



# Insulator Defect Detection Based on the CDDCR–YOLOv8 Algorithm

Tingyao Jiang<sup>1</sup> · Xuan Hou<sup>1</sup> · Min Wang<sup>2</sup>

Received: 25 June 2024 / Accepted: 7 September 2024  
© The Author(s) 2024

## Abstract

Insulator defect detection is a critical aspect of grid inspection in reality, yet it faces intricate environmental challenges, such as slow detection speed and low accuracy. To address this issue, we propose a YOLOv8-based insulator defect detection algorithm named CDDCR–YOLOv8. This algorithm divides the input insulator images into multiple grid cells, with each grid cell responsible for predicting the presence and positional information of one or more targets. First, we introduce the Coordinate Attention (CA) mechanism module into the backbone network and replace the original C2f module with the enhanced C2f\_DCN module. Second, improvements are made to the original upsampling and downsampling layers in the neck network, along with the introduction of the lightweight module RepGhost. Finally, we employ Wise-IoU (WIoU) to replace the original CIoU as the loss function for network regression. Experimental results demonstrate that the improved algorithm achieves an average precision mean (mAP @ 0.5) of 97.5% and 90.6% on the CPLID and IPLID data sets, respectively, with a frame per second (FPS) of 84, achieving comprehensive synchronous improvement. Compared to traditional algorithms, our algorithm exhibits significant performance enhancement.

**Keywords** Object detection · Insulator defect detection · YOLOv8 · Attention mechanism

## 1 Introduction

Insulators are crucial components in power systems, and traditional methods for insulator defect detection primarily rely on manual inspections conducted by experienced personnel who visually assess whether defects exist [1, 2]. However, manual inspections suffer from subjectivity and fatigue errors, resulting in inefficient operations that struggle to meet the demands of large-scale power grids [3, 4]. Moreover, traditional approaches incorporate machine learning-based image processing techniques, such as image segmentation and feature extraction, to assist in detection [5]. Yet, these methods require manual feature design and

parameter tuning, leading to poor adaptability to complex backgrounds and diverse forms of defects, thus compromising detection accuracy [6, 7].

In recent years, the rapid advancement of deep learning technology has offered new solutions for insulator defect detection [8]. Deep learning, a neural network-based machine learning approach, automatically learns feature representations and pattern recognition from vast amounts of data [9]. Among these, object detection algorithms represent a significant application of deep learning in computer vision, enabling simultaneous localization and identification of objects in images with high accuracy and real-time performance [10].

Recent advancements in deep learning have significantly improved insulator defect detection, yet several significant problems remain. First, two-stage methods exhibit excellent detection accuracy but suffer from high computational complexity, making real-time application difficult [11, 12]. Second, one-stage methods, such as the YOLO series, offer speed advantages but fail in accuracy, particularly with complex backgrounds and small object detection. Furthermore, current object detection algorithms are limited in addressing the varied and complex nature of insulator defects [13].

---

✉ Xuan Hou  
202308540621011@ctgu.edu.cn

Tingyao Jiang  
jiangty@ctgu.edu.cn

Min Wang  
26974747@qq.com

<sup>1</sup> College of Electrical Engineering and New Energy, China Three Gorges University, Yichang 443002, China

<sup>2</sup> School of Electronic Information, Hubei Three Gorges Polytechnic, Yichang 443002, China

In response to these challenges, this paper presents an insulator defect detection method based on the CDDCR–YOLOv8 algorithm. Our main contributions include: first, we improved the structure and loss function of the YOLOv8 model to enhance its performance in detecting small targets in complex backgrounds. Second, we introduced various module mechanisms that dynamically adjust convolutional kernel parameters to enhance the model's feature extraction capabilities, thereby improving detection accuracy and robustness. Finally, extensive experiments have verified the effectiveness and superiority of the proposed algorithm in insulator defect detection tasks.

## 2 Related Works

With the continuous evolution and refinement of the YOLO series algorithms, an increasing number of researchers are inclined towards adopting one-stage detection methods to address object detection challenges. This trend has not only gained widespread recognition and discussion in academia but has also found practical application and promotion in industry.

Xie et al. [14] extracted the main features of YOLOv3 from the specialized transposed network Darknet19, reducing network complexity and achieving faster detection speeds. They replaced the YOLOv3 loss function with GIoU, enhancing insulator image recognition accuracy, albeit without corresponding evaluation of recall rates. Hu et al. [15] proposed an insulator defect detection algorithm based on Faster R-CNN and YOLOv3, leveraging the Faster R-CNN algorithm as the foundation. They introduced the EfficientNet-B3 module into the YOLOv3 backbone network and achieved high-precision insulator defect detection and recognition by incorporating the CBAM attention mechanism. However, this algorithm lacks evaluation of average precision (mAP) and detection speed (FPS).

Wang et al. [16] replaced the C3 module in the YOLOv5 backbone network with the improved C2f\_DG module and performed knowledge distillation on YOLOv5m. Their algorithm achieved a detection speed (FPS) of 63.6 frames per second with reduced parameters and computational complexity, albeit at the expense of lower accuracy and recall rates. Zhou et al. [17] designed a rotation mechanism on top of the YOLOv5 baseline to generate anchors with offset angles, aligning the framework with target edge information, and added attention mechanisms at detection points. Their algorithm achieved significant improvements in both average precision (mAP) and image processing speed (FPS), yet it lacked sufficient comparative experiments for convincing validation.

Zhao et al. [18] improved the YOLOv7 backbone network with the MobileNetv3 module and utilized image

augmentation techniques, which played a crucial role in detecting small objects under low-light conditions. Their algorithm exhibited better target detection accuracy and speed on the DIFD data set, yet it had higher hardware dependencies, requiring sensitive cameras and advanced imaging sensors for capturing usable images under low-light conditions. He et al. [19] combined YOLOv8s with the Swin Transformer and implemented an enhanced Bidirectional Feature Pyramid Network (BiFPN) structure in the neck network to enhance feature extraction capabilities, thereby improving the accuracy of insulator defect detection. However, the algorithm lacked detailed information regarding data set sources. He et al. [20] constructed various types of insulator fault scenarios and introduced the MSA–Ghost-Block feature extraction structure into the YOLOv8 algorithm, utilizing attention mechanisms built with GhostNet and asymmetric convolutions. Their algorithm achieved a 4.7% increase in average accuracy, albeit without an analysis of detection speed (FPS).

Wu et al. [21] developed a detection technique employing a Multi-Scale Feature Interaction Transformer Network (MFITN) for small insulator defect identification. This approach uses a super-resolution module to create high-resolution images that fulfill the requirements for object detection, thereby significantly improving detection capabilities for small targets. However, the algorithm is specialized for small insulator defects, which limits its general use. Zhang et al. [22] enhanced the YOLOv8 model by incorporating a Multi-Scale Large Kernel Attention (MLKA) module, which improves the model's focus on features of varying scales. They also developed the GSC\_C2f module, which features dense residual connections that facilitate gradient flow and make the network easier to train. However, this algorithm has a high computational cost and slower real-time monitoring speed.

The studies mentioned above have achieved notable results in detecting insulator defects, with corresponding optimizations in accuracy, precision, and speed. However, these optimizations have been overly focused in one direction, exhibiting significant limitations. With the continuous improvement of YOLO series algorithms, there remains substantial room for further optimization.

To address the issue of overly singular optimizations and enable detection algorithms to integrate better into detection devices, such as drones, facilitating more convenient and efficient grid inspection, this paper proposes an algorithm, CDDCR–YOLOv8, which simultaneously satisfies lightweight, high-speed, and high-accuracy requirements for insulator defect detection. The main contributions of this paper are as follows:

- (1) Introducing the Coordinate Attention (CA) mechanism module into the YOLOv8 backbone network and

replacing the original C2f module with the improved C2f\_DCN module to enhance the network's feature extraction capabilities. This enhancement suppresses interference features during detection, thereby improving target detection accuracy.

- (2) Improving the upsampling and downsampling network layers in the YOLOv8 neck network by introducing the DySample and CGNet\_D modules, capturing rich feature information at different levels to enhance detection performance and robustness. In addition, the RepGhost module is introduced to redesign the parameter structure of the Ghost module, making hardware implementation more efficient and improving model performance and efficiency.
- (3) Replacing the original CIoU loss function with the Wise-IoU (WIoU) loss function to further improve detection accuracy, accelerate network convergence speed, and enhance detection speed.
- (4) Conducting experimental validation of the improved algorithm using two data sets: the Chinese Power Line Insulator Data Set (CPLID) and a comprehensive data set (IPLID) composed of data collected from various sources, including Baidu, Google, and public data sets, combined with data captured by drones. Using multiple data sets for experimental validation helps avoid performance biases or overfitting issues associated with a single data set, providing a more comprehensive evaluation of the improved model's performance.

### 3 Materials and Methods

#### 3.1 Improved Algorithm CDDCR–YOLOv8

Despite the enhancements in detection accuracy and speed over YOLOv5 and YOLOv7, YOLOv8 still exhibits certain limitations. Notably, YOLOv8 may experience missed detections when handling small objects, attributed to its suboptimal performance under default settings. In addition, the increased algorithm size of YOLOv8 compared to YOLOv5 necessitates a higher consumption of computational resources and storage space when processing large-scale images. To address these challenges, this paper proposes an improved detection algorithm, CDDCR–YOLOv8, which builds upon the YOLOv8 framework. As illustrated in Fig. 1, this algorithm incorporates tailored enhancements to the backbone and neck networks of YOLOv8, augmenting geometric transformation learning capabilities and refining feature extraction accuracy, particularly in complex backgrounds. Furthermore, by replacing the original CIoU loss function with the Wise-IoU loss function, the algorithm enhances the alignment between predicted results and

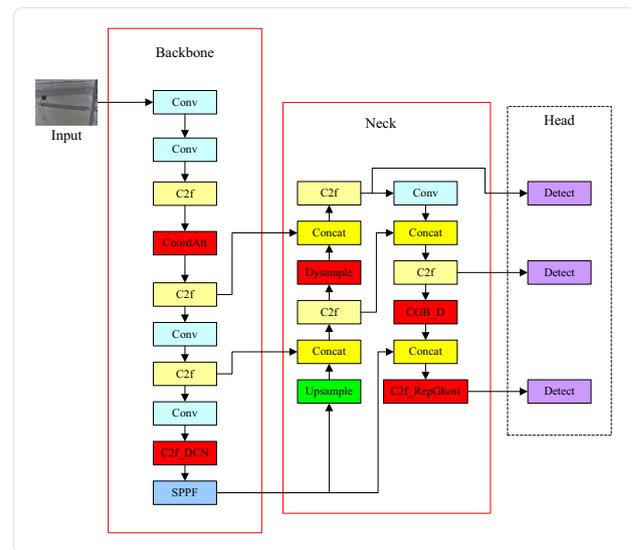


Fig. 1 CDDCR–YOLOv8 network structure

ground truth values, thereby accelerating the convergence speed of the network model.

#### 3.2 Coordinate Attention Module

Based on human visual attention mechanisms, the attention mechanism significantly improves the efficiency, speed, and accuracy of insulator defect detection. While the SE module [23] and the CBAM module [24] offer advantages, they have limitations. In contrast, the CA module [25] excels in capturing inter-channel information, direction, and position awareness. In addition, it is flexible, lightweight, requires less computational resources, and exhibits superior performance. The basic structure of this module is depicted in Fig. 2.

The CA attention mechanism consists primarily of two components: coordinate information embedding and coordinate attention. It encodes channel relationships and long-range dependencies through precise positional information. During the coordinate information embedding stage, the model integrates the input image feature map with positional information to obtain the location information of each pixel. Subsequently, these positional details are further processed to generate a two-dimensional positional attention map. The next stage involves coordinate attention generation. Here, the model combines the positional attention map generated in the previous stage with the input feature map. Consequently, the features of each channel are weighted by attention from different positions. Finally, the processed feature map is outputted as the model's final result, as shown in the following equation:

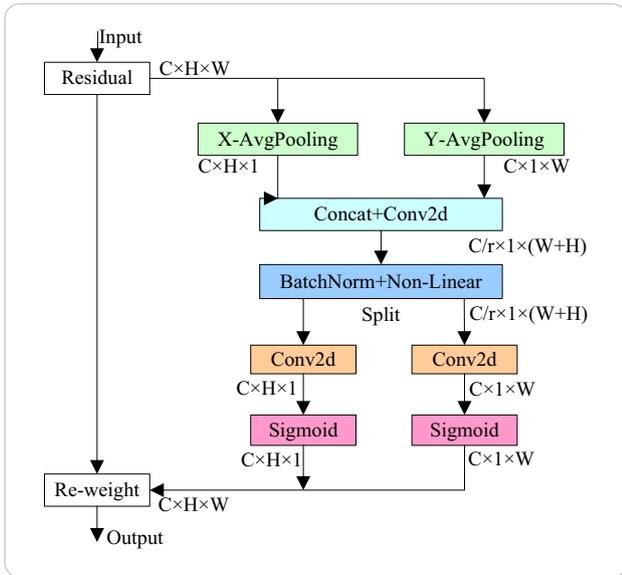


Fig. 2 Structure of CA attention mechanism

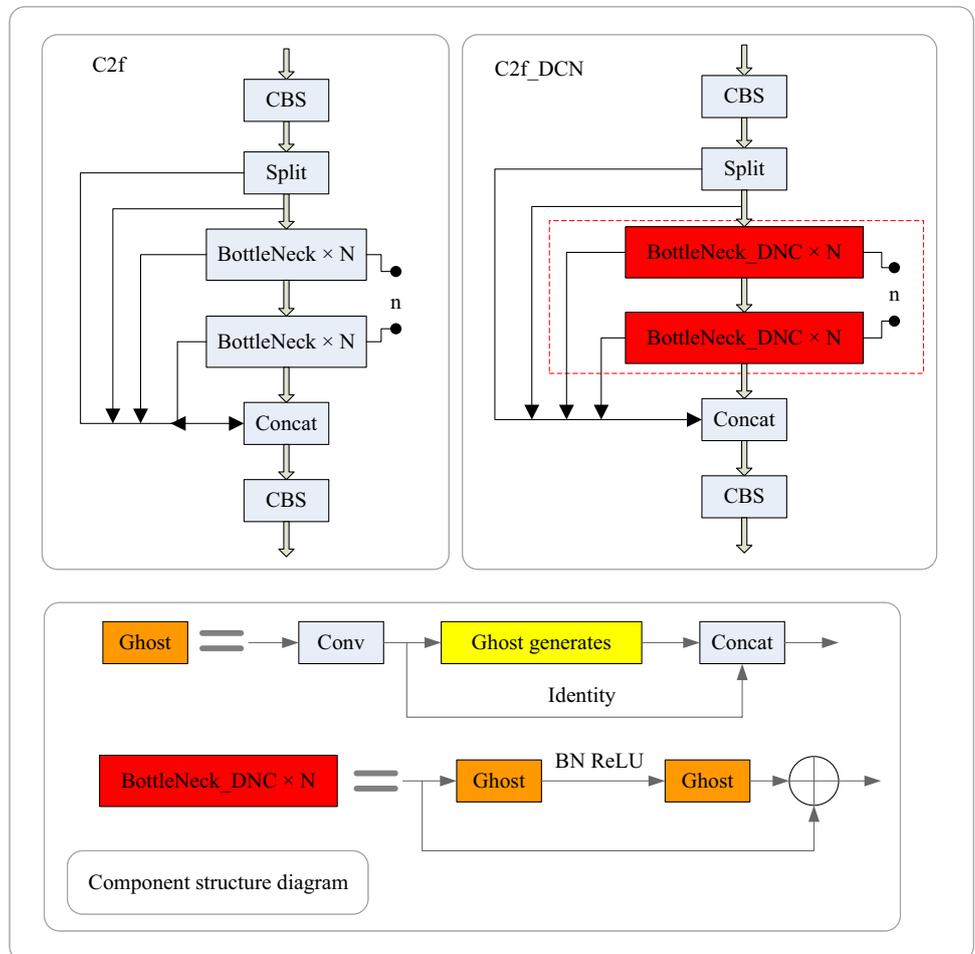
$$y_c(i, j) = x_c(i, j) \times g_h^c(i) \times g_w^c(j). \tag{1}$$

### 3.3 C2f\_DCN Module

The shapes, sizes, and positions of insulator defects vary often within a complex background alongside other objects. Traditional convolutional approaches struggle with adaptability, leading to difficulties in accurately pinpointing insulator defect locations. To overcome this limitation, we integrate an enhanced C2f\_DCN convolutional module [26] into the network architecture, replacing the conventional C2f module.

As depicted in Figs. 3 and 4, C2f\_DCN utilizes an operation known as deformable convolution, contrasting with traditional convolutional networks. This operation enables the addition of a displacement to the regular sampling coordinates, thereby generating new sampling points. Moreover, it introduces a weighting coefficient  $\Delta m$ . Specifically, it manipulates by introducing learnable offset vectors on a regular grid, where each point incorporates a learnable offset  $\Delta p$ . This process generates 2N feature maps corresponding

Fig. 3 Structure of C2f and C2f\_DCN



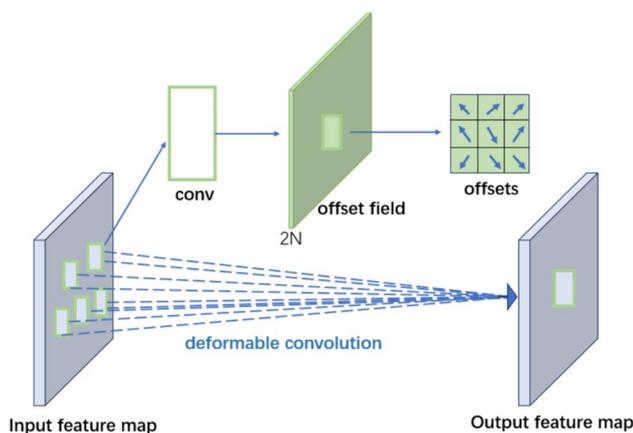


Fig. 4 Principle of C2f\_DNC

to  $N$  input channels. By augmenting offsets during training, the network's adaptability to targets is enhanced, consequently improving its performance and robustness.

In Fig. 3, C2f denotes the original module in YOLOv8, whereas C2f\_DCN replaces the bottleneck in the original C2f with Bottleneck\_DCN. The C2f\_DCN module incorporates cross-layer fusion and gradient clipping mechanisms to enable effective cross-layer feature extraction, reduce the model size, and improve training performance. The Bottleneck\_DCN structure consists of two Ghost modules. The first Ghost module increases the channels of the input feature map, providing expansion for subsequent operations. The second Ghost module reduces the channels of the output feature map to match the network structure, facilitating information transfer between the Ghost modules. The main distinction between the two Ghost modules is that the first is followed by a ReLU activation function, while the subsequent layers undergo batch normalization. This design reduces model parameters and computational complexity while optimizing feature maps via the Ghost modules, enhancing the model's detection efficiency.

The deformable convolutional output features are

$$y(p) = \sum_{k=1}^k w_k \cdot x(p + p_k + \Delta p_k) \cdot m_k. \tag{2}$$

### 3.4 DySample Module

YOLOv8 implements multi-scale detection capability via an upsampling module. Given the diversity in object sizes during detection, this module merges multi-scale feature maps, allowing the model to detect objects of various sizes. Consequently, it enhances detection effectiveness for both small and large objects. Nonetheless, the upsampling module in

YOLOv8 substantially elevates computational complexity and exhibits insensitivity towards detecting small objects.

The Dysample module [27], as depicted in Fig. 5, operates as follows: given an upsampling factor  $s$  and a feature map  $X$  of size  $C \times H \times W$ , a linear layer with input and output channel numbers of  $C$  and  $2s^2$  respectively, is employed to generate offsets  $O$  of size  $2s^2 \times H \times W$ . These offsets are then reshaped into  $2 \times sH \times sW$  using Pixel shuffling [28]. The sampling set  $S$  is obtained by adding the offsets  $O$  to the original sampling grid  $G$ , that is

$$O = \text{linear}(X), \tag{3}$$

$$S = G + O. \tag{4}$$

The input feature is represented by  $X$ , the upsampled feature by  $X'$ , the generated offsets by  $O$ , and the original grid by  $G$ . The sampling set is the sum of the generated offsets and the original grid positions. Figure A depicts the structure with a “static scope factor,” where offsets are generated using a linear layer. Figure B outlines the structure with a “dynamic scope factor,” where the range factor is first generated and then utilized to modulate the offsets. ‘ $\sigma$ ’ denotes the sigmoid function.

The Dysample module eliminates the need for additional dynamic convolutions and sub-networks, thus reducing parameter count, floating-point operation count (FLOPs), GPU memory, and latency. By learning sampling positions, it can reconstruct feature maps more accurately, mitigating common artifacts and blurring effects often encountered in traditional upsampling methods, thereby enhancing image clarity. Replacing the original upsampling module with the Dysample module has improved the algorithm's computational complexity and enhanced target detection accuracy.

### 3.5 Improved CGNet\_D Module

CGNet\_D [29] is an improved module based on CGNet [30], serving as a lightweight contextual guidance network primarily designed for semantic segmentation tasks. The structure of this module is illustrated in Fig. 6.

Comprising four sub-modules—the local feature extractor ( $f_{loc}^*$ ), the surrounding context feature extractor ( $f_{sur}^*$ ), the joint feature extractor ( $f_{joi}^*$ ), and the global feature extractor ( $f_{glo}^*$ )—the module begins by learning joint features from local elements and surrounding contexts. It then utilizes global context to perform channelwise weighting on the joint features and enhances information flow through residual learning.

Subsampling: The Conv  $1 \times 1$  layer initially halves the spatial dimensions of the input and adjusts the channel numbers accordingly.

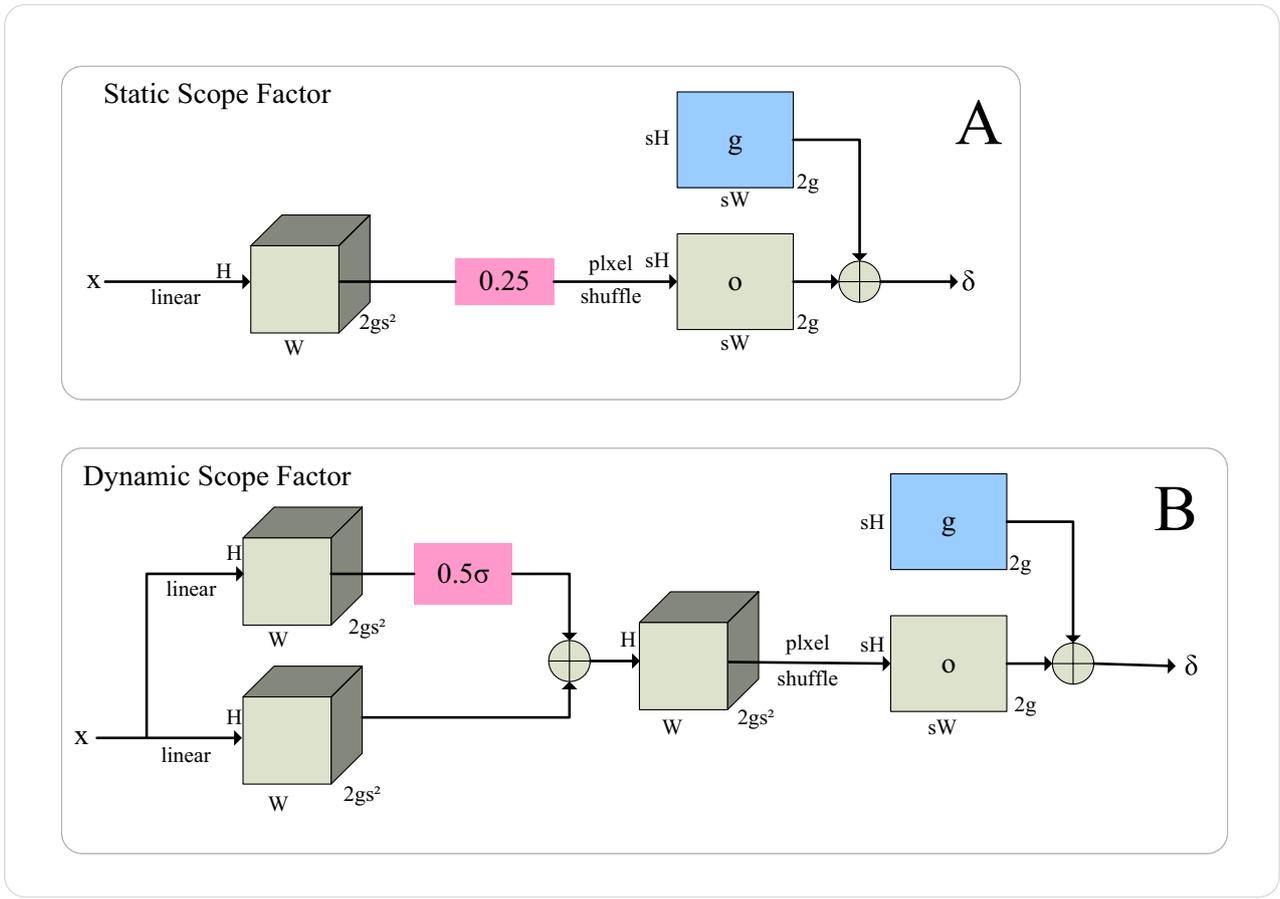


Fig. 5 DySample module

**Feature Integration:** This step encompasses the concatenation of local ( $f_{loc}$ ) and surrounding ( $f_{sur}$ ) features, followed by additional processing to effectively merge these elements. Subsequently, downsampling reduces the spatial dimensions of the input, with corresponding adjustments made to the Conv  $1 \times 1$  layer and channel numbers. Finally, the global feature layer ( $f_{glo}$ ) is employed to refine these integrated features:

$$f_{joi}^* = f_{joi}(f_{loc}^*, f_{sur}^*) = BN(PReLU([f_{loc}^*, f_{sur}^*])). \tag{5}$$

In the equation above,  $f_{loc}^*$ ,  $f_{sur}^*$ , and  $f_{joi}^*$  represent the local, surrounding, and joint features, respectively. The notation  $[f_{loc}^*, f_{sur}^*]$  signifies the concatenation operation between local and surrounding features. PReLU represents Parametric Rectified Linear Unit, and BN stands for Batch Normalization.

The global context feature is obtained through the joint feature, and the expression is as follows:

$$f_{glo}^* = f_{glo}(f_{joi}^*) = FC(FC(GAP(f_{joi}^*))), \tag{6}$$

where  $f_{joi}^*$  and  $f_{glo}^*$  represent joint feature and global context feature, respectively, GAP represents average pooling, and FC represents the fully connected layer.

By weighting the joint feature and the global context feature at the channel level, the output feature is obtained, and the expression is as follows:

$$f_{out}^* = f_{glo} \odot f_{joi}^*, \tag{7}$$

where  $f_{out}^*$  represents output features,  $f_{glo}^*$  and  $f_{joi}^*$  represent global context features and joint features,  $\odot$  represent element multiplication.

We apply it to the downsampling layer of YOLOv8, aiming to effectively utilize contextual information from images to enhance the algorithm’s understanding of semantic content, thereby improving the performance of semantic segmentation.

### 3.6 RepGhost Module

The RepGhost module [31] is a novel hardware component for deep neural networks, with its bottleneck structure

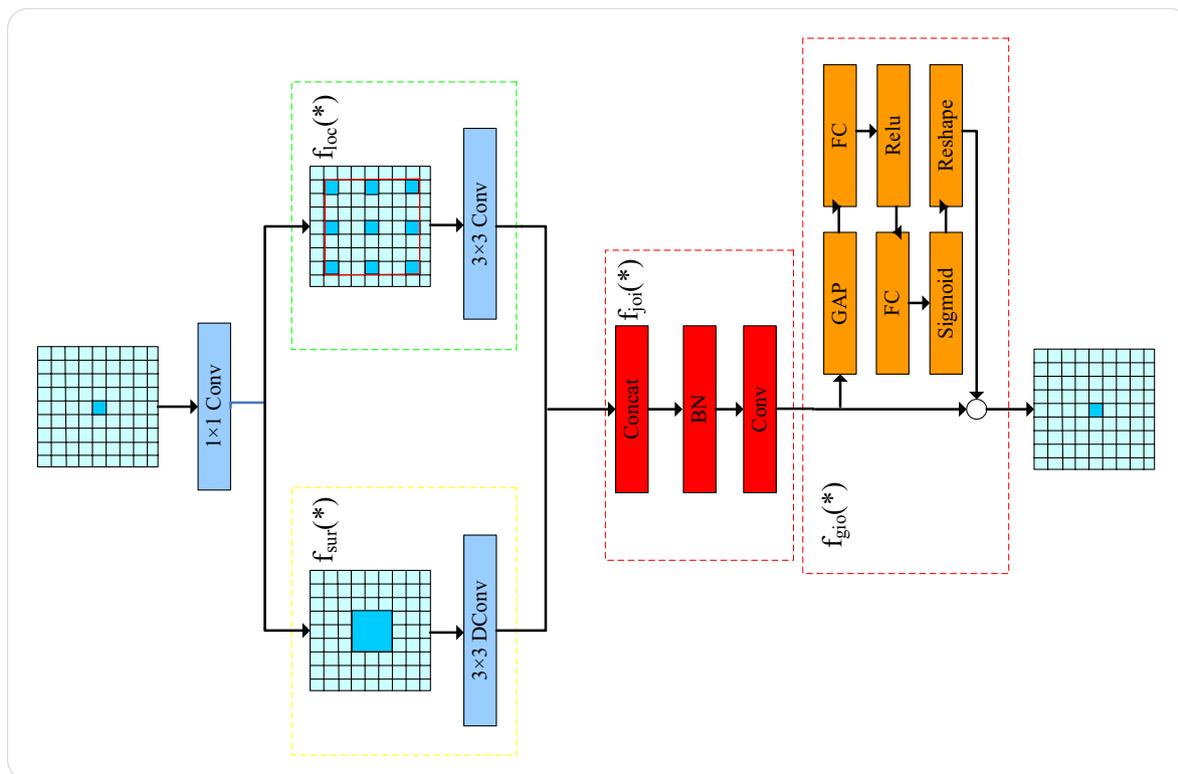


Fig. 6 Structure of CGNet\_D

depicted in Fig. 7. By incorporating convolutional layers, depthwise separable convolutions, the Squeeze-and-Excitation (SE) mechanism, expanded convolutional layers, batch normalization, and ReLU activation functions, these components are interconnected or summed to collectively achieve efficient feature extraction and reuse within the module while maintaining its performance.

### 3.7 WIoU Loss Function

In neural network training, the loss function plays a crucial role. First, it serves as the optimization objective. By minimizing the loss function, neural networks continuously adjust their parameters during training to predict unseen data more accurately. Second, the value of the loss function can also serve as an indicator for evaluating model performance.

The CIoU loss function used in YOLOv8 exhibits the following shortcomings:

- (1) When there is no intersection between the predicted box and the ground truth box, the loss function value becomes 0 during training, leading to hindered gradient backpropagation and ineffective model learning.
- (2) When the predicted box and the ground truth box have the same intersection over union (IoU) but are located

differently, the loss calculated remains the same, making it challenging to accurately determine which prediction is more precise.

This paper proposes replacing the CIoU loss function with the WIoU loss function [32] to further enhance the model's performance. WIoU (Weighted IoU) is a novel IoU loss function that introduces focusing coefficients to weight the IoU loss. This weighting strategy better reflects the overlap between predicted boxes and ground truth boxes, particularly when the predicted box encompasses the ground truth box, reducing the loss value.

The definition of Wise-IoU v1 is given by the following equation:

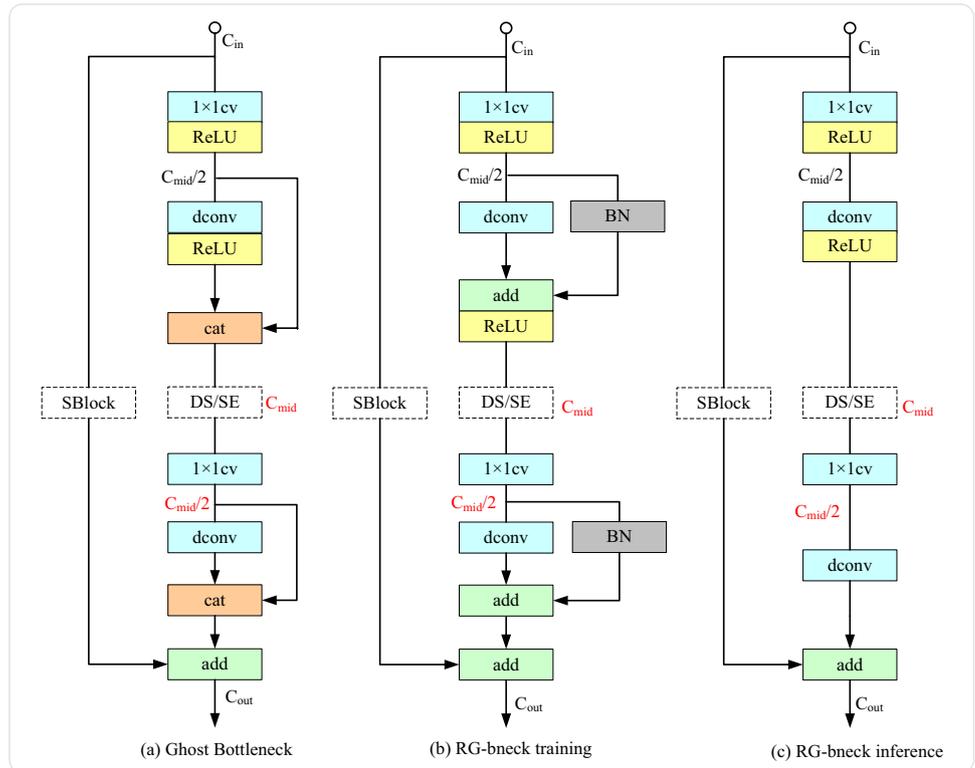
$$L_{WIoUv1} = R_{WIoU} L_{IoU}, \tag{8}$$

$$R_{WIoU} = \exp\left(\frac{(x - x_{gt})^2 + (y - y_{gt})^2}{(W_2^g + H_2^g)^*}\right). \tag{9}$$

Among  $R_{WIoU} \in [1, e)$ ,  $R_{WIoU} \in [1, e)$ ,  $L_{IoU} \in [0, 1]$ .

Compared to v1, Wise-IoU v3 introduces a gradient amplification allocation strategy with dynamic non-monotonic FM. This strategy enables the model to dynamically adjust the strength of gradient amplification based on the

**Fig. 7** RepGhost bottleneck structure



quality of samples, thereby enhancing training efficiency and performance.

The definition of Wise-IoU v3 is given by the following equation:

$$L_{WIoUv3} = r L_{WIoUv1} \tag{10}$$

$$\text{Among } \beta = \frac{L_s^{IoU}}{L_{IoU}} \in [0, +\infty) \beta = \frac{L_s^{IoU}}{L_{IoU}} \in [0, +\infty), r = \frac{\beta}{\delta \alpha^{\beta-\delta}}$$

Utilizing WIOU v3 as the loss function can enhance the accuracy and robustness of the model without significantly increasing computational burden.

## 4 Experiment Preparation

### 4.1 Experimental Environment

During network training, we resize the input image size to 640×640 pixels, set the training epochs to 300, batch size to 16, initial learning rate to 0.01, and maximum number of working threads to 4. Throughout the experiments, we maintain a consistent experimental environment. The configuration of the experimental environment is presented in Table 1.

**Table 1** Experimental environment configuration

Parameter	Configuration
Operating system	Linux Ubuntu 18.04.6 LTS
CPU	Intel(R) Core(TM) i9-10900X CPU @ 3.70 GHz
GPU	NVIDIA GeForce RTX 3080
Video memory	10 GB
Programming language	Python 3.8.18
Frame	Pytorch1.12.1+CUDA11.3+cudnn8.2.0
IED	PyCharm

### 4.2 Data Set

To validate the superiority of our proposed algorithm and mitigate potential performance bias or overfitting issues stemming from a single data set, we employed two data sets: the China Power Line Insulator Data Set (CPLID) [33] and the Integrated Power Line Data Set (IPLID) [34], which is compiled from various sources including Baidu, Google, and public data sets.

The original CPLID data set comprises 600 images of normal insulators and 248 images portraying insulator defects. Given the data set’s limited size, we employed measures to augment the data, thus enhancing the efficacy of network algorithms in training and mitigating the risk

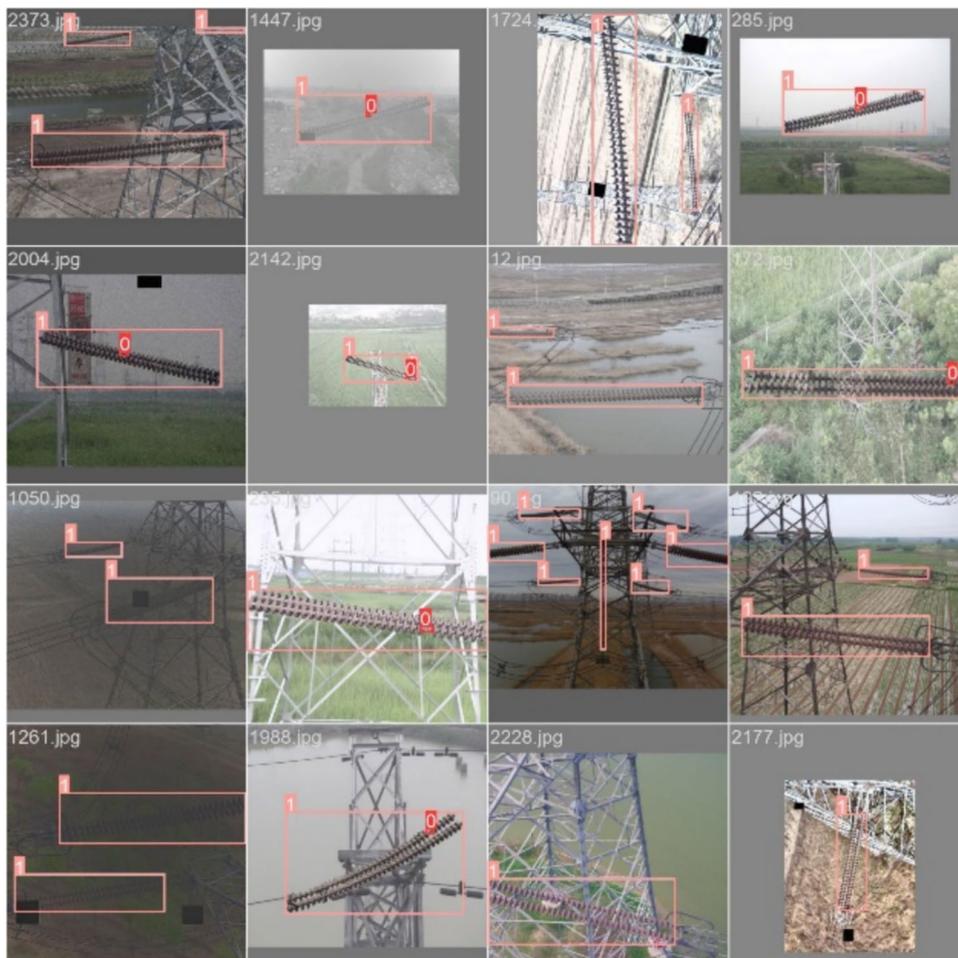
of overfitting due to insufficient data. Augmentation techniques, including histogram equalization, gamma correction, spatial domain filtering, and frequency domain filtering, were utilized, resulting in the expansion of the data set to over 3500 images, as illustrated in Fig. 8. Subsequently, 3000 images were selected from this augmented data set for use as the experimental data set in this study. These images were then divided into training and testing sets in an 8:2 ratio to facilitate comprehensive evaluation. The IPLID data set, released by Changzhou University and the Chinese Academy of Sciences, has been augmented to include 1600 images, as depicted in Fig. 9. It encompasses three categories: insulators, pollution flashovers, and fractures. For this study, we randomly selected 1440 images from this data set to serve as the experimental data set, dividing them into training and testing sets at a 9:1 ratio. Employing distinct partitioning ratios for the two data sets allowed for a thorough evaluation of algorithm performance and the validation of algorithmic generalizability.

The data sets were annotated using the CVAT tool. In the CPLID data set, annotations were divided into two categories: “damaged” and “insulator,” while the IPLID data set



Fig. 9 IPLID data set data enhancement

Fig. 8 CPLID data set data enhancement



annotations encompassed “pollution-flashover,” “broken,” and “insulator” categories. Details regarding the allocation of training and testing sets are outlined in Table 2.

### 4.3 Evaluation Index

This paper evaluates the model detection performance using metrics, such as Recall, Precision, Average Precision (AP), Intersection over Union (IoU), and mean Average Precision at 0.5 Intersection over Union threshold (mAP0.5).

Recall measures the proportion of true positive instances correctly identified among all actual positive instances, reflecting the model's capability to detect all true positives. Precision, on the other hand, quantifies the accuracy of the model's predictions by assessing the proportion of true positive instances among all instances classified as positive. Average Precision provides a comprehensive assessment by considering both recall and precision, calculated as the average of recall and precision across all classes, offering a more holistic reflection of the model's performance. Intersection over Union is a metric that gauges the overlap between predicted and ground truth bounding boxes, with values closer to 1 indicating higher overlap and better localization performance of the model.

The formulas for the calculations mentioned above are as follows:

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (11)$$

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (12)$$

$$AP = \int_1^0 P(R)dR, \quad (13)$$

$$IoU = \frac{\text{DetectionResult} \cap \text{GroundTruth}}{\text{DetectionResult} \cup \text{GroundTruth}}. \quad (14)$$

The term “TP” represents the number of accurately detected targets, “FP” indicates the number of falsely detected samples, and “FN” denotes the number of missed detections.

**Table 2** Training set and test set partitioning of CPLID and IPLID data sets

Data	CPLID data set	IPLID data set
Training set	2400	1296
Test set	600	144

Mean Average Precision (mAP0.5) is a crucial evaluation metric in object detection tasks, primarily used to measure the performance of algorithms. Specifically, mAP0.5 calculates the average precision value across all classes for all images under the condition of an Intersection over Union (IoU) threshold of 0.5.

The calculation formulas are as follows:

$$mAP = \frac{\sum_{i=1}^N AP_i}{N}. \quad (15)$$

Here,  $N$  represents the number of categories when labeling the data set.

## 5 Experimental Results and Analysis

### 5.1 Confusion Matrix Analysis

The confusion matrix delineates the correspondence between the algorithm's predictive outcomes and the true labels across various classes within the data set. Analyzing the confusion matrix allows for an insight into the algorithm's identification accuracy for each class, thereby facilitating the evaluation of its overall and specific category performance.

In Fig. 10, the confusion matrix results of YOLOv8 on the CPLID data set are presented in sub-figure (a). An examination of sub-figure a indicates a 96% accuracy in detecting insulator defects and a 91% accuracy in identifying regular insulator strings. Sub-figure (b) illustrates the confusion matrix outcomes of CDDCR–YOLOv8, revealing a 99% accuracy in detecting insulator defects and a 92% accuracy in identifying typical insulator strings. Consequently, the enhanced CDDCR–YOLOv8 algorithm demonstrates a 3% improvement in detecting insulator defects and a 1% enhancement in identifying standard insulator strings.

Figure 11 presents the confusion matrix outcomes of YOLOv8 on the IPLID data set in sub-figure (a). Analysis of sub-figure a reveals a 72% accuracy in predicting flashover insulators, an 81% accuracy in identifying damaged insulators, and a 100% accuracy in recognizing normal insulator strings. Sub-figure (b) depicts the confusion matrix results of CDDCR–YOLOv8, indicating a 77% accuracy in predicting flashover insulators, a 90% accuracy in identifying damaged insulators, and a 100% accuracy in recognizing normal insulator strings. Hence, the refined CDDCR–YOLOv8 algorithm exhibits a 5% enhancement in predicting flashover insulators and a 9% improvement in identifying damaged insulators.

By comparing the pre- and post-improvement confusion matrices from both the CPLID and IPLID data sets, the

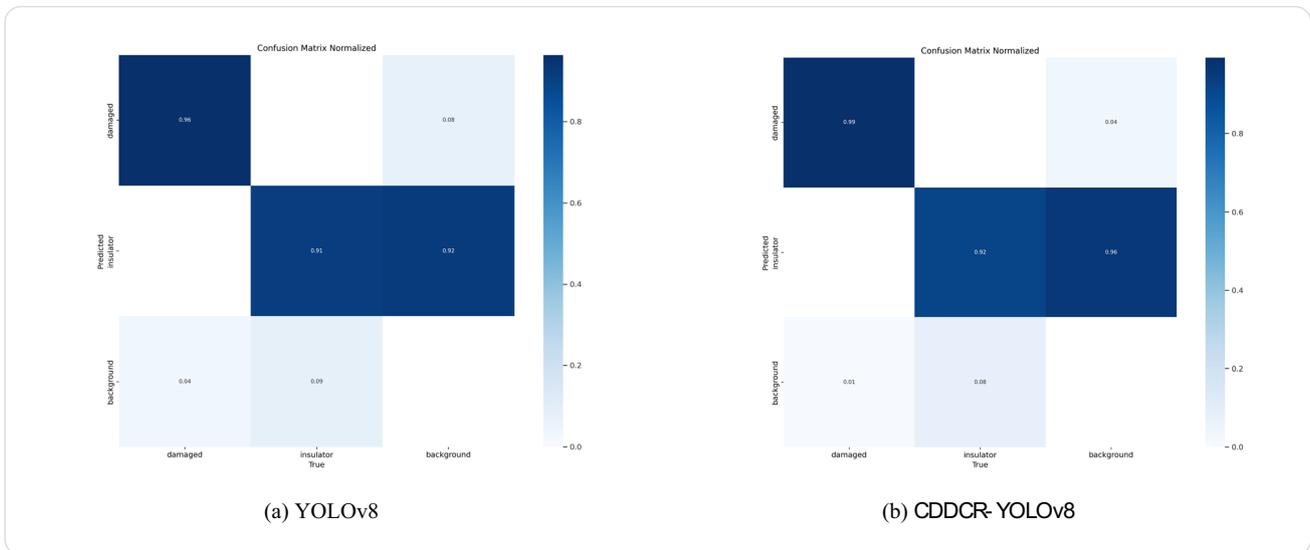


Fig. 10 Confusion matrix of original algorithm and improved algorithm in CPLID data set

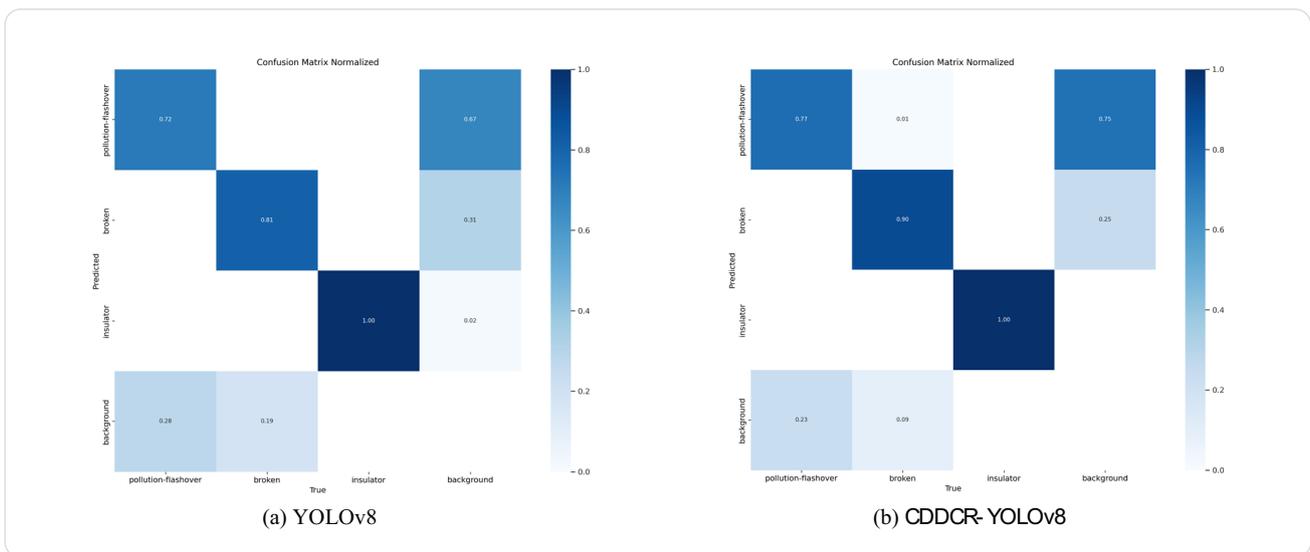


Fig. 11 Confusion matrix of original algorithm and improved algorithm in IPLID data set

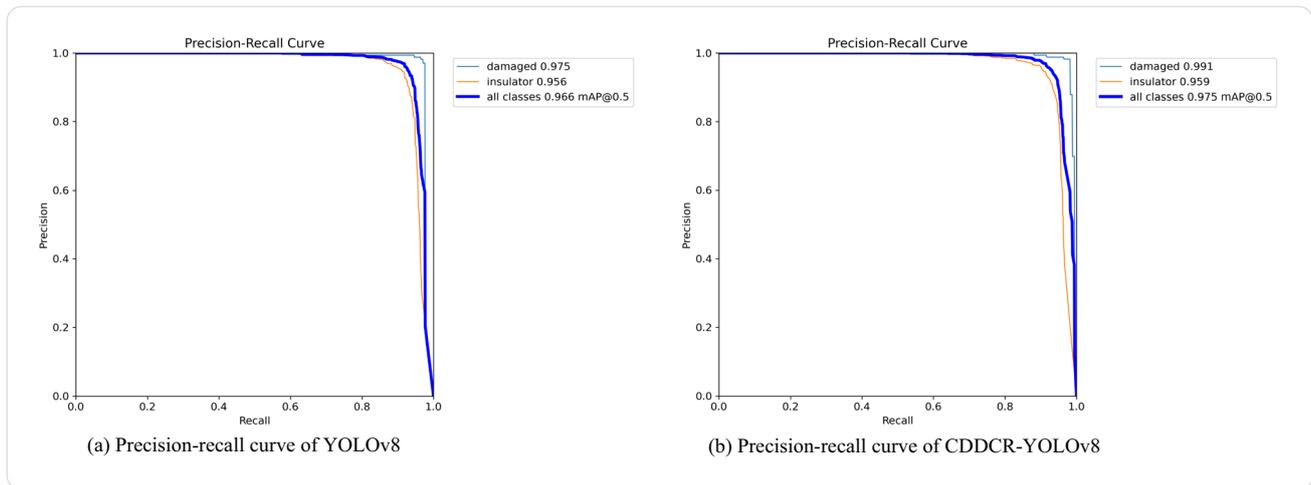
superiority of the enhanced CDDCR–YOLOv8 algorithm's performance has been validated.

### 5.2 PR Curve Analysis

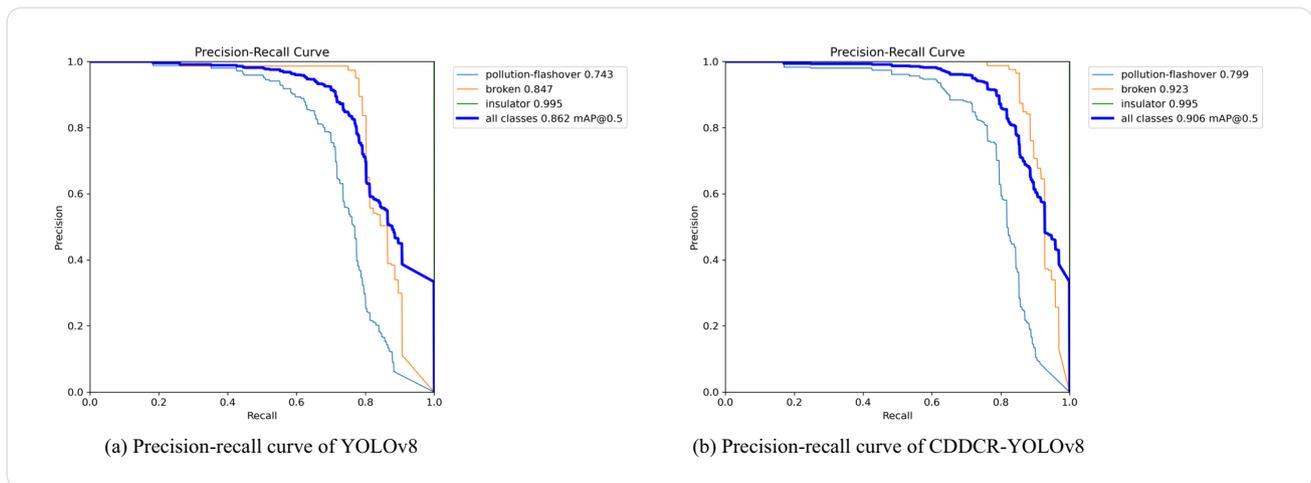
The Precision–Recall (PR) curve depicts the association between precision and recall. In this representation, ‘P’ represents precision, and ‘R’ signifies recall. Precision is mapped on the vertical axis, whereas recall is on the horizontal axis. Each point along the curve denotes the precision and recall values of the algorithm at varying thresholds. In

general, the closer the PR curve approaches the upper-right corner, the higher the algorithm’s performance.

Figure 12 displays the PR curves of YOLOv8 and CDDCR–YOLOv8 on the CPLID data set in sub-figures (a) and (b), respectively. The enhanced algorithm achieves a 1.6% improvement in the mean Average Precision (mAP@0.5) for predicting insulator defects over YOLOv8. Moreover, it records a 0.3% increase in mAP@0.5 for predicting normal insulator strings, culminating in an overall improvement of 0.9%.



**Fig. 12** PR curves in CPLID data set before and after algorithm improvement



**Fig. 13** PR curves in IPLID data set before and after algorithm improvement

As illustrated in Fig. 13, sub-figure (a) displays the PR curve of YOLOv8 on the IPLID data set, while sub-figure (b) exhibits that of CDDCR-YOLOv8. The improved algorithm demonstrates a 5.6% increase in the mean Average Precision (mAP@0.5) for predicting flashover insulators compared to YOLOv8. In addition, it showcases a 7.6% enhancement in mAP@0.5 for predicting damaged insulators, resulting in an overall improvement of 4.4%.

In both the CPLID and IPLID data sets, the improved algorithm demonstrates a substantial increase in the average precision at an IoU threshold of 0.5 (mAP@0.5). Moreover, its PR curves are significantly closer to the upper-right corner in comparison to YOLOv8, thereby reinforcing its superior performance.

### 5.3 Ablation Experiment

To assess the impact of the proposed enhancement modules on the performance of the detection algorithm, we conducted ablation experiments using YOLOv8n as the baseline model on both the CPLID and IPLID data sets. The experiments involved the inclusion of CA, C2f\_DCN, DySample, CGNet\_D, RepGhost, and WIoU modules, where “×” indicates the absence of the module and “√” denotes its inclusion.

The ablation experiment results on the CPLID data set, as depicted in Table 3, reveal that incorporating the CA module results in a minor decline in both precision and recall. However, there is an enhancement in the mean Average Precision (mAP@0.5). Subsequently, the stepwise addition

**Table 3** Results of ablation experiments in the CPLIP data set

Model	CA	C2f_DCN	DySample	CGNet_D	RepGhost	WIoU	Precision/%	Recall/%	mAP@0.5/%
YOLOV8n	×	×	×	×	×	×	96.7	93.9	96.6
	√	×	×	×	×	×	96.2	92.1	96.7
	√	√	×	×	×	×	96.5	92.3	96.7
	√	√	√	×	×	×	96.7	92.4	96.8
	√	√	√	√	×	×	96.8	92.7	96.9
	√	√	√	√	√	×	96.8	93.4	96.7
	√	√	√	√	√	√	96.9	94.3	97.5
	√	√	√	√	√	√	96.9	94.3	97.5

of various modules leads to a steady increase in precision and recall. Upon the inclusion of the WIoU loss function, the final fusion of modules results in improvements across precision, recall, and mAP@0.5.

The results of the ablation experiments on the IPLID data set are presented in Table 4. The experimental findings indicate that upon integrating the CA module, there is a decrease in precision by 4.9%. However, there are respective increases in recall and the mean Average Precision (mAP@0.5) by 0.6% and 2.7%. Subsequently, with the gradual inclusion of the C2f\_DCN and DySample modules, there are slight improvements in precision by 0.1% and 0.2% compared to the previous modules. Upon incorporating the enhanced CGNet\_D module, precision significantly improves, accompanied by steady increases in recall and mAP@0.5. Finally, with the addition of the WIoU loss function, precision reaches 93.4%, equivalent to that of the original YOLOv8 algorithm. However, the enhanced algorithm demonstrates notable improvements in recall and mAP@0.5 compared to the original YOLOv8 algorithm, increasing by 5.2% and 4.4%, respectively.

Through ablation experiments conducted on both the CPLID and IPLID data sets, we have validated the effectiveness and efficiency of the proposed enhancement modules.

#### 5.4 Loss Function Analysis

The loss function not only enhances the accuracy of the detection algorithm but also accelerates the convergence

speed of the network, thereby improving training efficiency, enhancing real-time detection capabilities, and bolstering the algorithm's robustness.

The curves in Fig. 14 compare the WIoU (Wise-IoU) loss function with the original loss function on both the CPLID and IPLID data sets in sub-figures (a) and (b), respectively. It is evident from the plots that the WIoU loss function achieves a significantly faster convergence speed, resulting in a substantial improvement in the performance of the enhanced algorithm.

#### 5.5 Comparative Experiments of Different Algorithms

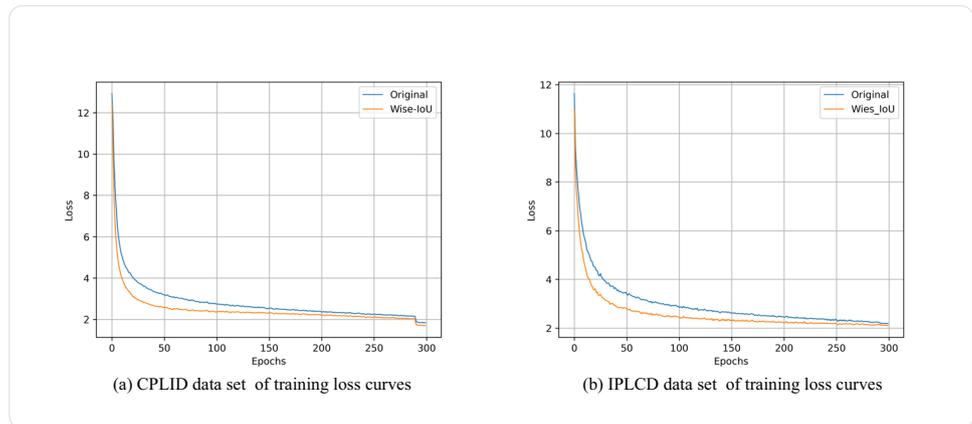
To further validate our improved algorithm's performance, we selected popular and widely used algorithms: YOLOv3-tiny, YOLOv5, YOLOv6, YOLOv7-Tiny, and YOLOv8 for comparative experiments. We conducted these experiments with corresponding literature [14, 16, 19, 35, 36], and [37–39] using the same original data sets. These algorithms and experiments represent significant advancements in complex environment detection, providing robust references for our research. The experimental results appear in Tables 5 and 6.

Based on the experimental results, on the CPLID data set, our improved algorithm shows significant improvements in accuracy compared to other mainstream algorithms, such as YOLO3-tiny, YOLOv5, YOLOv6, YOLOv7-Tiny, and YOLOv8, with increases of 13.8%, 1.8%, 3.5%, 2.2%, and

**Table 4** Results of ablation experiments in the IPLIP data set

Model	CA	C2f_DCN	DySample	CGNet_D	RepGhost	WIoU	Precision/%	Recall/%	mAP@0.5/%
YOLOV8n	×	×	×	×	×	×	93.4	79.6	86.2
	√	×	×	×	×	×	88.5	80.2	88.9
	√	√	×	×	×	×	88.6	80.3	89.1
	√	√	√	×	×	×	88.7	81.6	89.3
	√	√	√	√	×	×	90.1	81.9	89.2
	√	√	√	√	√	×	91.7	82.9	89.8
	√	√	√	√	√	√	93.4	84.8	90.6
	√	√	√	√	√	√	93.4	84.8	90.6

**Fig. 14** Loss function contrast curves in CPLID and IPLID data set



**Table 5** Comparison of experimental results of different algorithms in CPLID data set

Model	P/%	R/%	mAP@0.5/%	GFLOPs	FPS
YOLOv3-tiny	83.1	92.9	93.5	12.9	96.2
Literature [14]	88.43	–	88.43	–	–
YOLOv5	95.1	93.3	96.8	7.1	74.6
ML-YOLOv5 [16]	96.5	94.7	97.0	9.0	63.6
Literature [35]	94.8	91.9	93.0	15.65	–
YOLOv6	93.4	92.7	95.8	11.8	78.7
YOLOv7-tiny	94.7	93.5	96.9	13.0	59.9
YOLOv8	96.7	93.9	96.6	8.2	80.6
YOLOv8s-SwinT [19]	–	–	97.7	29.3	88.0
Literature [38]	99.2	95.7	–	–	–
Literature [39]	90.5	–	88.15	–	–
Ours	96.9	94.3	97.5	8.0	84.0

**Table 6** Comparison of experimental results of different algorithms in IPLID data set

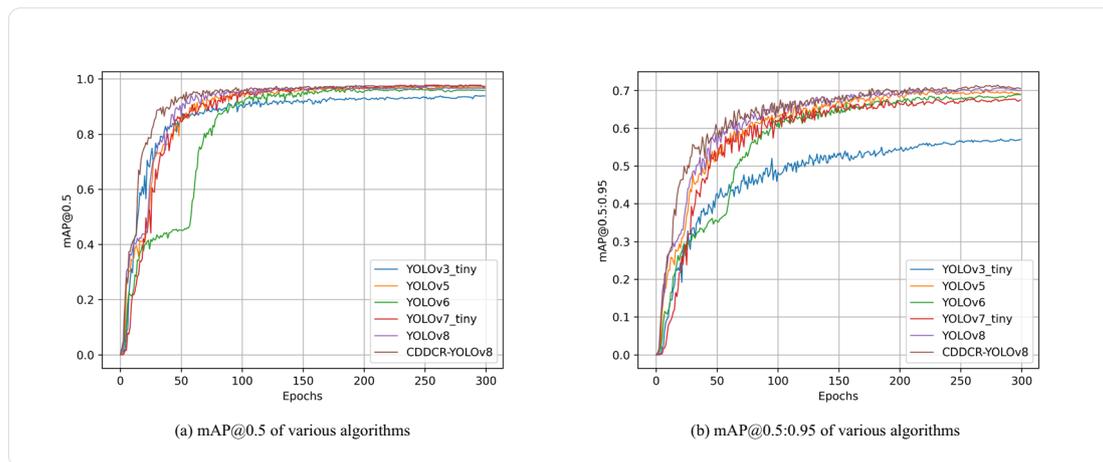
Model	P/%	R/%	mAP@0.5/%	GFLOPs	FPS
YOLOv3-tiny	87.5	76.8	80.3	12.9	96.2
YOLOv5	87.4	81.6	86.1	7.1	74.6
ML-YOLOv5 [16]	86.1	77.7	82.8	9.0	63.6
YOLOv6	89.2	79.5	83.3	11.8	78.7
YOLOv7-tiny	87.2	78.5	83.8	13.0	59.9
Literature [36]	–	–	88.7	8.6	–
YOLOv8	93.4	79.6	86.2	8.2	80.6
Literature [37]	93.2	84.3	90.0	–	–
Ours	93.4	84.8	90.6	8.0	84.0

0.2%, respectively. Recall and mAP@0.5 also exhibit corresponding enhancements. When compared to the improvement algorithms in references [14, 16, 35], our algorithm demonstrates accuracy improvements of 8.47%, 0.4%, and 2.1%, respectively, with corresponding mAP@0.5

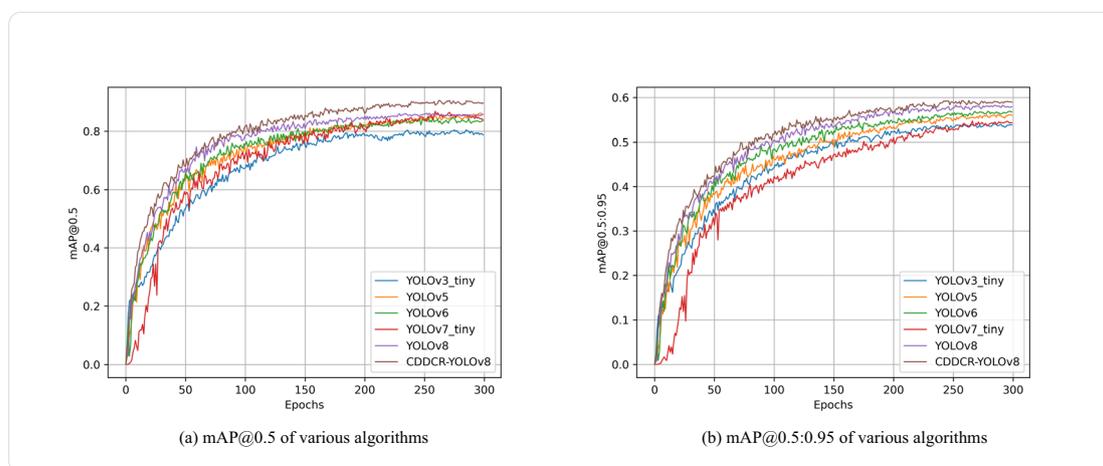
enhancements of 9.07%, 0.5%, and 4.5%. The GFLOPs reduced significantly from 29.3 to 8.0 compared to the improvement algorithm in reference [19], resulting in a considerable decrease in computational complexity. On the IPLID data set, our improved algorithm achieves comparable accuracy to the YOLOv8 algorithm but surpasses it in recall and mAP@0.5 by 5.2% and 4.4%, respectively. Compared to other mainstream algorithms, there are substantial improvements in accuracy, recall, and mAP@0.5. Furthermore, compared to the improvement algorithm in reference [36], our algorithm achieves a 1.9% increase in mAP@0.5. Relative to the improvement algorithm in reference [16], our algorithm demonstrates improvements of 7.3%, 7.1%, and 7.8% in accuracy, recall, and mAP@0.5, respectively, while slight enhancements are observed compared to the improvement algorithm in reference [37].

In the CPLID data set, we compared our results with the most recent studies [38, 39]. According to Table 5, the algorithm in [38] has slightly higher precision and recall than our improved algorithm. However, it does not test for average precision, computational complexity, and FPS, which limits its comprehensiveness and does not sufficiently demonstrate its superiority. Our improved algorithm, CDDCR–YOLOv8, exhibits improvements in several evaluation metrics, proving its advantages. Compared to the latest findings in [39], our improved algorithm achieves an increase of 6.4% in precision and 9.35% in average precision. This substantial improvement demonstrates the superior performance of our algorithm in detecting insulator defects.

To provide a clearer insight into the superior mAP performance of our improved algorithm, we conducted visual comparisons with other algorithms on the CPLID and IPLID data sets. The comparative outcomes are illustrated in Figs. 15 and 16: with Fig. 15a presenting the comparison curves of different algorithms on mAP@0.5, and Fig. 15b depicting the comparison curves of different algorithms on mAP@0.5:0.95. From the plotted curves, it is evident that



**Fig. 15** Comparison curves between  $mAP@0.5$  and  $mAP@0.5:0.95$  of different algorithms in CPLID data set



**Fig. 16** Comparison curves between  $mAP@0.5$  and  $mAP@0.5:0.95$  of different algorithms in IPLID data set

our improved algorithm outperforms other algorithms on both data sets.

Comparison experiments on the CPLID and IPLID data sets demonstrate that our improved algorithm, CDDCR–YOLOv8, outperforms the original YOLOv8 algorithm and other comparison algorithms across various metrics. Our algorithm exhibits significant enhancements in accuracy, recall, and average precision while also reducing computational complexity. The complexity has been lowered to 8.0, effectively conserving memory and energy consumption, thus enhancing overall detection efficiency. Furthermore, our improved algorithm achieves 84 FPS, meeting the requirements for embedding in detection devices for real-time detection, which substantially impacts industrial applications.

## 5.6 Visualization of Detection Results

To evaluate the effectiveness of our improved algorithm in detecting insulator defects, we randomly selected images of insulator defects from two data sets for detection, as shown in Figs. 17 and 18.

In the CPLID data set, (a1), (b1), and (c1) represent YOLOv8 detections in normal, hazy, and dark environments, while (a2), (b2), and (c2) depict the enhanced algorithm CDDCR–YOLOv8 detections in those environments. The results show an 11% higher confidence in insulator defect detection by the improved algorithm compared to YOLOv8 in normal conditions, a 4% increase in dark environments, and a remarkable 43% improvement in hazy conditions. This confirms the algorithm’s superiority in detecting insulator defects in complex scenarios, highlighting its exceptional interference resistance.

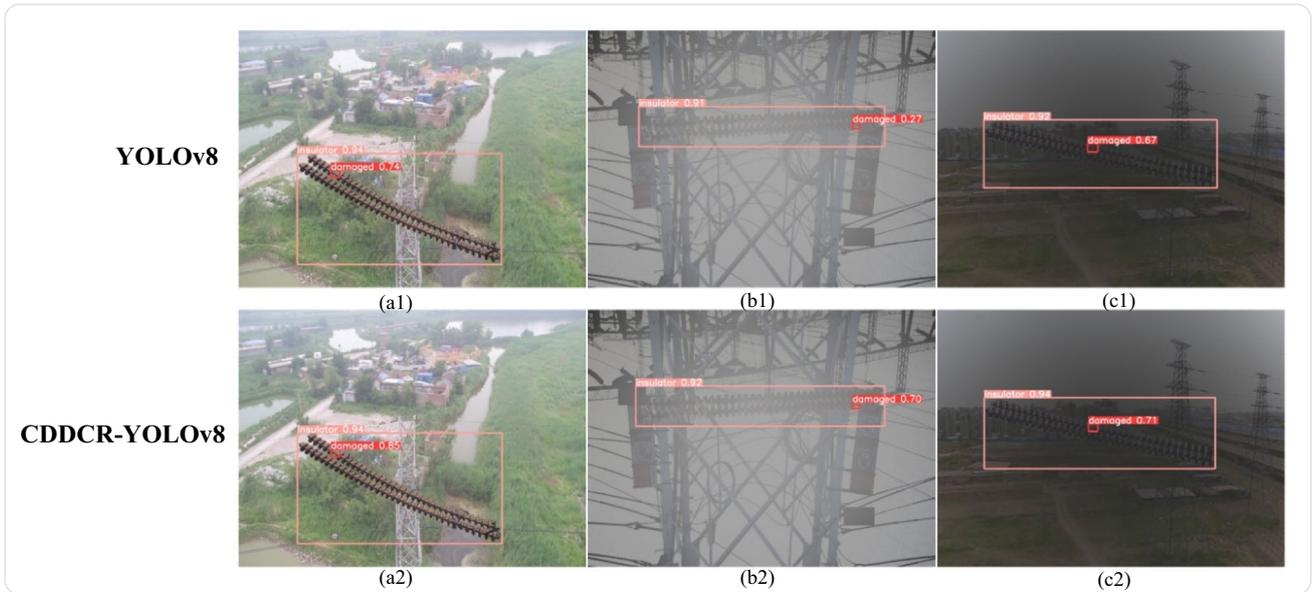


Fig. 17 Results of insulator defects were detected in CPLID data set before and after the algorithm improvement

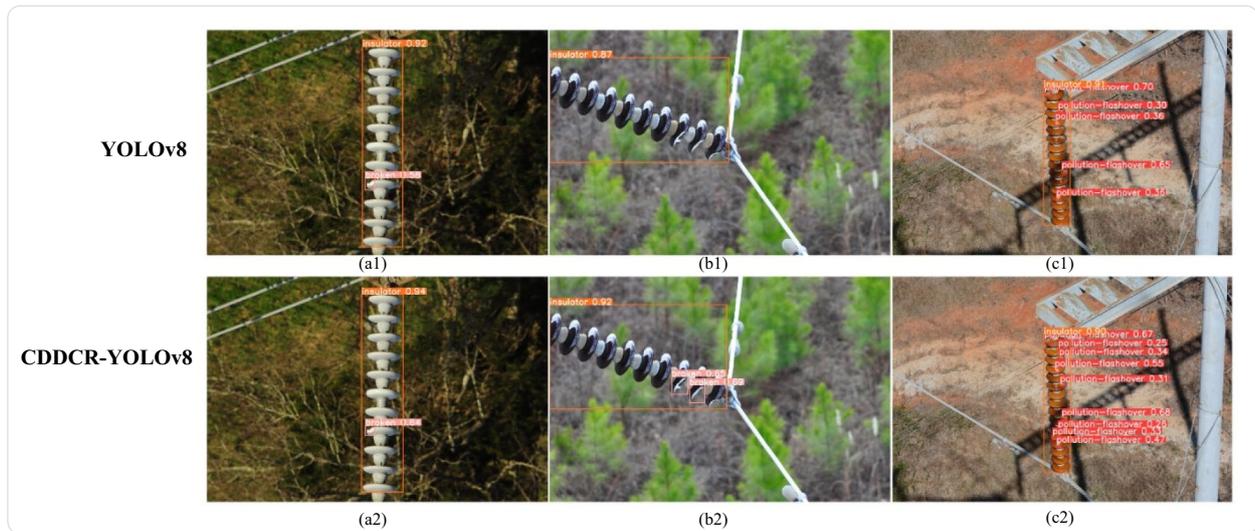


Fig. 18 Results of insulator defects were detected in IPLID data set before and after the algorithm improvement

In the IPLID data set, (a1), (b1), and (c1) represent the detection results of YOLOv8, whereas (a2), (b2), and (c2) denote the detection outcomes of the enhanced algorithm CDDCR–YOLOv8. Upon comparing (a1) with (a2), a noticeable enhancement of 6% in confidence is discerned in the improved algorithm’s detection of damaged insulators compared to YOLOv8. Contrasting (b1) with (b2), it

is evident that YOLOv8 exhibits conspicuous instances of missed detections for damaged insulators, a shortcoming effectively addressed by the improved algorithm. Finally, juxtaposing (c1) with (c2) reveals that YOLOv8 displays a higher frequency of missed detections for flashover insulators, whereas the enhanced algorithm almost entirely captures flashover insulators with heightened confidence

levels surpassing those of YOLOv8. This underscores the versatility of the improved algorithm and its efficacy in detecting various insulator defects.

## 6 Conclusion

Through enhancements in the convolutional modules within the backbone network of YOLOv8, coupled with the integration of the CA attention mechanism, and optimizations in the structure of the upsampling and downsampling layers within the neck network, alongside the introduction of RepGhost lightweight modules to replace the neck convolutions, we propose an improved algorithm, namely, CDDRC–YOLOv8. In addition, we employ the WIoU loss function to enhance detection speed and convergence rate. Experimental results demonstrate that compared to traditional algorithms, our proposed improvement algorithm can more effectively detect insulator defects in various complex environments. Moreover, it exhibits diversity and achieves higher detection accuracy for different types of insulator defects, displaying robustness and generalization capabilities. We validate the performance of the improved algorithm using two data sets, thus avoiding issues of sample imbalance and incompleteness caused by biases during sampling or collection processes, thereby better assessing the generalization ability and performance of the improved algorithm. Furthermore, with a GFLOPs of only 8 and an FPS reaching 84, the algorithm meets real-time monitoring requirements effectively, laying a solid groundwork for embedding the algorithm into detection devices, such as drones.

**Author Contributions** Conceptualization: T.J., M.W.; methodology: X.H.; formal analysis and investigation: X.H.; writing—original draft preparation: X.H.; writing—review and editing: T.J., M.W.; funding acquisition: T.J.; supervision: M.W. All authors reviewed the manuscript.

**Funding** This work is supported by the Natural Science Foundation of Hubei Province, grant number 2021CFB004.

**Data Availability** All data used to support the findings of this study are included within the article. No data sets were generated or analysed during the current study.

## Declarations

**Conflict of Interest** The authors declare no conflicts of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are

included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Sadykova, D., Pernebayeva, D., Bagheri, M., et al.: IN-YOLO: real-time detection of outdoor high voltage insulators using UAV imaging. *IEEE Trans. Power Delivery* **35**(3), 1599–1601 (2020)
2. Shihab, U.A., Apsaan, M., Ahamed, M.F., et al.: An adaptive YOLO model for detection of faulty insulators in power transmission network using unmanned aerial vehicle. In: *Proceedings of the 2023 Moratuwa Engineering Research Conference (MERCon)*, pp. 282–287. IEEE (2023)
3. Panigrahy, S., Karmakar, S.: Real-time condition monitoring of transmission line insulators using the YOLO object detection model with a UAV. *IEEE Trans. Instrum. Meas.* **73**, 1–9 (2024)
4. Cheng, Y.: Detection of power line insulator based on enhanced Yolo Model. In: *Proceedings of the 2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, pp. 626–632 (2022)
5. Ali, S., Ghazal, R., Qadeer, N., et al.: A novel approach of botnet detection using hybrid deep learning for enhancing security in IoT networks. *Alex. Eng. J.* **103**, 88–97 (2024)
6. Ebadollahi, S., Gill, B., Khosravani, H., et al.: Assessing the contamination intensity of Porcelain insulators using deep learning networks by UAV. In: *Proceedings of the 2024 IEEE 14th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 0111–0118. IEEE (2024)
7. Liu, J., Hu, M., Dong, J., et al.: Summary of insulator defect detection based on deep learning. *Electric Power Syst. Res.* **224**, 109688 (2023)
8. Yang, S., Li, J., Li, Y., et al.: Imbalanced segmentation for abnormal cotton fiber based on GAN and multiscale residual U-Net. *Alex. Eng. J.* **106**, 25–41 (2024)
9. Liu, Y., Liu, D., Huang, X., et al.: Insulator defect detection with deep learning: a survey. *IET Gener. Transm. Distrib.* **17**(16), 3541–3558 (2023)
10. Behera, T.K., Bakshi, S., Khan, M.A., et al.: A lightweight multi-scale-multiobject deep segmentation architecture for UAV-based consumer applications. *IEEE Trans. Consum. Electron.* **70**(1), 3740–3753 (2024)
11. Shuang, F., Wei, S., Li, Y., et al.: Detail R-CNN: insulator detection based on detail feature enhancement and metric learning. *IEEE Trans. Instrum. Meas.* **72**, 1–14 (2023)
12. Ren, Z., Wang, Y.: Research on the rapid check and identification of insulator faults in transmission lines based on a modified faster RCNN network. In: *Proceedings of the 2022 International Conference on Image Processing and Computer Vision (IPCV)*, pp. 17–21 (2023)
13. Gao, J., Chen, X., Lin, D.: Insulator defect detection based on improved YOLOv5. In: *Proceedings of the 2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT)*, pp. 53–58 (2021)
14. Xie, J., Long, Z., Song, Q., et al.: Visible-light insulator defect detection based on improved YOLOv3. In: *Proceedings of the 2023 3rd International Conference on Electrical Engineering and Mechatronics Technology (ICEEMT)*, pp. 287–292 (2023)

15. Hu, P., Lu, J., Cui, Y., et al.: Insulator defect detection based on faster R-CNN and YOLOv3 algorithm. In: Proceedings of the 2023 IEEE 4th China International Youth Conference On Electrical Engineering (CIYCEE), pp. 1–4 (2023)
16. Wang, T., Zhai, Y., Li, Y., et al.: Insulator defect detection based on ML-YOLOv5 algorithm. *Sensors*. **24**(1), 204 (2023)
17. Zhou, M., Li, B., Wang, J., et al.: Fault detection method of glass insulator aerial image based on the improved YOLOv5. *IEEE Trans. Instrum. Meas.* **72**, 1–10 (2023)
18. Zhao, Y., Wu, J., Chen, W., et al.: A small object real-time detection method for power line inspection in low-illuminance environments. *IEEE Trans. Emerg. Topics Comput. Intell.* (2024). <https://doi.org/10.1109/TETCI.2024.3378651>
19. He, Z., Yang, W., Liu, Y., et al.: Insulator defect detection based on YOLOv8s-SwinT. *Information* **15**(4), 206 (2024)
20. He, M., Qin, L., Deng, X., et al.: MFI-YOLO: multi-fault insulator detection based on an improved YOLOv8. *IEEE Trans. Power Delivery* **39**, 168–179 (2024)
21. Wu, J., Jing, R., Bai, Y., et al.: Small insulator defects detection based on multi-scale feature interaction transformer for UAV-assisted power IoT. *IEEE Internet Things J.* **11**(13), 23410–23427 (2024)
22. Zhang, L., Li, B., Cui, Y., et al.: Research on improved YOLOv8 algorithm for insulator defect detection. *J. Real-Time Image Proc.* **21**(1), 22 (2024)
23. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
24. Woo, S., Park, J., Lee, J.-Y., et al.: Cbam: Convolutional block attention module. In: Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19 (2018)
25. Hou, Q., Zhou, D., Feng, J.: Coordinate attention for efficient mobile network design. In: Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13713–13722 (2021)
26. Zhu, X., Hu, H., Lin, S., et al.: Deformable convnets v2: More deformable, better results. In: Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9308–9316 (2019)
27. Liu, W., Lu, H., Fu, H., et al.: Learning to Upsample by Learning to Sample. In: Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6027–6037 (2023)
28. Shi, W., Caballero, J., Huszár, F., et al.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1874–1883 (2016)
29. Niu, Y., Cheng, W., Shi, C., et al.: YOLOv8-CGRNet: a light-weight object detection network leveraging context guidance and deep residual learning. *Electronics* **13**(1), 43 (2023)
30. Wu, T., Tang, S., Zhang, R., et al.: CGNet: a light-weight context guided network for semantic segmentation. *IEEE Trans. Image Process.* **30**, 1169–1179 (2021)
31. Chen, C., Guo, Z., Zeng, H., et al.: Repghost: a hardware-efficient ghost module via re-parameterization. *arXiv preprint*. <http://www.arxiv.org/2211.06088> (2022)
32. Tong, Z., Chen, Y., Xu, Z., et al.: Wise-IoU: bounding box regression loss with dynamic focusing mechanism. *arXiv preprint*. <http://www.arxiv.org/2301.10051> (2023)
33. Tao, X., Zhang, D., Wang, Z., et al.: Detection of power line insulator defects using aerial images analyzed with convolutional neural networks. *IEEE Trans. Syst. Man Cybern.: Syst.* **50**(4), 1486–1498 (2018)
34. Zheng, J., Wu, H., Zhang, H., et al.: Insulator-defect detection algorithm based on improved YOLOv7. *Sensors*. **22**(22), 8801 (2022)
35. Qi, Y., Li, Y., Du, A.: Research on an insulator defect detection method based on improved YOLOv5. *Appl. Sci.* **13**(9), 5741 (2023)
36. Hong, X., Wang, F., Ma, J.: Improved YOLOv7 model for insulator surface defect detection. In: Proceedings of the 2022 IEEE 5th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), pp. 1667–1672 (2022)
37. Zhang, Y., Wu, Z., Wang, X., et al.: Improved YOLOv8 Insulator Fault Detection Algorithm Based on BiFormer. In: Proceedings of the 2023 IEEE 5th International Conference on Power, Intelligent Computing and Systems (ICPICS), pp. 962–965 (2023).
38. Li, D., Yang, P., Zou, Y.: Optimizing Insulator defect detection with improved DETR models. *Mathematics*. **12**(10), 1507 (2024)
39. Liu, Y., Huang, X., Liu, D.: Weather-domain transfer-based attention YOLO for multi-domain insulator defect detection and classification in UAV images. *Entropy* **26**(2), 136 (2024)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.