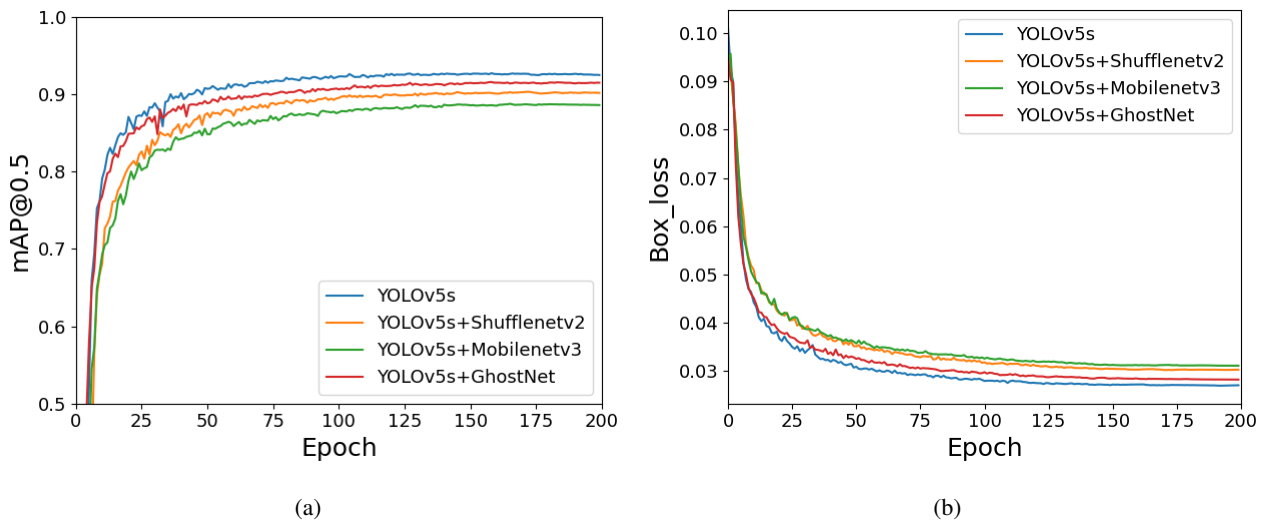


Figure 7. Comparison experiment curve of different lightweight methods. (a) The change in curve in mAP@0.5; (b) The change in curve in loss value.

3.4. Ablation experiment

To further verify the effectiveness of the improved method in this paper, an ablation experiment of the improved network model was conducted, and a total of 5 groups of experiments were set up. The ablation experiment results are shown in Table 2. Compared with Group 1, after using the GhostNet lightweight backbone network to reconstruct CSPDarknet53 in Group 2, mAP@0.5 decreased by 1.1%, but the parameters, FLOPs, and model size decreased by 34.6%, 39.9%, and 33.3%, respectively. Group 3 introduces GSConv convolutions with less computational effort into the neck network. Compared with Group 2, mAP@0.5 increases by 0.3%, and the number of parameters, FLOPs, and model size decrease by 9.8%, 5.3%, and 9.8%, respectively, indicating that GSConv convolutions can not only compress the computational cost, but also enhance the detection precision of the model. However, using lightweight modules may sacrifice some detection accuracy. Therefore, in Group 4, a 160×160 feature layer conducive to fine-grained target detection was introduced on the basis of Group 3 to more accurately identify small insulator targets. Although the parameters and FLOPs of the model are slightly increased compared with Group 3, mAP@0.5 is increased by 0.5%, which fully indicates that the 160×160 scale feature layer remarkably improves the detection capability of the model. Finally, Group 5 used Focal-EIoU loss to optimize CIoU loss, and further improved the positioning precision and convergence speed of the model as mAP@0.5 reached 93.1%. The ablation experiments from Group 2 to Group 5 show that each improved method has the role of the optimization model. Compared with YOLOv5s in Group 1, the mAP@0.5 of the improved GhostNet-YOLOv5s model (Group 5) was increased by 0.1%, and the parameters, FLOPs, and model size were dropped by 39.3%, 27.8%, and 37.7%, respectively. Although the FPS dropped by 6.4 frames, the FPS of 101.1 frames can still meet the needs of drone aerial photography detection. The comparison curve of ablation experimental results in this paper is shown in Figure 8. It can be seen from Figure 8(a) that each improvement method adopted gradually improves the mAP@0.5 index, and it can be seen from Figure 8(b) that the boundary box regression loss value of the proposed method is minimal, indicating that adopting

Focal-EIoU to optimize the loss function can indeed improve the positioning precision of the model.

Table 2. Results of ablation experiments.

Group	Model	mAP@0.5 (%)	Parameters (M)	FLOPs (G)	Model size (M)	FPS (f/s)
1	YOLOv5s	93.0	7.02	15.8	13.8	107.5
2	YOLOv5s + GhostNet	91.9	4.59	9.5	9.2	101.1
3	YOLOv5s + GhostNet + GSConv	92.2	4.14	9.0	8.3	120.5
4	YOLOv5s + GhostNet + GSConv + 160×160 feature layers	92.7	4.26	11.4	8.6	98.0
5	YOLOv5s + GhostNet + GSConv+ 160×160 feature layers + Focal-EIoU	93.1	4.26	11.4	8.6	101.1

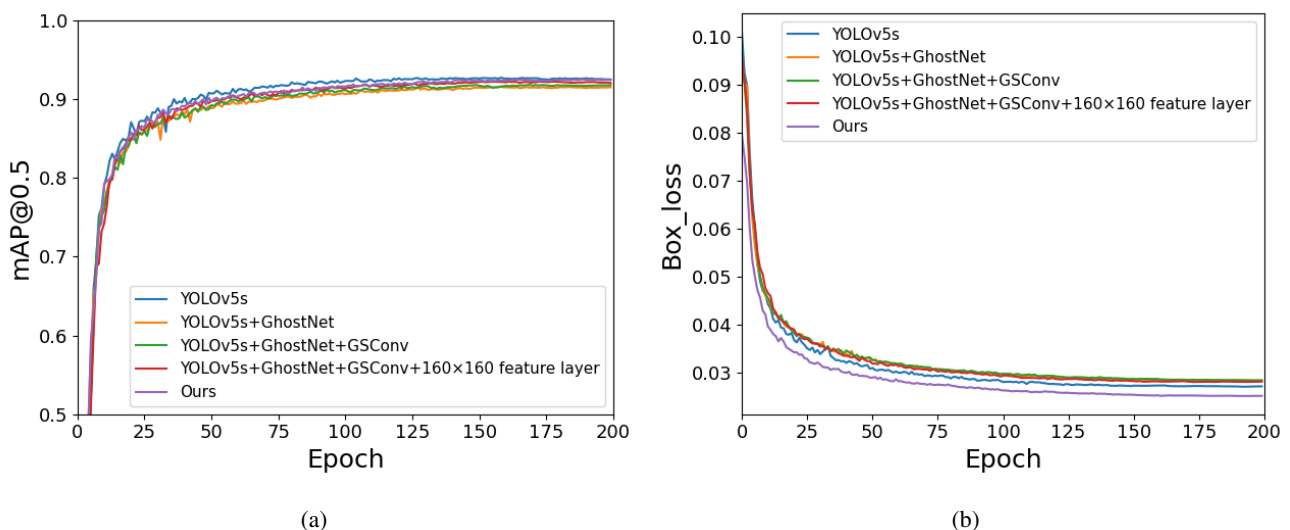


Figure 8. Comparison curve of ablation experimental results. (a) The change in curve in mAP@0.5; (b) The change in curve in loss value.

In order to show the detection effect of the improved model more directly, three insulator defect images with different backgrounds were selected in the test set to test the trained model. As shown in Figure 9, the detection results of the insulator images using the YOLOv5s and GhostNet-YOLOv5s models were visualized. The YOLOv5s algorithm model has a false detection of an insulator defect in the second image, and a false detection of an insulator defect and an overlapping false detection of an insulator in the third image. However, the improved GhostNet-YOLOv5s algorithm model in this paper can accurately locate the insulator and defect position in the aerial image. The problem of false detection and missing detection in insulator defect detection is solved effectively.

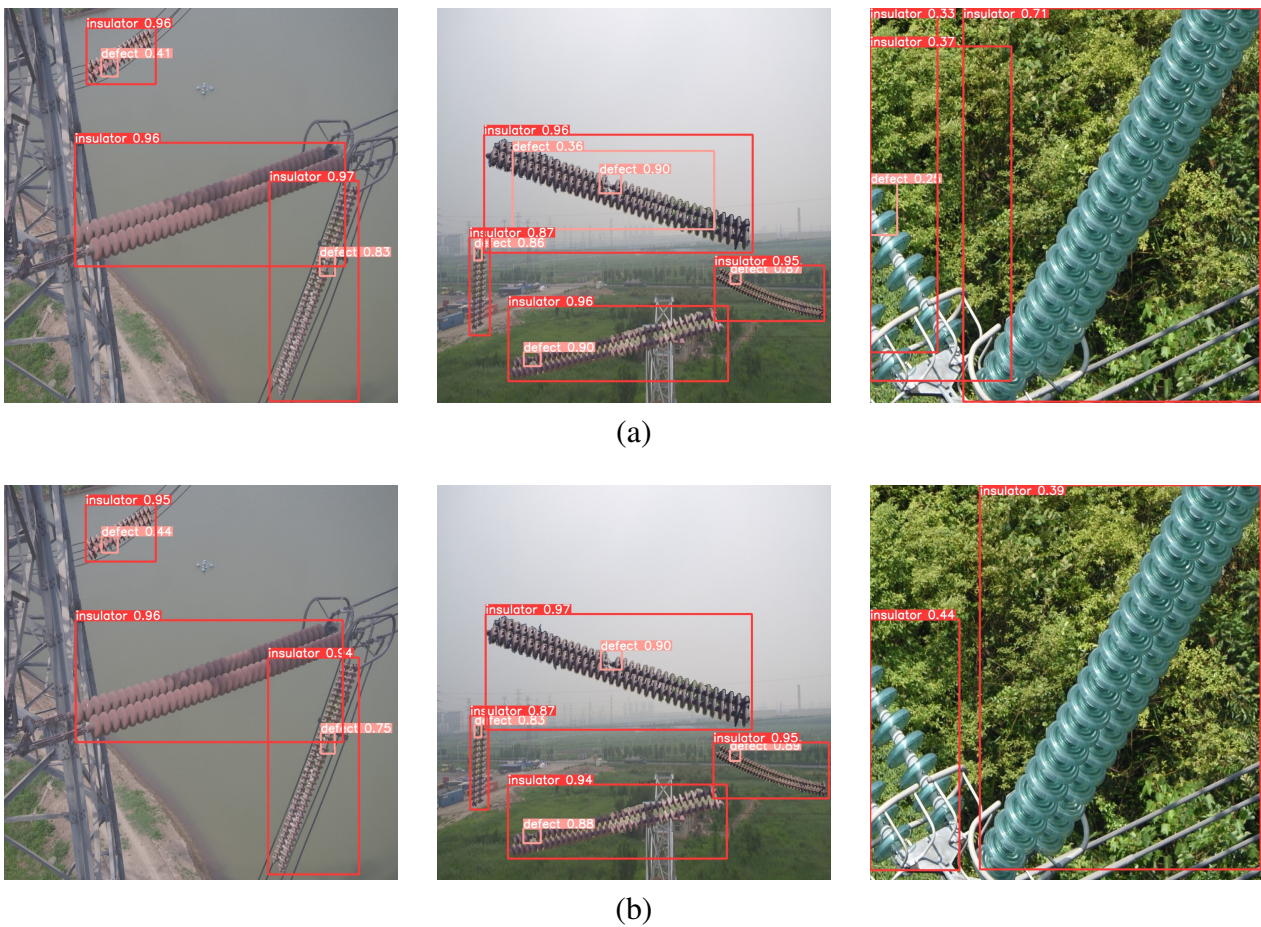


Figure 9. Comparison of effects before and after improvement. (a) Visual effects of YOLOv5s; (b) Visual effects of GhostNet-YOLOv5s.

3.5. Comparative experiments of different algorithms

To better evaluate the performance of the improved model in this paper, we selected different target detection algorithms to conduct comparative experiments in the same dataset, namely Faster-RCNN, SSD, YOLOv7-tiny [33], and YOLOv8s. As can be seen from Table 3, mAP@0.5 of GhostNet-YOLOv5s reaches 93.1%, the number of parameters is 4.26 M, the FLOPs is 11.4 G, and the model size is only 8.6 MB. It can be seen from the experimental results that compared with other algorithms, the GhostNet-YOLOv5s model has the least parameters, FLOPs, and model size. Although the mAP@0.5 of the improved algorithm in this paper is 1.1% and 0.2% lower compared to Faster R-CNN and YOLOv7-tiny, respectively, the parameter number, FLOPs, and model size are greatly reduced. Through the comparative experimental analysis of the above different algorithms, the improved algorithm proposed in this paper can take into account both precision and being lightweight, so that the model accuracy and weight can achieve a good balance effect, and further demonstrate the superiority of the GhostNet-YOLOv5s algorithm. GhostNet-YOLOv5s not only greatly reduces the number of parameters, FLOPs, and model size, but also further reduces the requirements for hardware configuration. Therefore, the improved model can meet the need of fast real-time detection of insulator defect

images taken by UAVs.

Table 3. Comparative experimental results of different algorithms.

Model	Precision (%)	Recall (%)	mAP@0.5 (%)	Parameters (M)	FLOPs (G)	Model size (MB)
Faster R-CNN	96.7	89.8	94.2	136.72	401.7	167.2
SSD	87.5	84.3	87.2	62.70	26.3	94.6
YOLOv5s	97.0	88.2	93.0	7.02	15.8	13.8
YOLOv7-tiny	95.9	88.0	93.3	5.73	13.0	11.7
YOLOv8s	96.0	88.4	93.0	10.64	28.4	21.5
GhostNet-YOLOv5s (Ours)	96.8	88.4	93.1	4.26	11.4	8.6

4. Conclusions

This paper presents a defect detection model of a lightweight insulator based on the GhostNet-YOLOv5s. First, the backbone feature extraction network was reconstructed with the lightweight GhostNet network. Although the detection accuracy is slightly reduced, the complexity of the network is greatly reduced. Then the traditional convolution of the neck network was replaced by the lighter GSConv module, which not only reduces the calculation cost of the model, but also improves the mAP@0.5 by 0.3%. In addition, a 160×160 feature layer was added to the neck network, which is conducive to fine-grained target detection, so as to enhance the feature extraction ability of the network for small targets and fuzzy targets, and further improves the detection performance of the model. Finally, the Focal-EIoU loss function was introduced to solve the problem of sample imbalance in boundary box regression, which effectively improved the convergence speed and target location accuracy of the model. Experiments show that, compared with YOLOv5s, GhostNet-YOLOv5s greatly reduces the number of network parameters and FLOPs, and saves the memory space occupation on the premise that mAP@0.5 is not reduced. Therefore, the algorithm proposed in this paper is conducive to the deployment of UAV equipment with limited memory space and computing resources, so that the UAV can complete the real-time detection of insulator defects during the transmission line inspection task.

However, due to the limited availability of publicly accessible insulator datasets, it is challenging to collect a diverse range of insulator defects that would effectively enhance model training. As a result, the detection ability of the proposed algorithm for untrained defects needs to be further improved. Additionally, the detection method employed in this study has certain limitations. For instance, the detection of heavily occluded targets may result in false positives or missed detections, and it is unable to directly detect internal damage of insulators. Therefore, future research can focus on the following aspects: 1) Introducing an attention mechanism to enable the model to better focus on the key parts of occluded targets. 2) Collecting infrared or thermal images to train a dedicated model specifically for detecting internal damage, which can then be combined with the object detection model to separately handle external and internal defect detection. Finally, in real-world scenarios, we plan to deploy the algorithm proposed in this paper on edge devices such as drones equipped with chips. The deployed system will be used for real-time defect detection, with the results being promptly fed back to operators or control centers to enable timely maintenance or repairs.

Use of AI tools declaration

The authors declare that they have not used artificial intelligence tools in the creation of this article.

Acknowledgments

This research was supported by the Research and Application Projects of Several Key Technologies of Intelligent Office Building Monitoring Systems (Grant No. KY350631).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. H. Liu, S. Geng, J. Wang, B. Xu, Y. Yang, L. Liang, Aging analysis of porcelain insulators used in UHV AC transmission line (in Chinese), *Insulators Surg. Arresters*, **310** (2022), 159–164. <https://doi.org/10.16188/j.isa.1003-8337.2022.06.023>
2. V. E. Ogbonna, P. I. Popoola, O. M. Popoola, S. O. Adeosun, A comparative study on the failure analysis of field failed high voltage composite insulator core rods and recommendation of composite insulators: A review, *Eng. Fail. Anal.*, **138** (2022), 106369. <https://doi.org/10.1016/j.engfailanal.2022.106369>
3. J. Chen, Z. Fu, X. Cheng, F. Wang, An method for power lines insulator defect detection with attention feedback and double spatial pyramid, *Electr. Power Syst. Res.*, **218** (2023), 109175. <https://doi.org/10.1016/j.epsr.2023.109175>
4. X. Luo, F. Yu, Y. Peng, UAV power grid inspection defect detection based on deep learning (in Chinese), *Power Syst. Prot. Control*, **50** (2022), 132–139. <https://link.cnki.net/doi/10.19783/j.cnki.pspc.211664>
5. X. Jia, Y. Yu, Y. Guo, Y. Huang, B. Zhao, Lightweight detection method of self-explosion defect of aerial photo insulator (in Chinese), *High Voltage Eng.*, **49** (2023), 294–300. <https://link.cnki.net/doi/10.13336/j.1003-6520.hve.20220334>
6. R. Girshick, Fast R-CNN, in *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, (2015), 1440–1448. <https://doi.org/10.1109/ICCV.2015.169>
7. S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, *IEEE Trans. Pattern Anal. Mach. Intell.*, **39** (2016), 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
8. L. Yao, N. Zhang, A. Gao, Y. Wan, Research on fabric defect detection technology based on EDSR and improved faster RCNN, in *Knowledge Science, Engineering and Management*, (2022), 477–488. https://doi.org/10.1007/978-3-031-10989-8_38
9. Z. Hu, X. Yao, Identification and extraction of catenary insulators based on improved faster-RCNN (in Chinese), *Insulators Surg. Arresters*, (2023), 146–152. <https://doi.org/10.16188/j.isa.1003-8337.2023.03.021>

10. P. Fan, H. M. Shen, C. Zhao, Z. Wei, J. G. Yao, Z. Q. Zhou, et al., Defect identification detection research for insulator of transmission lines based on deep learning, *J. Phys. Conf. Ser.*, **1828** (2021), 012019. <https://doi.org/10.1088/1742-6596/1828/1/012019>
11. H. Hu, J. Xu, Y. Huang, K. Wei, Defect detection of tower insulators based on improved Faster R-CNN transmission (in Chinese), *Inf. Technol. Informatization*, (2023), 63–66. <https://doi.org/10.3969/j.issn.1672-9528.2023.07.016>
12. Y. Chen, C. Deng, Q. Sun, Z. Wu, L. Zou, G. Zhang, et al., Lightweight detection methods for insulator self-explosion defects, *Sensors*, **24** (2024), 290. <https://doi.org/10.3390/s24010290>
13. Z. Na, L. Cheng, H. Sun, B. Lin, Survey on UAV detection and identification based on deep learning (in Chinese), *J. Signal Process.*, **40** (2024), 609–624. <https://doi.org/10.16798/j.issn.1003-0530.2024.04.001>
14. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, et al., SSD: Single Shot MultiBox Detector, in *Computer Vision – ECCV 2016*, (2016), 21–37. https://doi.org/10.1007/978-3-319-46448-0_2
15. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, **8** (2016), 779–788. <https://doi.org/10.1109/CVPR.2016.91>
16. B. Wei, Z. Xie, Y. Liu, K. Wen, F. Deng, P. Zhang, Online monitoring method for insulator self-explosion based on edge computing and deep learning, *CSEE J. Power Energy Syst.*, **8** (2022), 1684–1696. <https://doi.org/10.17775/CSEEJPES.2020.05910>
17. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, et al., MobileNets: Efficient convolutional neural networks for mobile vision applications, preprint, arXiv:1704.04861.
18. J. Li, L. Liu, Y. Niu, L. Li, Y. Peng, YOLOv3 identification method incorporating attention for insulator string (in Chinese), *High Voltage Appar.*, **58** (2022), 67–74. <https://doi.org/10.13296/j.1001-1609.hva.2022.11.009>
19. J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, preprint, arXiv:1804.02767.
20. H. Xia, B. Yang, Y. Li, B. Wang, An improved centerNet model for insulator defect detection using aerial imagery, *Sensors*, **22** (2022), 2850. <https://doi.org/10.3390/s22082850>
21. X. Zhou, D. Wang, P. Krähenbühl, Objects as points, preprint, arXiv:1904.07850.
22. G. Han, L. Zhao, Q. Li, S. Li, R. Wang, Q. Yuan, et al., A lightweight algorithm for insulator target detection and defect identification, *Sensors*, **23** (2023), 1216–1225. <https://doi.org/10.3390/s23031216>
23. K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, C. Xu, GhostNet: More features from cheap operations, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, **92** (2020), 1577–1586. <https://doi.org/10.1109/CVPR42600.2020.00165>
24. A. Bochkovskiy, C. Y. Wang, H. Y. M. Liao, YOLOv4: Optimal speed and accuracy of object detection, preprint, arXiv:2004.10934.
25. K. Chen, X. Liu, L. Jia, Y. Fang, C. Zhao, Insulator defect detection based on lightweight network and enhanced multi-scale feature fusion (in Chinese), *High Voltage Eng.*, **50** (2023), 1289–1300. <https://doi.org/10.13336/j.1003-6520.hve.20221652>

26. N. Ma, X. Zhang, H. T. Zheng, J. Sun, ShuffleNet V2: Practical guidelines for efficient CNN architecture design, in *Proceedings of the European Conference on Computer Vision (ECCV)*, (2018), 116–131.
27. N. Ma, X. Zhang, H. T. Zheng, J. Sun, YOLOv5 (accessed on 22 November 2022). Available from: <https://github.com/ultralytics/yolov5>.
28. Y. Li, M. Ni, Y. Lu, Insulator defect detection for power grid based on light correction enhancement and YOLOv5 model, *Energy Rep.*, **8** (2022), 807–814. <https://doi.org/10.1016/j.egy.2022.08.027>
29. D. Wei, B. Hu, C. Shan, H. Liu, Insulator defect detection based on improved Yolov5s, *Front. Earth Sci.*, **11** (2023), 1337982. <https://doi.org/10.3389/feart.2023.1337982>
30. H. Li, J. Li, H. Wei, Z. Liu, Z. Zhan, Q. Ren, Slim-neck by GSConv: A lightweight-design for real-time detector architectures, preprint, arXiv:2206.02424.
31. Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, et al., Enhancing geometric factors in model learning and inference for object detection and instance segmentation, *IEEE Trans. Cybern.*, **52** (2022), 8574–8586. <https://doi.org/10.1109/TCYB.2021.3095305>
32. Y. F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, T. Tan, Focal and efficient IOU loss for accurate bounding box regression, *Neurocomputing*, **506** (2022), 146–157. <https://doi.org/10.1016/j.neucom.2022.07.042>
33. C. Y. Wang, A. Bochkovskiy, H. Y. M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, BC, Canada, (2023), 7464–7475. <https://doi.org/10.1109/CVPR52729.2023.00721>



AIMS Press

©2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0>)