

Article

Insulator-YOLO: Transmission Line Insulator Risk Identification Based on Improved YOLOv5

Nan Zhang¹, Jingyi Su^{1,*}, Yang Zhao² and Hua Chen³

¹ Artificial Intelligence Department NARI Group Co., Ltd., Nanjing 210000, China; zhangnan2@sgepri.sgcc.com.cn

² College of Mechanical and Electrical Engineering, Hohai University, Changzhou 213200, China; 221619010116@hhu.edu.cn

³ Department of Mathematics, Hohai University, Nanjing 210098, China; chenhua112@163.com

* Correspondence: sjingyi2021@163.com

Abstract: This study introduces an innovative method for detecting risks in transmission line insulators by developing an optimized variant of YOLOv5, named Insulator-YOLO. The model addresses key challenges in small-defect detection, complex backgrounds, and computational efficiency. By incorporating GhostNetV2 in the backbone to streamline feature extraction and introducing SE and CBAM attention mechanisms, the model enhances its focus on critical features. The Bibi-directional Feature Pyramid Network (BiFPN) is applied to enhance multi-scale feature fusion, and the integration of CIoU and NWD loss functions optimizes bounding box regression, achieving higher accuracy. Additionally, focal loss mitigates the imbalance between positive and negative samples, leading to more accurate and robust defect detection. Extensive evaluations demonstrate that Insulator-YOLO significantly improves detection accuracy and efficiency in real-world power line insulator defects, providing a reliable solution for maintaining the integrity of transmission systems.

Keywords: computer vision; risk identification; electricity transmission line insulator; deep neural network; YOLOv5



Citation: Zhang, N.; Su, J.; Zhao, Y.; Chen, H. Insulator-YOLO: Transmission Line Insulator Risk Identification Based on Improved YOLOv5. *Processes* **2024**, *12*, 2552. <https://doi.org/10.3390/pr12112552>

Academic Editor: Massimo Caruso

Received: 12 October 2024

Revised: 5 November 2024

Accepted: 13 November 2024

Published: 15 November 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the ongoing expansion of the power system [1] and with transmission line insulators being a core component in high-voltage circuits, the reliability of transmission line these insulators is closely linked to the operational safety of transmission lines. Insulators not only assume the role of supporting the power conductor but also need to effectively isolate the voltage and prevent current leakage. However, extended exposure to outdoor conditions renders insulators susceptible to environmental factors and external influences, such as pollution, climate change, mechanical stress, and other factors, which will lead to cracks, dirt, or other defects on the insulator surface. Once these defects occur, they will substantially increase the risk of flashover, breakdown, etc., which may eventually lead to serious power accidents. In addition, because large-span overhead transmission lines are in an environment with complex terrain and changing climate, traditional manual inspection [2] requires operators to patrol on foot and measure manually with instruments. This inspection method not only has low efficiency and high risk factors but also has low accuracy and high cost of inspection results, which makes it difficult to meet the requirements of daily monitoring and inspection.

With technological advancements, UAVs have become an affordable solution for inspecting power line systems [3], making vision-based insulator defect detection a mainstream method. UAV inspections cover extensive transmission line areas, significantly improving efficiency compared to traditional, error-prone visual assessments by inspectors. Computer vision (CV) techniques enhance target detection and image recognition, enabling

automated analysis and identification of inspection data [4]. These technologies provide power grid personnel with advanced tools for processing large datasets, accurately identifying faults, and reducing error rates [5]. Overall, CV significantly improves inspection efficiency and plays a crucial role in ensuring grid safety and stability.

Nowadays, deep learning models have been widely applied in various CV fields. Deep learning-based defect detection algorithms can generally be categorized into three main types. The first type consists of two-stage algorithms that combine detection frameworks with high-precision classifiers, such as R-FCN [6] and Faster faster R-CNN [7]. Although these methods achieve high accuracy, the complexity of their advanced network architecture leads to reduced processing speeds, limiting their suitability for real-time detection. For example, an improved faster R-CNN model was applied to railroad cotter pin defect detection in [8], achieving a mean accuracy (mAP) of 97.87%, but only 18.89 frames per second in detection speed. The second category comprises transformer-based approaches that have recently gained attention [9], incorporating attention mechanisms into defect detection. For example, De-Jun Cheng et al. [10] proposed the adaptive global dynamic detection transformer (AGD-DETR) framework, which effectively solves the problem of detecting low-contrast, multiscale casting defects by means of data augmentation, feature refinement, and global dynamic detection methods. The literature [11] proposes an improved Swin transformer model that enhances the multiscale feature capture capability by replacing the traditional shift window with a shunted large–small window mechanism. The simultaneous introduction of the local connectivity module further enhances the boundary interaction between markers, resulting in significant performance improvement on the bearing defect detection task. However, the computational complexity and memory occupation of the transformer model are high, especially when dealing with long sequences, and the self-attention mechanism has a time complexity of $O(n^2)$, which restricts its application on resource-constrained devices. The third category encompasses single-stage algorithms grounded in regression models, exemplified by the single-shot detector (SSD) [12] and YOLO [13–16]. These models provide expedited inference and enhanced practicality, rendering them particularly suitable for real-time defect detection. For example, Wang et al. [17] presented the YOLO-RLC model by introducing a large kernel backbone and a bi-directional weighted feature fusion network, which significantly enhances both the precision and speed of defect identification in printed circuit boards under complex backgrounds. Tao et al. [18] integrated the YOLO framework with traditional image processing techniques for detecting gaps in switches under complex conditions. Kumar et al. [19] compared detection methods like R-CNN, SSD, and YOLO, revealing that the YOLO network demonstrated superior speed and accuracy. The evolution of object detection frameworks has led to models like YOLOv5 and YOLOv8. YOLOv5 features an efficient architecture for real-time inference, making it ideal for applications requiring rapid decision-making. Its simpler structure enhances training speed and minimizes overfitting, especially for datasets with limited labeled examples. Additionally, its user-friendly implementation and robust documentation facilitate quick deployment in real-world scenarios. In contrast, YOLOv8 offers a more complex architecture aimed at improving accuracy and adaptability across various tasks. However, this complexity can lead to longer inference times and higher computational demands, limiting its use in resource-constrained environments. While YOLOv8 enhances accuracy and robustness with state-of-the-art features, its higher resource requirements may not be feasible in all contexts. Given these considerations, we chose YOLOv5 for our study on insulator fault detection. Its proven efficiency, reliable performance, ease of use, and suitability for real-time applications align perfectly with our objectives, enabling effective fault detection while optimizing resource utilization.

However, in the small-objective task of insulator defect detection [20], YOLOv5 still has some limitations: (1) Insulator defects tend to be small in area and easily confused with the background, leading to insufficient detection accuracy. (2) For resource-limited devices, such as UAVs or embedded systems, meeting real-time requirements is challenging due to the computational complexity and inference speed associated with the model. (3) In defect

detection tasks, the imbalance between positive and negative samples results in high false positives and missed detections.

We have optimized the YOLOv5 model for insulator defect detection in power transmission lines, emphasizing higher detection precision, improved computational speed, and enhanced performance on small-scale targets. The upgraded model, Insulator-YOLO, offers several major advancements:

1. Backbone network optimization to reduce computational complexity: This paper replaces the original CSPDarknet53 structure with GhostNetV2 to minimize redundant computations and improve inference speed. Additionally, the SE module is integrated to enhance inter-channel feature adaptivity, thereby improving the network's ability to capture important features.
2. To boost small-target detection, the CBAM attention mechanism is incorporated into the backbone network. This improves detection in complex backgrounds by enhancing the model's focus on relevant feature channels and spatial regions, leading to the more precise identification of insulator defects.
3. Improved feature fusion network: The BiFPN is utilized to enhance multiscale feature fusion, allowing the model to capture small defects more effectively through top-down and bottom-up fusion paths. This method proves especially beneficial in multiscale detection scenarios.
4. To enhance the robustness of small-target detection, a new loss function combining NWD and focal loss is introduced. By improving bounding box regression and classification loss, the model's accuracy and robustness for small targets are significantly enhanced, especially in imbalanced positive and negative sample situations. This effectively reduces false detections and missed detections.

2. Proposed Method

2.1. Image Preprocessing of Electricity Transmission Line Insulators

We propose a novel image enhancement technique that integrates non-local means (NLM) filtering and Laplacian sharpening to effectively reduce noise while preserving important image details. The NLM filtering is applied to the input image I to obtain a denoised output $I'(p)$ for each pixel p as follows:

$$I'(p) = \frac{1}{Z(p)} \sum_{q \in \Omega} I(q) \cdot w(p, q), \quad (1)$$

where Ω denotes the search window, and the weight $w(p, q)$ is computed based on the similarity between the pixel neighborhoods B_p and B_q ,

$$w(p, q) = e^{-\frac{\|I(B_p) - I(B_q)\|^2}{h^2}}, \quad (2)$$

here h is a parameter that controls the sensitivity of similarity measurement. The normalization factor $Z(p)$ is calculated as follows:

$$Z(p) = \sum_{q \in \Omega} w(p, q) \quad (3)$$

Subsequently, to enhance the image edges and details, we apply the Laplacian operator $L(x, y)$, which is expressed as follows:

$$L(x, y) = \frac{\partial^2 I'(x, y)}{\partial x^2} + \frac{\partial^2 I'(x, y)}{\partial y^2} \quad (4)$$

The Laplacian can also be represented by the convolution of the image with the Laplacian kernel I :

$$L(p) = I' * \text{Laplacian_Kernel} \quad (5)$$

where the Laplacian kernel is given by

$$\text{Laplacian_Kernel} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (6)$$

The sharpening process utilizes the Laplacian to adjust the pixel values, resulting in the enhanced image $I''(p)$,

$$I''(p) = I'(p) + k \cdot L(p), \quad (7)$$

where k is a parameter that controls the degree of sharpening. The final output can be expressed in terms of the gradient magnitude,

$$G(p) = \sqrt{G_x(p)^2 + G_y(p)^2}, \quad (8)$$

where G_x and G_y represent the gradients calculated in the horizontal and vertical directions. This combined approach not only effectively reduces noise but also enhances the clarity and quality of the image, making it more suitable for further analysis and applications.

2.2. Original YOLOv5 Model

The YOLO series has undergone four major updates. YOLOv5 retains the core structure of the YOLO series and is composed of three main components [21]. The model is primarily composed of three parts: backbone, neck, and head. The backbone integrates CBS, C3_X, and SPPF structures, with CBS responsible for reducing image dimensions while preserving essential information. C3_X enhances feature extraction capabilities and reduces computational complexity by adding convolutional layers and optimizing network depth. SPPF (spatial pyramid pooling) enhances the feature extraction capabilities and reduces the computational complexity through the neck and consists of the C3_X_F structure and the FPN + PAN architecture, in which the FPN [22] (feature pyramid network) better conveys high-level information through top-down upsampling connections but may lead to blurring of the underlying details. PAN [23] (path aggregation network) reinforces the underlying features through bottom-up paths and representation of features, which compensates for the lack of detail recovery in FPN. The FPN + PAN network structure can be seen in Figure 1. The output layer utilizes the CIOU_Loss function to optimize bounding box regression by considering both the angle and center distance of the boxes. Additionally, non-maximal suppression (NMS) eliminates overlapping redundant boxes, ensuring that detection results remain accurate and stable. This overall architecture equips YOLOv5 with efficient detection performance, making it especially suitable for detecting insulator defects in transmission lines. Figure 2 illustrates the architecture of the YOLOv5 network.

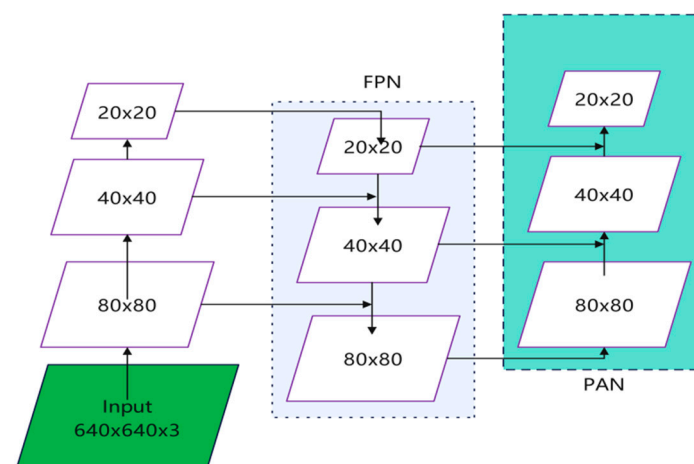


Figure 1. FPN + PAN network structure.

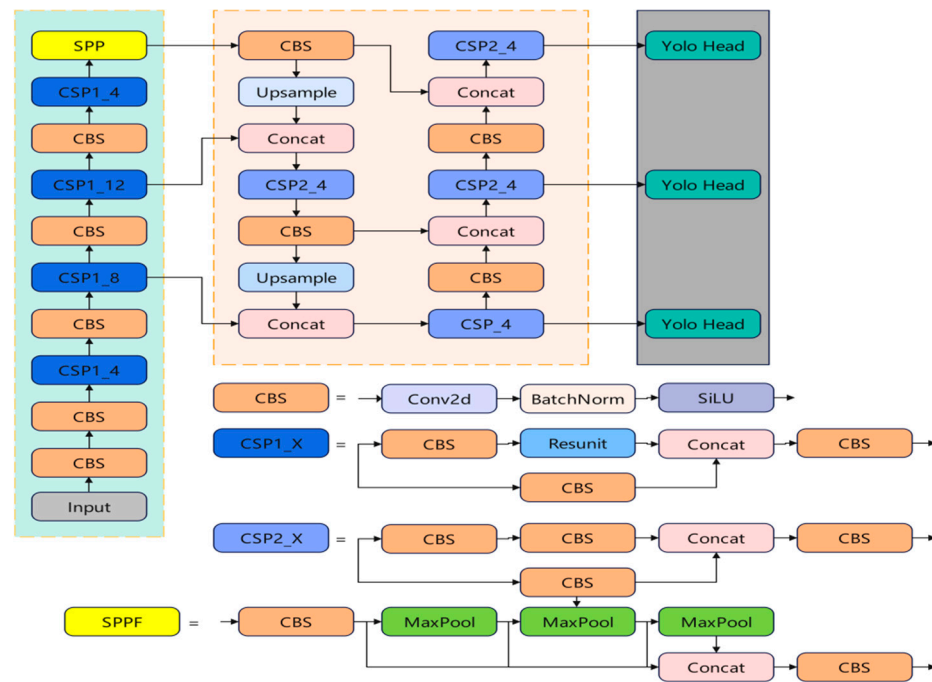


Figure 2. YOLOv5 network structure.

2.3. Insulator-YOLO Model

To enhance YOLOv5's performance in detecting transmission line insulator faults, this paper presents the Insulator-YOLO model, which optimizes the network structure specifically for this task. Firstly, GhostNet v2 [24] replaces the original CSPDarknet53, decreasing computational complexity and parameter quantity using efficient Ghost modules and depthwise separable convolutions, while preserving strong feature extraction capabilities. Secondly, the SE module [25] and CBAM [26] are incorporated to enhance feature map representation across channel and spatial dimensions, respectively, improving the recognition of small objects and complex backgrounds. For feature fusion, the BiFPN is employed to optimize multiscale feature integration through bidirectional flow, thereby enhancing small-object detection accuracy. Finally, the accuracy of bounding box regression is improved by combining the CIoU (complete intersection over union) and NWD (normalized Wasserstein distance) [27] loss functions, and focal loss is introduced to address the imbalance between positive and negative samples, further enhancing detection performance. These improvements to Insulator-YOLO jointly enhance the performance of the network in detecting insulator defects in complex environments and enable it to show higher detection accuracy and efficiency in practical applications. The framework of the Insulator-YOLO model network is shown in Figure 3.

2.3.1. Optimization of YOLOv5 Backbone Network Based on Ghost-SE Module

In YOLOv5, the backbone module takes on the task of feature extraction, and although the original CSPDarknet53 structure can effectively extract multiscale features, its convolutional computation is large. To enhance computational efficiency and improve the inference speed of the model, as well as its performance in applications on resource-constrained devices, we introduce two improvements to the backbone of YOLOv5: replacing the original CSP module with GhostNetV2 and introducing the SE module to improve the inter-channel feature adaptivity. The convolution operation in CSPDarknet53 is shown in Formula (9),

$$\mathcal{O}(\text{CSP}) = \sum_{l=1}^L H_l \times W_l \times C_l \times C_{\text{in},l} \times K_l^2, \quad (9)$$

here H_l , W_l indicate the height and width of the feature map at layer l ; C_l denotes the output channels, and $C_{in,l}$ represents the input channels; and K_l is the convolution kernel size, and l stands for the total number of convolution layers.

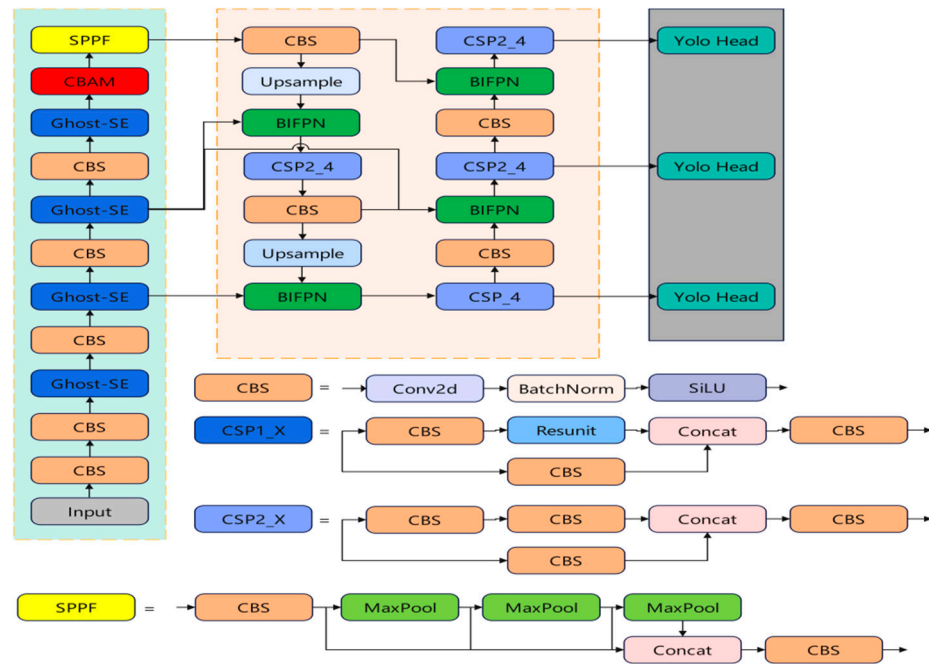


Figure 3. Network structure diagram of our Insulator-YOLO algorithm.

To reduce the redundant computation in convolutional operation, this paper adopts the GhostNetV2 structure to replace the CSP module. Figure 4 shows the GhostNetV2 bottleneck module. The core idea of GhostNetV2 is to reduce unnecessary computation by generating features in two steps. Compared with the traditional convolution operation, GhostNetV2 extracts the primary features using a limited number of standard convolutions and subsequently produces additional ‘ghost features’ through a lightweight linear operation, effectively minimizing computational load. The computation process of GhostNetV2 is as follows:

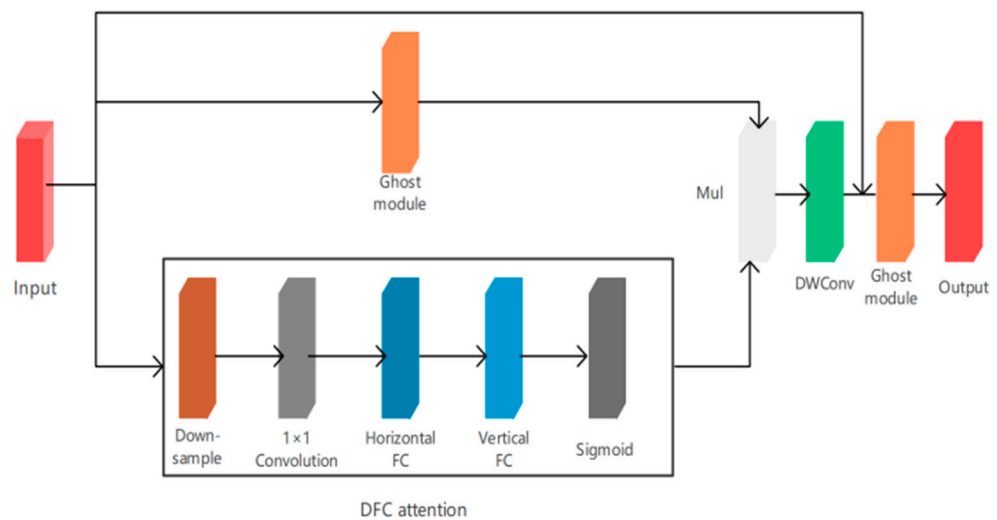


Figure 4. GhostNetV2 bottleneck.

Step 1: Master feature generation. A small number of core features are generated by standard convolution F_1 , whose computational complexity is as follows:

$$F_1 = \text{Conv}(X; W_1), \quad \mathcal{O}(F_1) = H \times W \times C_1 \times C_{\text{in}} \times K_1^2, \quad (10)$$

among them, $C_1 \ll C_{\text{out}}$ (the number of feature channels produced in the initial step is reduced).

Step 2: Ghost feature generation on the basis of generating the main features, additional ghost features are generated using a lightweight linear transformation F_2 , whose computational complexity is as follows:

$$F_2 = \text{LinearTransform}(F_1; W_2), \quad \mathcal{O}(F_2) = H \times W \times (C_{\text{out}} - C_1) \times K_2^2, \quad (11)$$

where $(C_{\text{out}} - C_1)$ denotes the number of additional channels generated by the lightweight operation.

The complexity of GhostNetV2 is shown in Formula (13):

$$F = F_1 \cup F_2 \quad (12)$$

$$\mathcal{O}(\text{GNetV2}) = H \times W \times C_1 \times C_{\text{in}} \times K_1^2 + H \times W \times (C_{\text{out}} - C_1) \times K_2^2 \quad (13)$$

Through this two-step feature generation mechanism, GhostNetV2 effectively decreases the computational complexity and parameter size compared to the original CSP module, particularly during deep feature extraction. This enhancement accelerates the network inference speed by reducing redundant feature computations.

Although GhostNetV2 reduces the computational effort by generating features in two steps, it is still fixed in the channel selection of features. To strengthen the network's attention to key channel characteristics, the SE module is integrated into the feature extraction process. Figure 5 shows the SE module. The SE module improves network feature expression by dynamically assigning weights to each channel. It comprises two main components: squeeze and excitation.

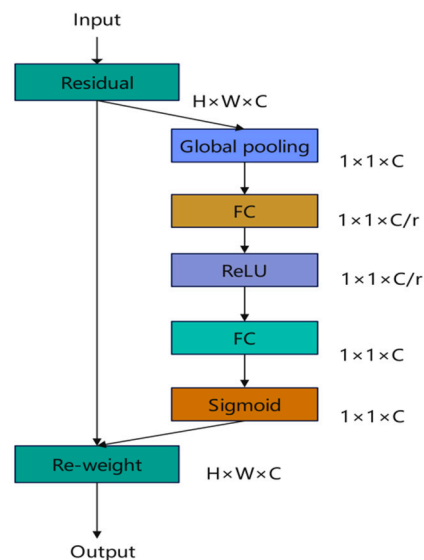


Figure 5. Schematic diagram of SE attention mechanism.

Firstly, the input feature map $F \in \mathbb{R}^{H \times W \times C}$ is obtained. Global average pooling is performed to generate a global description vector for each channel $z \in \mathbb{R}^C$, which is computed by the following formula:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W F(i, j, c), \quad (14)$$

where z_c is the global average of the first c global mean of the individual channels.

Then, after passing through two fully connected layers and nonlinear activation functions, the weights for each channel are learned, calculated as follows:

$$s = \sigma(W_2 \cdot \text{ReLU}(W_1 \cdot z)), \quad (15)$$

where $s \in \mathbb{R}^C$ represents the weights of each channel; the W_1 and W_2 denote the weight matrices of the fully connected layer, respectively, and σ denotes the sigmoid activation function.

Lastly, the weight vector s is applied channel wise to the original feature map to derive the weighted output feature map,

$$F'(i, j, c) = F(i, j, c) \times s_c \quad (16)$$

Thus, with the introduction of the SE module, the network can selectively prioritize essential feature channels during extraction, thus further enhancing the feature representation capability and detection accuracy.

The YOLOv5 feature extraction module after combining GhostNetV2 and SE not only greatly reduces the computation amount but also improves the ability of adaptive attention to features through SE module. Finally, the computational complexity is as follows:

$$\mathcal{O}(\text{Ghost-SE}) = \sum_{l=1}^L (H_l \times W_l \times C_{1,l} \times C_{in,l} \times K_1^2 + H_l \times W_l \times (C_{out,l}) \times K_2^2 + \mathcal{O}(\text{SE})), \quad (17)$$

among them, $\mathcal{O}(\text{SE})$ is the additional computational effort of the SE module, mainly for global pooling and fully connected operations.

2.3.2. CBAM Attention Mechanism

Insulator defect signals typically occupy a limited number of pixels, categorizing them as a small-object detection task, and are easily affected by factors such as background interference. To overcome YOLOv5's challenges in detecting small objects and to improve the network's focus on targets of interest, this study introduces the CBAM attention mechanism following the last Ghost-SE module. Figure 6 shows the CBAM module's structure. CBAM enhances the network's capability to detect multiscale and deformed targets by selectively weighting different feature maps across channel and spatial dimensions. It integrates the channel attention module (CAM) and the spatial attention module (SAM), thereby enhancing accuracy in detecting small defects. The mechanism first selects important channel information in the feature map through CAM and then further focuses on important spatial regions of the feature map through SAM.

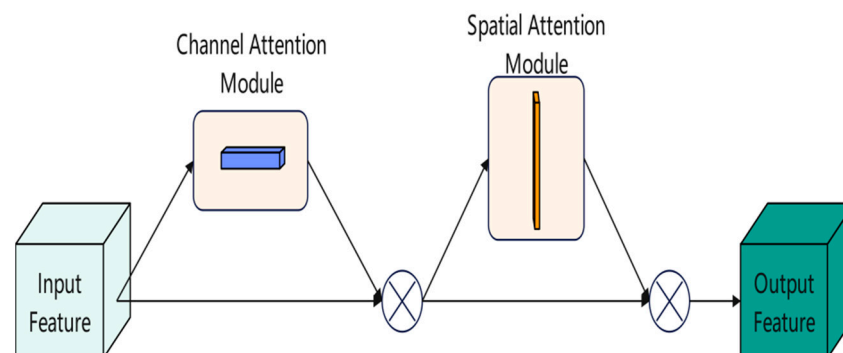


Figure 6. The modular structure of CBAM.

Given the input feature map $F \in \mathbb{R}^{C \times H \times W}$, the CBAM module obtains a one-dimensional channel attention graph by sequential inference $\mathcal{M}_C \in \mathbb{R}^{C \times 1 \times 1}$ and constructs spatial attention maps in two dimensions $\mathcal{M}_S \in \mathbb{R}^{1 \times H \times W}$.

Channel attention generates two different global description vectors by applying global average pooling and global maximum pooling to the input feature map $z_{avg}, z_{max} \in \mathbb{R}^C$. These feature vectors undergo processing via a shared fully connected layer, followed by applying a sigmoid activation function to generate the channel attention map,

$$M_C = \sigma(MLP(z_{avg}) + MLP(z_{max})) \quad (18)$$

where M_C is the channel attentional weight and σ denotes the sigmoid function, and MLP stands for multilayer perceptual machine.

Next, the channel attention map M_C is mapped onto the input feature map F on each channel of the input feature map to obtain the channel-weighted feature map F' :

$$F' = M_C(F) \otimes F \quad (19)$$

where \otimes represents the element-wise multiplication operation.

Spatial attention is then generated by performing global average pooling and global maximum pooling on the F' performing global average pooling and global maximum pooling over channel dimensions to produce two spatial attention feature maps $F_{avg}^{spatial}, F_{max}^{spatial} \in \mathbb{R}^{1 \times H \times W}$. The spatial attention maps are then generated by the convolution operation M_S :

$$M_S = \sigma\left(\text{Conv}_{7 \times 7}\left(\left[F_{avg}^{spatial}, F_{max}^{spatial}\right]\right)\right), \quad (20)$$

The spatial attention maps M_S are employed on the channel-weighted feature map F' onto the final output feature map F'' :

$$F'' = M_S(F') \otimes F', \quad (21)$$

The final output feature map F'' is a channel- and spatially weighted feature map that contains more information that is useful for the defect detection task than the original feature map F . The integration of CBAM enhances the network's sensitivity to the scale, shape, and location of targets, thereby greatly enhancing small-object detection performance.

2.3.3. Bidirectional Eigenpyramid Networks

In transmission line insulator fault detection, the detection accuracy for small defects puts higher requirements on feature fusion. Since insulator defects may appear at different scales and different locations, effective multiscale feature fusion is the key to improving the detection accuracy. Traditionally, YOLOv5 uses FPN and PAN for multiscale feature fusion, but these methods suffer from the problem of level-specific information loss during the fusion process, especially when dealing with small targets, detailed features are easily lost. To solve this problem, this paper introduces a bidirectional feature pyramid network in the neck module of YOLOv5 to enhance multiscale feature fusion, especially in insulator defect detection, to better capture small defect features.

The core idea of the BiFPN [28] is to realize the bidirectional fusion of multiscale features through top-down and bottom-up bidirectional paths. The feature fusion operation in each path is realized by weighted convolution operation, and different weights are assigned to the features of different scales in the fusion process. Compared with the traditional FPN, BiFPN adds a flexible feature fusion mechanism to make features interoperable between different levels as shown in Figure 7.

Let the feature layers generated by the network be $\{P_3, P_4, P_5\}$, where P_3 denotes shallow feature maps with high spatial resolution and P_5 denotes deep feature maps containing more semantic information. In the top-down path, the deep feature maps are convolutionally upsampled and fused with the shallow feature maps. Let the convolution kernel be $K_{i,j}$, and the input feature map be P_i . In bottom-up path, shallow feature maps are fused

with deeper feature maps by convolutional downsampling process. The downsampling operation can be implemented by pooling layers or convolution as follows:

$$P_i^{td} = \alpha_1 \cdot Conv(P_i, K_{i,j}) + \alpha_2 \cdot Up(Conv(P_{i+1}, K_{i+1,j})), \quad i = 3, 4, \quad (22)$$

$$P_i^{bu} = \beta_1 \cdot Conv(P_i, K_{i,j}) + \beta_2 \cdot Down(Conv(P_{i-1}, K_{i-1,j})), \quad i = 4, 5, \quad (23)$$

where $Conv(P, K)$ denotes the use of the convolution kernel K on the feature map P to perform a convolution operation, $Up(\cdot)$ denotes the upsampling operation, and α_1 and α_2 are the learned weighting parameters, and $\alpha_1 + \alpha_2 = 1$, which is used to balance the fusion of features at different scales.

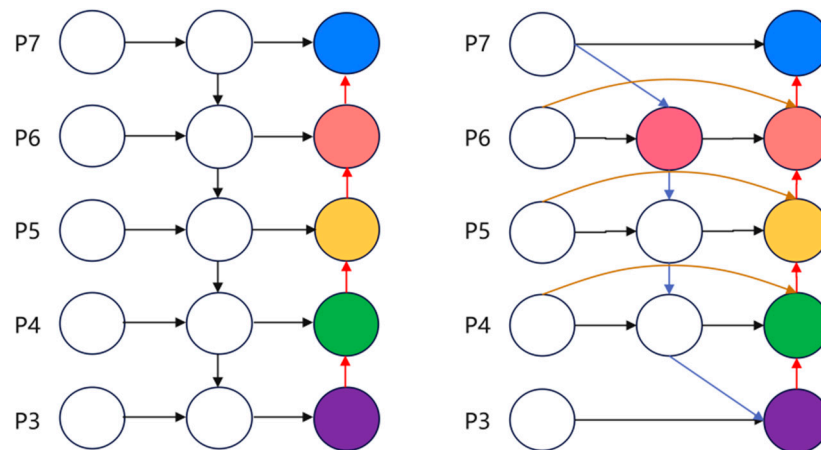


Figure 7. Structure comparison of PANet (left) and BiFPN (right).

The output feature map P_i^{out} can be expressed as follows:

$$P_i^{out} = \gamma_1 \cdot P_i^{td} + \gamma_2 \cdot P_i^{bu}, \quad i = 3, 4, 5, \quad (24)$$

of these, the γ_1 and γ_2 are the fusion weights, respectively, and are used to control the influence of bidirectional paths on the final feature output.

2.3.4. Improved Loss Function

In the transmission line insulator fault detection task, insulator defects usually have a small area and are easily confused with the background. This necessitates a more precise bounding box regression and classification from the target detection network. The original loss function of YOLOv5, although capable of handling most of the detection tasks, shows certain limitations in small-object detection, especially when the defective area accounts for a small proportion of the image. To address this issue, we improved the loss function of YOLOv5 by introducing NWD (normalized Wasserstein distance) and focal loss, and combined the existing CIoU (complete intersection over union) loss function. This approach aims to improve detection accuracy for small defects and mitigate the imbalance between positive and negative samples.

The original loss function of YOLOv5 comprises three components: bounding box regression loss \mathcal{L}_{box} , confidence loss \mathcal{L}_{obj} , and classification loss \mathcal{L}_{cls} . Among them, the bounding box regression loss adopts the CIoU loss, which not only considers the intersection over union (IoU) but also introduces the consistency of the distance from the center point and the aspect ratio. This enhances the accuracy of bounding box shape and position regression:

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b_{gt})}{c^2} + \alpha v, \quad (25)$$

where the b and b_{gt} denote the centroids of the prediction and real frames, respectively; and $\rho(b, b_{gt})$ represents the Euclidean distance between the center points; and c denotes the diagonal length of the minimum enclosing rectangle of the prediction frame and the real frame. Parameter v is used to measure the consistency of the aspect ratio of the predicted box, and the real box is defined as follows:

$$v = \frac{4}{\pi^2} \left(\arctan \frac{w_{gt}}{h_{gt}} - \arctan \frac{w}{h} \right)^2, \quad (26)$$

where w_{gt}, h_{gt} represent the real box's width and height, and w, h denote the predicted box's dimensions. The parameter α is a balancing factor. Although CloU loss performs well in most target detection tasks, it falls short when dealing with small objects. When the IoU of the prediction frame to the real frame is small (e.g., when detecting small objects), CloU converges slowly to the bounding box, especially when the IoU is close to 0, which makes it difficult to provide effective gradient updates. This issue is especially evident in the defect detection of transmission line insulators, highlighting the need for a new loss function to address the limitations of CloU.

To enhance the detection of small objects, we introduce the NWD loss. NWD has better robustness for small-object detection by using the Wasserstein distance to measure the similarity between the prediction frame and the real frame, which is especially suitable for dealing with targets of different scales. The Wasserstein distance first models the predictor frame and the real frame as a two-dimensional Gaussian distribution \mathcal{N}_a and \mathcal{N}_b ; the mean of each distribution is the center point of the bounding box, and the variance is the width and height of the box. It is defined as follows:

$$W_2^2(\mathcal{N}_a, \mathcal{N}_b) = \|c_a - c_b\|^2 + \left(\frac{w_a}{2} - \frac{w_b}{2}\right)^2 + \left(\frac{h_a}{2} - \frac{h_b}{2}\right)^2, \quad (27)$$

where $c_a = (c_{xa}, c_{ya})$ and $c_b = (c_{xb}, c_{yb})$ denote the coordinates of the centers of the predicted and real frames, respectively. The Wasserstein distance provides a more precise measure of similarity between predicted and real boxes. However, as a distance metric, it cannot be directly applied to quantify the similarity of bounding boxes; the NWD loss is normalized and defined as follows:

$$\text{NWD}(\mathcal{N}_a, \mathcal{N}_b) = \exp\left(-\frac{1}{C} \sqrt{W_2^2(\mathcal{N}_a, \mathcal{N}_b)}\right), \quad (28)$$

here C is a dataset-specific constant. The NWD loss is ultimately defined as follows:

$$\mathcal{L}_{NWD} = 1 - \text{NWD}(\mathcal{N}_a, \mathcal{N}_b) \quad (29)$$

The NWD loss measures the geometric similarity of the predicted frame to the real one via the Wasserstein distance metric, showing better sensitivity especially for small targets.

To further enhance the regression accuracy of YOLOv5 in small-defect detection, we combined CloU and NWD [29] to form a new bounding box regression loss function. CloU can better handle larger targets and ensure that the centroid and aspect ratio of the predicted box and the real box are consistent as much as possible, whereas NWD is able to enhance the robustness of the detection of small objects. The combined loss function is defined as follows:

$$\mathcal{L}_{box} = \alpha \mathcal{L}_{CloU} + (1 - \alpha) \mathcal{L}_{NWD}, \quad (30)$$

where α is a weight parameter to adjust the share of CloU and NWD in the bounding box regression loss. This combination enables the network to adapt its regression strategy for targets of varying scales, enhancing the detection accuracy of small targets.

In the insulator risk detection task, the number of positive and negative samples is often severely imbalanced due to the small and sparse defect region. The traditional BCE

(binary cross entropy) loss function is easily interfered with by negative samples, which leads to insufficient attention to the positive samples during the model training process, thus affecting the detection accuracy. For this reason, this paper introduces the focal loss [30], which is used to replace the confidence loss and classification loss in YOLOv5. \mathcal{L}_{obj} and classification loss \mathcal{L}_{cls} , which is defined as follows:

$$\mathcal{L}_{Focal}(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t), \quad (31)$$

where α_t is the balancing factor for positive and negative samples, and γ is a parameter that regulates the weights of difficult samples, and p_t denotes the predicted probability of correct classification. By decreasing the weight of simple background samples, focal loss encourages the network to focus more on challenging foreground targets, thereby enhancing the model's detection of small objects.

Ultimately, the improved YOLOv5 loss function consists of a bounding box regression loss \mathcal{L}_{box} , confidence loss \mathcal{L}_{obj} , and classification loss \mathcal{L}_{cls} which are composed of confidence and classification losses using focal loss for optimization with the following formulas:

$$\mathcal{L}_{total} = \mathcal{L}_{box} + \mathcal{L}_{Focal}(\mathcal{L}_{obj} + \mathcal{L}_{cls}) \quad (32)$$

By introducing NWD and focal loss, the improved loss function can effectively enhance the ability of the model to detect small objects and reduce the false detection phenomenon caused by the positive and negative sample imbalance problem.

3. Experiments

3.1. Introduction to Data

We used the SFID insulator self-explosion defect dataset, which contains two categories: normal insulators and self-explosion defect insulators. For this study, we extracted 10,000 high-resolution images from the dataset, covering a variety of complex outdoor scenes. These scenes include different weather conditions (e.g., sunny or cloudy) and complex backgrounds (e.g., trees, utility poles, and buildings), which provide an important testing environment for the robustness of the model in real applications. Figure 8 shows data examples of insulator self-detonation defects.



Figure 8. Part of the insulator self-explosion defect data.

The dataset uses the YOLO format for image annotations, in which each label includes five parameters: category number, normalized center coordinates, and width and height of the target box. For comprehensive model training, validation, and testing, the dataset is divided into 70% training, 15% validation, and 15% test sets. This results in 7000 images for training, and 1500 images each for validation and testing. Such a distribution enables the comprehensive evaluation of the model's performance across different stages. Figure 8 for self-exploding insulators. The data labels are shown in Table 1.

Table 1. Sample data label.

Serial Number	Labels	Bounding Box
005368	0	0.465278, 0.604745, 0.727431, 0.153935
	1	0.741753, 0.613426, 0.058160, 0.041667
005958	0	0.424479, 0.684606, 0.607639, 0.408565
	1	0.236545, 0.564815, 0.054688, 0.057870
006586	0	0.516493, 0.432292, 0.680556, 0.228009
	1	0.753472, 0.358796, 0.059028, 0.057870
fogged_006787	0	0.293981, 0.527778, 0.425926, 0.602431
	1	0.418981, 0.710069, 0.062500, 0.052083

3.2. Parameterization

This experiment was conducted on a compute node equipped with high-performance GPUs, including NVIDIA Tesla V100 32GB GPUs, Intel Xeon Gold 6226R CPUs, 512GB RAM. The deep learning framework is PyTorch 1.8 with CUDA version 11.2. cuDNN and NVIDIA Apex are used for mixed-precision training to accelerate the training and optimization process. Meanwhile, the training process uses distributed data parallelism to improve the training efficiency. The configuration of model parameters is shown in Table 2.

Table 2. Model hyperparameter settings.

Parameter Name	Parameter Value
Optimizer	SGD
Initial learning rate	0.01
Momentum	0.937
Weight decay	0.0005
Batch size	16
Training round	300
Loss function	CIoU + focal loss
Data enhancement	Random crop, zoom, flip

In the experiments, a comprehensive set of evaluation metrics was utilized to assess the model's training effectiveness. This set includes traditional performance metrics such as mean average precision (mAP), accuracy, and recall rate. It also features metrics reflecting model complexity and computational efficiency, including giga floating-point operations (GFLOPs), which indicates resource demands, model parameters, and frames per second (FPS) for evaluating inference speed. Table 3 shows the confusion matrix. The specific formulas for these metrics are provided below:

$$Precision = \frac{TP}{TP + FP} \quad (33)$$

$$Recall = \frac{TP}{TP + FN} \quad (34)$$

$$AP = \int_0^1 P(R) dR \quad (35)$$

$$mAP = \frac{\sum_{i=1}^C AP}{C} \quad (36)$$

Table 3. Confusion matrix.

	Actual	Prediction
TP	positive	positive
TN	negative	negative
FP	positive	negative
FN	negative	positive

3.3. Ablation Experiments

To comprehensively evaluate the effectiveness of the Insulator-YOLO, we conducted ablation experiments to systematically evaluate the impact of each component on model performance. All models were trained under identical hardware and software conditions to ensure consistent and comparable results.

The experiments include the following groups: firstly, we adopt the original YOLOv5 model as the baseline and observe its detection performance on standard datasets. Second, for the baseline model, we introduce GhostNetV2 as the backbone network (GhostNet) and analyze its impact on detection accuracy and computational efficiency. Then, based on the GhostNetV2, we add the SE module (GhostNet + SE) to evaluate its enhancement on feature adaptive capability. Then, the CBAM attention mechanism (GhostNet + SE + CBAM) is further introduced to explore its improvement on the model's small-target detection performance. Finally, the BiFPN structure (GhostNet + SE + CBAM + BiFPN) is used to analyze the enhancement effect of multiscale feature fusion on the detection performance.

The experimental results are assessed using metrics including mAP, P, R, and F1, as detailed in Table 4. The data demonstrate that the incremental introduction of each module notably enhances the overall model performance, particularly following the incorporation of CBAM and BiFPN. Notably, the introduction of CBAM and BiFPN significantly improves detection accuracy and the ability to detect small targets.

Table 4. Ablation experiment of Insulator-YOLO.

Model Configuration	mAP (%)	P (%)	R (%)	F1 (%)
Original YOLOv5	78.50	76.21	80.1	78.15
GhostNetV2	82.83	79.24	82.31	80.69
GhostNetV2 + SE	83.51	81.32	84.24	82.71
GhostNetV2 + SE + CBAM	87.15	83.00	86.01	84.50
Insulator-YOLO	89.65	87.92	90.11	86.02

From the results, it is observed that the mAP of the baseline model was 78.5%, while the introduction of GhostNetV2 improved the model performance to 82.83%. By incrementally introducing the SE and CBAM, the mAP further increased to 87.15%. Finally, with the adoption of the BiFPN structure, the model reached 89.65% in mAP, indicating that multiscale feature fusion significantly improves the detection ability of the model. These results validate the important role of each module in improving the insulator fault detection performance.

3.4. Comparative Experiments

We conducted comparisons with multiple popular object detection methods. The comparison algorithms include faster R-CNN, RetinaNet, SSD, EfficientDet, CenterNet, YOLOv5, YOLOv7, and YOLOv8. The selected algorithms cover two-stage detection models, single-stage detection models, and high-efficiency models, which can provide

a comprehensive benchmark reference for this study. Figures 9 and 10 demonstrate the higher convergence speed and training accuracy of the algorithmic models.

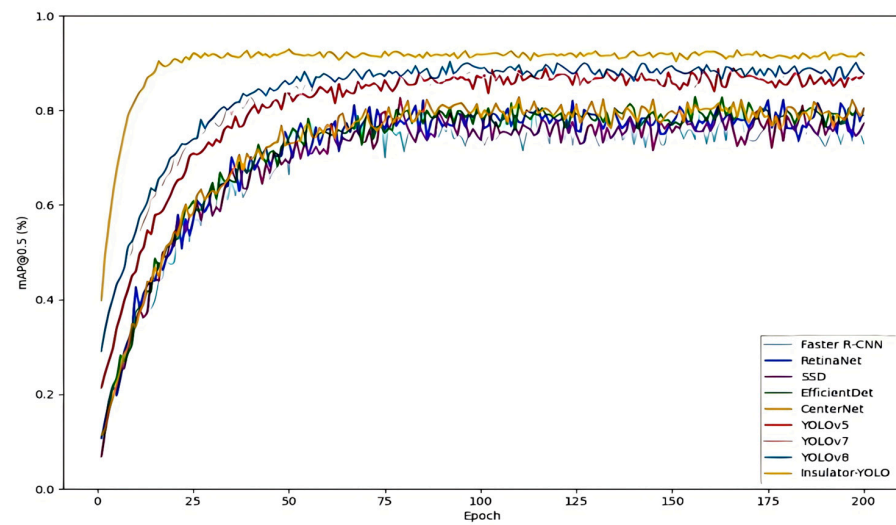


Figure 9. Model mAP curve comparison diagram.

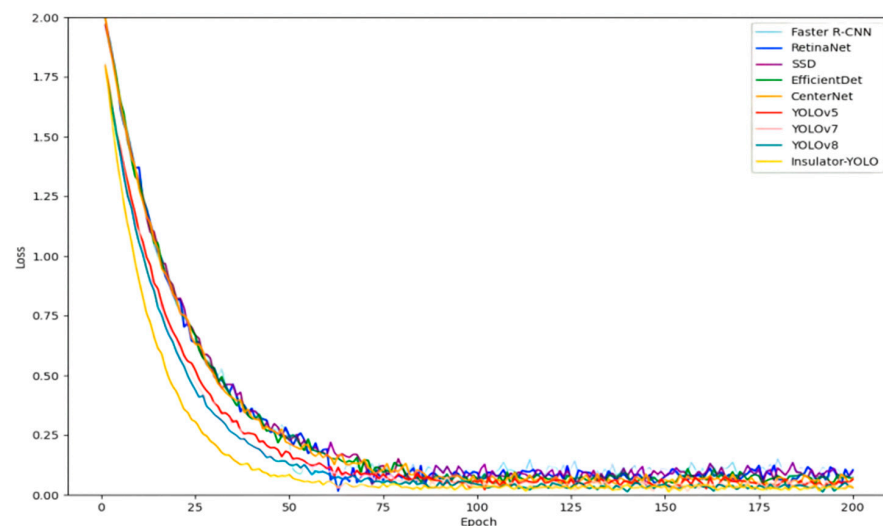


Figure 10. Model loss curve comparison diagram.

Each model was evaluated based on key performance metrics: precision (P%), recall (R%), mean average precision (mAP%), frames per second (FPS), model weight (MB), GFLOPs (G), and inference time (ms). Table 5 shows the performance of the different models in the test dataset and introduces YOLOv8-I, which implements similar enhancements as those in Insulator-YOLO.

The results indicate that the Insulator-YOLO model outperformed all other compared models in precision, recall, and mAP. Specifically, compared to the original YOLOv5, Insulator-YOLO improved the precision rate from 84.19% to 87.92%, the recall rate from 88.02% to 90.11%, and the mAP from 86.45% to 89.65%. These enhancements underscore the effectiveness of the modified YOLOv5 model in accurately detecting faulty targets, particularly in complex scenarios involving intricate backgrounds and small objects. Such improvements ensure higher accuracy and reliability in real-world applications. When comparing Insulator-YOLO to YOLOv8, while YOLOv8 achieved a slightly higher FPS of 72.56, Insulator-YOLO showed advantages in both precision and recall, with increases of 1.18% and 0.88%, respectively. Additionally, Insulator-YOLO's mAP improved by 1.20%. This indicates that Insulator-YOLO optimizes the detection capabilities for fault targets

without significantly increasing the computational overhead, making it particularly suitable for high-precision tasks in insulator fault detection. The introduction of YOLOv8-I, which implements similar modifications as Insulator-YOLO, highlights a decrease in performance relative to YOLOv8. YOLOv8-I achieved a precision of 86.00% and a recall of 88.50%, both lower than those of the original YOLOv8, suggesting that the improvements may not have effectively enhanced its overall detection capability. The mAP for YOLOv8-I stood at 87.75%, further illustrating that, while some advancements were made, they did not match the effectiveness of Insulator-YOLO. In contrast, traditional two-stage detection algorithms like faster R-CNN exhibit significant limitations in both accuracy and real-time performance. With an mAP of only 75.30% and an FPS of 5.20, faster R-CNN struggled to meet the demands of practical applications requiring real-time detection. Similarly, while SSD demonstrated a higher FPS of 59.67, its accuracy (73.82%) and mAP (76.47%) fell short compared to the YOLO series models, especially Insulator-YOLO.

Table 5. Correlation algorithm comparison.

Model	P%	R%	mAP (%)	FPS	Weight (MB)	GFLOPs (G)	Inference Time (ms)
Faster R-CNN	72.53	78.04	75.30	5.20	130.00	180.12	192.31
RetinaNet	74.23	80.51	77.84	14.93	140.00	120.21	66.85
SSD	73.82	79.50	76.47	59.67	87.00	56.10	16.74
EfficientDet	75.95	80.02	78.91	33.41	21.50	25.37	29.94
CenterNet	77.21	80.98	79.53	30.24	43.00	33.66	33.12
YOLOv5	84.19	88.02	86.45	60.13	38.00	20.00	16.64
YOLOv7	85.63	88.91	87.32	66.24	50.20	25.25	15.11
YOLOv8	86.74	89.23	88.45	72.56	52.30	27.37	13.78
YOLOv8-I	86.00	88.50	87.75	70.00	52.50	30.00	14.29
Insulator-YOLO	87.92	90.11	89.65	61.24	38.50	24.60	16.34

Overall, the Insulator-YOLO model achieved an optimal balance between detection accuracy and recall capability, making it particularly suitable for scenarios that demand high-precision identification and real-time processing, such as insulator fault detection. Although the FPS was not as good as that of YOLOv8, it maintained better real-time and model lightweight characteristics while guaranteeing accuracy, which provides strong support for practical applications. The visualization results are shown in Figure 11.



Figure 11. Partial visualization results.

4. Conclusions

This paper proposes the Insulator-YOLO model for insulator fault detection. The integration of GhostNetV2 as the backbone network decreases computational complexity while improving inference speed. The SE module improves the model's adaptive feature extraction capabilities and refines channel attention allocation. Additionally, the CBAM attention mechanism enhances focus on key regions in both channel and spatial dimensions, significantly boosting small-target detection accuracy. The BiFPN is applied to enhance multiscale feature fusion, and the integration of CIoU and NWD loss functions optimizes bounding box regression, achieving higher accuracy. Additionally, focal loss mitigates the imbalance between positive and negative samples, leading to more accurate and robust defect detection.

The Insulator-YOLO model surpassed other popular models, such as faster R-CNN, RetinaNet, SSD, EfficientDet, CenterNet, YOLOv7, and YOLOv8, in precision, recall, and mAP. These findings validate the model's effectiveness for insulator fault detection tasks, particularly in small-target and complex background scenarios, while maintaining high accuracy, real-time performance, and computational efficiency.

This study not only proposes an efficient fault detection model but also provides technical support for automated detection in transmission line inspection. Future research can further extend the dataset to cover more fault types and environmental conditions and integrate a more prospective deep learning architecture into the model to improve the classification ability of fault types. With these improvements, the model is expected to gain wider generalization in practical applications and solve more complex problems in power systems.

Author Contributions: Conceptualization, N.Z.; methodology, N.Z. and J.S.; software, J.S. and Y.Z.; validation, N.Z. and Y.Z.; formal analysis, N.Z.; investigation, N.Z. and J.S.; writing—original draft, Y.Z.; writing—review and editing, N.Z., J.S. and H.C.; visualization, Y.Z.; supervision, H.C.; project administration, J.S. and H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

Conflicts of Interest: Authors Nan Zhang and Jingyi Su were employed by the Artificial Intelligence Department NARI Group Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Nomenclature

YOLOv5	You only look once version 5
GhostNetV2	GhostNet version 2
SE	Squeeze and excitation
CBAM	Convolutional block attention module
BiFPN	Bi-directional feature pyramid network
CIoU	Complete intersection over union
NWD	Normalized Wasserstein distance
UAV	Unmanned aerial vehicle
CV	Computer vision
mAP	Mean average precision
CBS	Convolutional block with squeeze and excitation
SPPF	Spatial pyramid pooling
FPN	Feature pyramid network
PAN	Path aggregation network
NMS	Non-maximum suppression
CAM	Class activation map

SAM	Spatial attention module
PANet	Path aggregation network
GFLOPs	Giga floating-point operations
FPS	Frames per second
C3_X	C3 block with additional convolutional layers

References

- Wang, J.; Xiong, Z.; Zhang, T.; Ouyang, J. Short-term multiple fault risk assessment for power systems considering unexpected actions of protection system under weather disasters. *Int. J. Electr. Power Energy Syst.* **2024**, *162*, 110254. [CrossRef]
- Ahmed, M.F.; Mohanta, J.C.; Sanyal, A. Inspection and identification of transmission line insulator breakdown based on deep learning using aerial images. *Electr. Power Syst. Res.* **2022**, *211*, 108199. [CrossRef]
- Guan, H.; Sun, X.; Su, Y.; Hu, T.; Wang, H.; Wang, H.; Peng, C.; Guo, Q. UAV-lidar aids automatic intelligent powerline inspection. *Int. J. Electr. Power Energy Syst.* **2021**, *130*, 106987. [CrossRef]
- Zhang, Y.; Lv, C.; Wang, D.; Mao, W.; Li, J. A novel image detection method for internal cracks in corn seeds in an industrial inspection line. *Comput. Electron. Agric.* **2022**, *197*, 106930. [CrossRef]
- Witek, M. Structural integrity of steel pipeline with clusters of corrosion defects. *Materials* **2021**, *14*, 852. [CrossRef] [PubMed]
- Ren, S.; He, K.; Girshick, R.B.; Sun, J. R-FCN: Object Detection via Region-based Fully Convolutional Networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016.
- Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; Volume 28.
- Wu, X.; Duan, J.; Yang, L.; Duan, S. Intelligent cotter pins defect detection for electrified railway based on improved Faster R-CNN and dilated convolution. *Comput. Ind.* **2024**, *162*, 104146. [CrossRef]
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
- Cheng, D.-J.; Wang, S.; Zhang, H.-B.; Sun, Z.-Y. A novel framework for low-contrast and random multi-scale blade casting defect detection by an adaptive global dynamic detection transformer. *Comput. Ind.* **2024**, *162*, 104138. [CrossRef]
- Zhou, X.; Ren, Z.; Zhang, Y.; Mi, T.; Zhou, S.; Jiang, Z. A shunted-swin transformer for surface defect detection in roller bearings. *Measurement* **2024**, *238*, 115283. [CrossRef]
- Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot Multibox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
- Redmon, J.; Divvala, S.; Girshick, R.B.; Farhadi, A. You Only Look Once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
- Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
- ultralytics/yolov5: v7.0-YOLOv5 SOTA Realtime Instance Segmentation. Available online: <https://github.com/ultralytics/yolov5> (accessed on 10 August 2024).
- Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.
- Wang, Y.; Huang, J.; Dipu, M.S.K.; Zhao, H.; Gao, S.; Zhang, H.; Lv, P. YOLO-RLC: An advanced target-detection algorithm for surface defects of printed circuit boards based on YOLOv5. *Comput. Mater. Contin.* **2024**, *80*, 4973–4995. [CrossRef]
- Tao, T.; Dong, D.; Huang, S.; Chen, W. Gap detection of switch machines in complex environment based on object detection and image processing. *J. Transp. Eng. Part A Syst.* **2020**, *146*, 04020083. [CrossRef]
- Kumar, S.S.; Wang, M.; Abraham, D.M.; Jahanshahi, M.R.; Iseley, T.; Cheng, J.C. Deep learning-based automated detection of sewer defects in CCTV videos. *J. Comput. Civ. Eng.* **2020**, *34*, 04019047. [CrossRef]
- Lanhang, C. Defect identification of electricity transmission line insulators based on the improved lightweight network model with computer vision assistance. *Heliyon* **2024**, *10*, e30405. [CrossRef] [PubMed]
- Jiang, P.; Ergu, D.; Liu, F.; Cai, Y.; Ma, B. A review of YOLO algorithm developments. *Procedia Comput. Sci.* **2022**, *199*, 1066–1073. [CrossRef]
- Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
- Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
- Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1580–1589. [CrossRef]
- Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141. [CrossRef]

26. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19. [[CrossRef](#)]
27. Wang, J.; Xu, C.; Yang, W.; Yu, L. A normalized Gaussian Wasserstein distance for tiny object detection. *arXiv* **2023**, arXiv:2303.11358.
28. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO series in 2021. *arXiv* **2021**, arXiv:2107.08430.
29. Li, X.; Yu, Y.; Zhao, Z.; Huang, X.; Zhang, Z.; Zhang, S. DS-YOLOv5s: Lightweight detection algorithm for foreign objects on the surface of power transmission equipment based on multi-scale features. *IEEE Access* **2022**, *10*, 108315–108327.
30. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.