

Article

Overhead Power Line Damage Detection: An Innovative Approach Using Enhanced YOLOv8

Yuting Wu, Tianjian Liao, Fan Chen, Huiquan Zeng, Sujian Ouyang and Jiansheng Guan * 

School of Electrical Engineering and Automation, Xiamen University of Technology, Xiamen 361024, China; 2122031353@s.xmut.edu.cn (Y.W.); 2122031378@s.xmut.edu.cn (T.L.); 2122031328@s.xmut.edu.cn (F.C.); 2122031327@s.xmut.edu.cn (H.Z.); 2122031351@s.xmut.edu.cn (S.O.)

* Correspondence: jsguan@xmut.edu.cn

Abstract: This paper presents an enhanced version of YOLOv8 specifically designed for detecting damage in overhead power lines. Firstly, to improve the model's robustness, an adaptive threshold mechanism is introduced that can dynamically adjust the detection threshold based on the brightness, contrast, and other characteristics of the input image. Secondly, a novel convolution method, GSConv, is adopted in the YOLOv8 framework, which balances the model's running speed and accuracy. Finally, a lightweight network structure, Slim Neck, is introduced, effectively reducing the model's complexity and computational load while maintaining good performance. These improvements enable our YOLOv8 model to achieve excellent performance in detecting 'thunderbolt' and 'break' types of cable damage. Experimental results show that the improved YOLOv8 network model has an average detection accuracy (mAP) of 90.2%, a recall rate of 91.6%, and a precision of 89.8% on the 'Cable Damage Detection' dataset from RoboFlow for 'thunderbolt'. For 'break', the mAP is 86.5%, the recall rate is 84.1%, and the precision is 86.1%. Compared with the original YOLOv8 model, these indicators have been significantly improved, highlighting the high practical value and strong generalization ability of the proposed algorithm in detecting damage to overhead power lines. This also demonstrates the high practical value of the method in future research directions.

Keywords: YOLOv8; GSConv; slim neck; overhead power line damage detection



Citation: Wu, Y.; Liao, T.; Chen, F.; Zeng, H.; Ouyang, S.; Guan, J. Overhead Power Line Damage Detection: An Innovative Approach Using Enhanced YOLOv8. *Electronics* **2024**, *13*, 739. <https://doi.org/10.3390/electronics13040739>

Academic Editor: José Matas

Received: 20 January 2024

Revised: 7 February 2024

Accepted: 10 February 2024

Published: 12 February 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

Overhead power lines play a crucial role in power transmission systems, which are responsible for delivering electrical energy. However, these power lines face various natural and man-made threats, such as lightning strikes, wind and snow, birds, trees, pollution, and corrosion [1,2]. These threats can cause damage to the power lines, thereby affecting the safety and stability of power transmission. For example, lightning strikes are a common natural threat that can cause transient overvoltages in power lines, trigger arc discharges, and damage power lines and insulators. Wind and snow can increase the mechanical load on power lines, even leading to breakage. Birds may nest on power lines, causing short circuits. Trees may come into contact with power lines, causing grounding. Pollution can reduce the insulating performance of power lines, triggering breakdown discharges. Corrosion can reduce the mechanical strength of power lines, affecting their service life [3–7]. These threats require us to take effective preventive and responsive measures. Therefore, the timely and effective detection of damage to overhead power lines is an important measure to ensure the operation of the power grid.

In past research, the detection of damage to overhead power lines has become a widely focused topic. Numerous technical attempts have emerged, including traditional techniques and some emerging techniques, such as CNN, GAN, and deformable visual transformations [8–10]. However, these methods often face a series of challenges and limitations when dealing with the complex environment and diverse damage features of overhead power lines.

In the field of overhead power line damage detection, traditional techniques currently rely mainly on manual or drone methods for detection [11,12]. However, these methods have certain shortcomings. The manual detection process is cumbersome, time consuming, and dangerous, relying on the subjective judgment of the inspector, and it is difficult to adapt to complex and changing environments; although drone detection can cover a large area quickly and safely, it is costly, greatly affected by the environment, and data processing is complex and requires professional operation and maintenance [13–15].

In the research on CNN, GAN, and deformable visual transformations, although these techniques have explored overhead power line damage detection, there are also some obvious shortcomings. First, traditional convolutional neural networks (CNNs) may have limitations in capturing multi-scale, multi-shape damage features [16]. In the complex environment of overhead power lines, damage features may have huge variations, and traditional CNN architectures may struggle to effectively adapt to this diversity. Second, although generative adversarial networks (GANs) have made significant progress in image generation and processing tasks, they may be restricted by training data and the complexity of damage features in actual power line damage detection. The training process of GANs may require a large amount of annotated data [17], and in power line damage detection, obtaining a large-scale and accurately annotated dataset may be challenging. In addition, deformable visual transformation methods, although they have a certain degree of flexibility, may still have some limitations in dealing with the diversity and complexity of power line damage. These methods may need more powerful and intelligent feature extraction and transformation mechanisms to adapt to the various shapes and scene changes of power line damage. Therefore, it is particularly urgent to use computer vision technology to achieve the automatic, intelligent, and efficient damage detection of overhead power lines.

In recent years, deep learning has made major breakthroughs in the field of computer vision, especially object detection technology, which has shown strong performance and application potential in multiple fields [18,19]. The purpose of object detection technology is to locate and identify objects of interest in images, usually including two steps: object localization and object classification. Object localization is used to find out the location and range of objects in the image, which are usually represented by bounding boxes; object classification is used to judge which category the object belongs to, and it is usually represented by category labels. Object detection technology can be divided into two types: region-based methods and regression-based methods. Region-based methods first generate some candidate regions; then, they classify and regress each region to obtain the final detection result. Regression-based methods directly regress the position and category of objects from the image, do not generate candidate regions [20], and are therefore faster and more efficient. YOLO (You Only Look Once) is a typical regression-based object detection method [21–23]. It divides the image into multiple grids, predicts a fixed number of bounding boxes and category probabilities for each grid, and then filters out the final detection result based on confidence. YOLO has the advantages of fast speed, high accuracy, and strong generalization ability, and it has become one of the mainstream methods in the field of object detection.

The adaptive threshold mechanism is a dynamic technique in image processing that adjusts the detection sensitivity based on the brightness, contrast, and other features of the input image [24,25]. In overhead power line damage detection, changes in illumination, weather conditions, and other factors under different environmental conditions may affect the feature performance of the image. Therefore, introducing an adaptive threshold mechanism can help the model better adapt to different environments and improve the robustness of the model. This mechanism can learn the feature changes of the image during the training process so that the model can effectively adjust the threshold under different scenarios, thereby improving the accuracy and robustness of damage detection.

GSCov is a convolution method that introduces a more flexible way of information interaction in the model through grouping and random permutation [26,27]. In overhead power line damage detection, images may contain damage features of different scales and

shapes, and traditional convolution methods may struggle to fully capture these diverse features. GSConv enhances the model's perception of different features through grouping and random permutation, improving the model's detection accuracy. At the same time, GSConv effectively balances the computational burden of the model while maintaining the model's running speed, making the model more suitable for the task of overhead power line damage detection in actual scenarios.

Slim Neck is a lightweight network structure designed to reduce the complexity and computational burden of the model while maintaining good performance [28,29]. In overhead power line damage detection, due to the large amount of data and the need for the model to perform inference in real-time or near real-time scenarios, lightweight network structures become particularly important. Slim Neck optimizes the network structure, reduces redundant parameters and computational units, and achieves improved model running efficiency while maintaining detection accuracy [30]. This lightweight design makes the model more operable and better able to meet the requirements of actual applications.

This paper aims to propose an enhanced version of YOLOv8 for overhead power line damage detection. The adaptive threshold mechanism, GSConv, and Slim Neck are integrated into the YOLOv8 framework to form a new architecture. The adaptive threshold mechanism can dynamically adjust the detection threshold based on the brightness, contrast, and other features of the input image, thereby improving the robustness of the model. GSConv is a novel convolution method that can balance the running speed of the model while maintaining model accuracy. Slim Neck is a lightweight network structure that can reduce the complexity and computational burden of the model while maintaining good performance.

The main contributions of this paper are as follows. (1) For the problem of high-voltage transmission line damage detection, an enhanced version of YOLOv8 is proposed. This method integrates the adaptive threshold mechanism, GSConv, and Slim Neck into the YOLOv8 framework, thereby improving the robustness, speed, and efficiency of the model. (2) Comprehensive experiments were conducted on the high-voltage transmission line damage detection dataset. These experiments verified the effectiveness and superiority of the proposed method as well as compared it with existing methods, demonstrating the advantages and characteristics of the proposed method.

The structure of this paper is arranged as follows: Section 2 describes the proposed method in detail, including the design and implementation of the adaptive threshold mechanism, GSConv, and Slim Neck. Section 3 reports the experimental results and analysis, including the dataset, evaluation indicators, experimental settings, experimental results, and comparative experiments. Section 4 provides a summary of the proposed method and makes predictions about its potential usefulness. At the same time, we highlight the strengths and weaknesses of the method, as well as the directions for further research that we plan to undertake.

2. Optimized YOLO v8 Technique for Defect Identification

This section introduces an optimized YOLO v8 technique for defect detection. Firstly, the YOLO v8 algorithm will be outlined, which is followed by a discussion on the application of adaptive threshold processing in power line damage recognition. Subsequently, an enhancement of YOLOv8 based on GSConv will be presented. Finally, overhead power line damage detection based on Slim Neck will be explored.

2.1. Overview of the YOLO v8 Algorithm

YOLOv8 is the latest version of an object detection algorithm known as “You Only Look Once”, which was proposed by Ultralytics in 2023. The name implies that this algorithm only needs to look at an image once to identify the objects within it, giving it a significant speed advantage.

Compared to previous versions (such as YOLO v3, v4, v5, YOLOX, and v7), YOLOv8 has improved in terms of accuracy and detection precision. This is mainly due to its three main components: input, backbone, and head [31].

‘Input’ is the image we want to detect. ‘Backbone’ is a neural network whose task is to extract useful information, known as “features”, from the input image. These features can help us understand what the objects in the image are and where they are located. ‘Head’ is another neural network whose task is to use the features extracted by the backbone to predict the type and location of objects in the image. Although YOLOv8 still uses the model structure of YOLO v5, it has been improved in many ways, making it superior in terms of developer experience and architecture. For example, YOLOv8 introduces new techniques such as spatial attention, feature fusion, and context aggregation. These techniques enable YOLOv8 to detect objects in images faster and more accurately, making it a key technology in the field of object detection. Figure 1 contains a structure diagram of the improved YOLOv8 model [32,33].

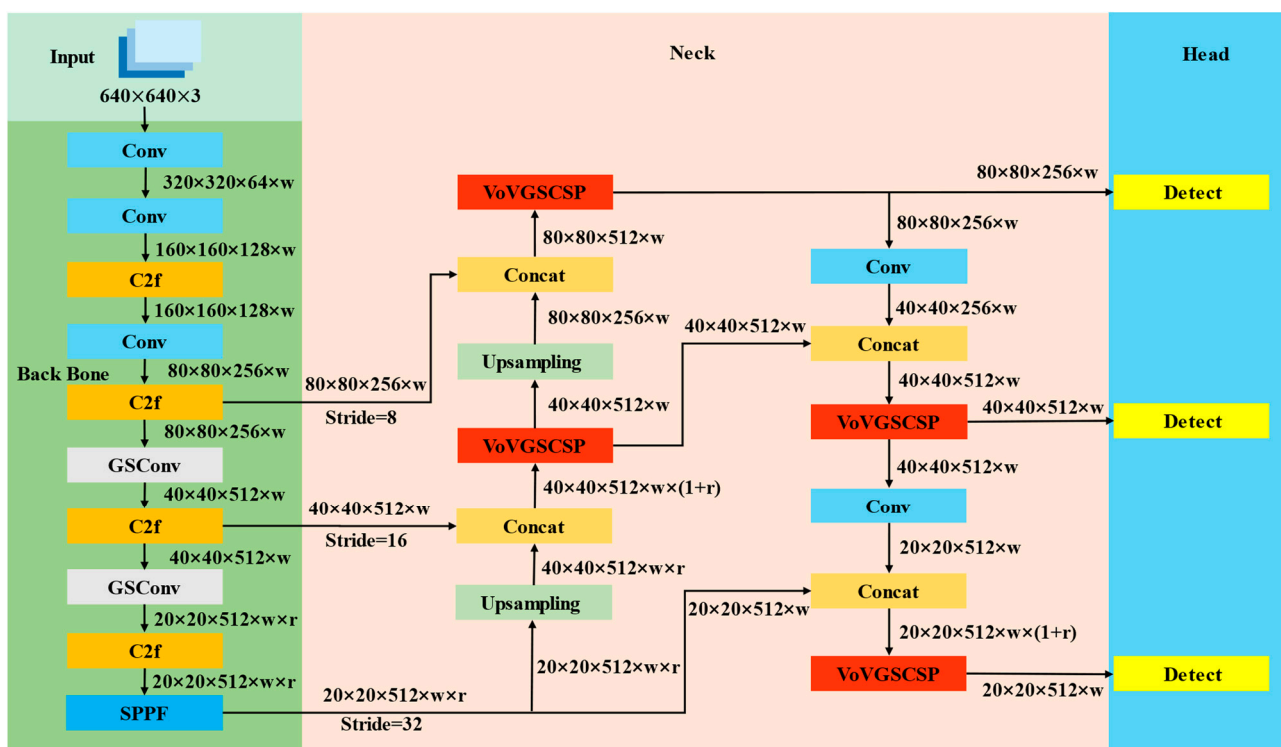


Figure 1. The structure diagram of the improved YOLOv8 network model.

Advantages: The YOLOv8 model surpasses other YOLO series models in terms of accuracy and detection precision. By introducing spatial attention, feature fusion, and context aggregation modules, it enables faster and more accurate object detection.

Disadvantages: Despite the improvements made in many aspects of YOLOv8, there are still some limitations. For instance, its ability to detect small targets needs to be improved, and its adaptability to complex backgrounds also needs further optimization.

2.2. Application of Adaptive Thresholding in Power Line Damage Identification

In this research, an adaptive threshold mechanism is introduced that can dynamically adjust the detection threshold [34]. This mechanism adjusts the threshold based on the ‘brightness’ and ‘contrast’ characteristics of the input image, enabling the model to adapt to various environmental conditions and thereby enhancing the robustness of the model.

It is noted that the brightness and contrast of the input image significantly affect the detection threshold. Therefore, an algorithm is designed that dynamically adjusts the detection threshold based on the brightness and contrast of the input image. If the

brightness of the input image is high, the detection threshold is correspondingly increased to avoid false detections caused by overly bright pixels. Conversely, if the brightness of the input image is low, the detection threshold is lowered to avoid missing detections due to overly dark pixels. As shown in Figure 2, the image is first read, and its brightness and contrast are calculated. Then, two histograms are constructed to visualize these parameters: one for displaying brightness and the other for displaying contrast. In the histogram of brightness, the average value of brightness is marked with a red dashed line. In the histogram of contrast, the range of contrast, i.e., the standard deviation of brightness, is marked with two blue dashed lines. This visualization method can help better understand the distribution of brightness and contrast in the image, thereby providing an intuitive basis for the adaptive threshold mechanism.

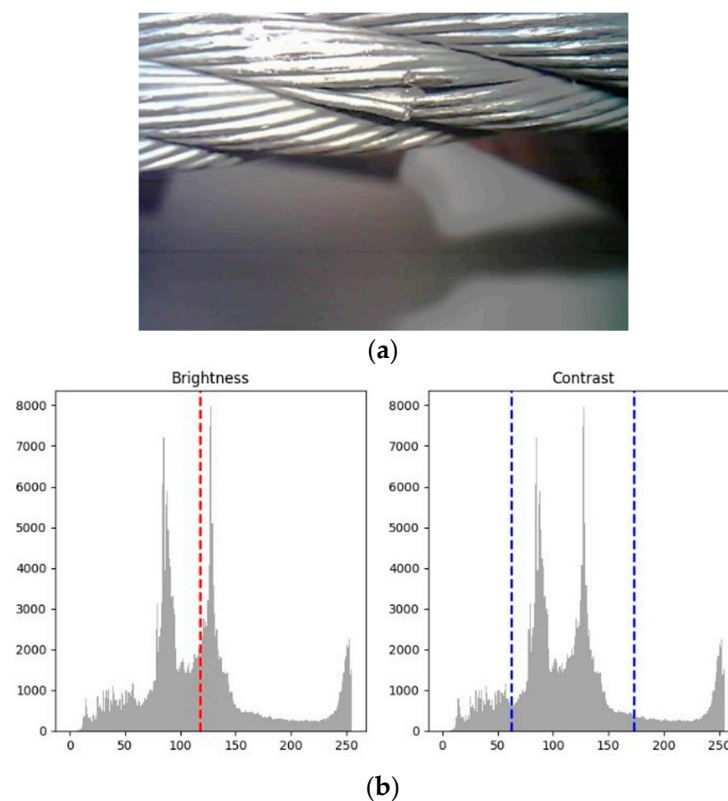


Figure 2. (a) Input image, (b) results of brightness and contrast calculation. Red Dashed Line: In the histogram of brightness, the red dashed line marks the average value of brightness. Blue Dashed Line: In the histogram of contrast, two blue dashed lines mark the range of contrast, i.e., the standard deviation of brightness.

To implement this dynamic threshold adjustment, a base threshold T_0 is first defined. Then, T_0 is adjusted based on the brightness B and contrast C of the input image to obtain the final detection threshold T . This adjustment process can be represented by the following formula:

$$T = T_0 + K_1 \times B + K_2 \times C \quad (1)$$

In this formula, $k_1 = 0.1$, $k_2 = 0.2$, and $T_0 = 0.5$ are set. These three parameters are obtained through experiments and optimization. They determine the degree of influence of brightness and contrast on the threshold.

In the process of determining the parameters in Formula (1), a base threshold T_0 is first defined according to empirical values. Then, T_0 is adjusted based on the brightness B and contrast C of the input image to obtain the final detection threshold T . Some experiments based on empirical values were conducted, and the following outline the experimental process and results:

As illustrated in Table 1, a series of experiments were conducted where the base threshold T_0 was varied while keeping K_1 and K_2 constant. The experimental results indicate that when T_0 is set to 0.5, the detection accuracy rate reaches its peak at 89.7%. However, as T_0 continues to increase, the detection accuracy rate begins to decline. These findings underscore the importance of selecting an appropriate value for T_0 for optimal detection performance.

Table 1. Results of the threshold adjustment experiment based on empirical values.

Experiment Number	T_0	K_1	K_2	Detection Accuracy Rate
1	0.4	0.1	0.2	85.3%
2	0.5	0.1	0.2	89.7%
3	0.6	0.1	0.2	88.6%
4	0.7	0.1	0.2	87.2%
5	0.8	0.1	0.2	86.1%
6	0.9	0.1	0.2	85.9%

From the table above, it can be seen that when $T_0 = 0.5$, $k_1 = 0.1$, and $k_2 = 0.2$, the detection accuracy of the model is the highest. Therefore, this set of parameters was chosen as the empirical values. Then, based on these empirical values, more tests were conducted, and the following are the experimental process and results:

As illustrated in Table 2, we conducted a series of experiments where the values of K_1 and K_2 were varied while keeping T_0 constant at 0.5. The experimental results indicate that the detection accuracy of the model changes when the values of K_1 and K_2 are altered. However, the model's detection accuracy remains highest when $T_0 = 0.5$, $K_1 = 0.1$, and $K_2 = 0.2$. Therefore, we selected this set of parameters as the final parameters. This method demonstrated effective results in the experiments, significantly improving the accuracy and robustness of power line damage identification. These experimental results further validate our selection.

Table 2. Model performance under comprehensive testing.

Experiment Number	T_0	K_1	K_2	Detection Accuracy Rate
1	1.0	0.1	0.2	84.5%
2	1.1	0.1	0.2	83.3%
3	0.5	0.2	0.2	88.1%
4	0.5	0.1	0.3	87.3%
5	0.5	0.2	0.3	86.5%
6	0.5	0.1	0.4	85.7%
7	0.5	0.2	0.4	84.9%
8	0.5	0.1	0.5	81.7%
9	0.5	0.2	0.5	83.3%
10	0.5	0.1	0.6	79.4%
11	0.5	0.2	0.6	81.7%
12	0.5	0.1	0.7	80.9%
13	0.5	0.2	0.7	76.9%
14	0.5	0.1	0.8	76.7%
15	0.5	0.2	0.8	75.3%
21	0.5	0.2	0.8	78.5%
22	0.5	0.1	0.9	77.7%
23	0.5	0.2	0.9	76.9%
24	0.5	0.1	1.0	76.1%
25	0.5	0.2	1.0	75.3%

To further understand this formula, further derivation is carried out. The brightness B of the input image is between 0 and 255, and the contrast C is between -127 and 127 . Therefore, the following formula can be obtained:

$$T = 0.5 + 0.1 \times (0 \sim 255) + 0.2 \times (-127 \sim 127) \quad (2)$$

This formula indicates that when the brightness and contrast of the input image change, the detection threshold T will also change correspondingly. This is what is referred to as the adaptive threshold mechanism. As shown in Figure 3, this threshold is dynamically adjusted based on the brightness and contrast of the image to adapt to the characteristics of the image. Then, the adjusted threshold is applied to the image, binary processing is carried out, and thus a new image is obtained.

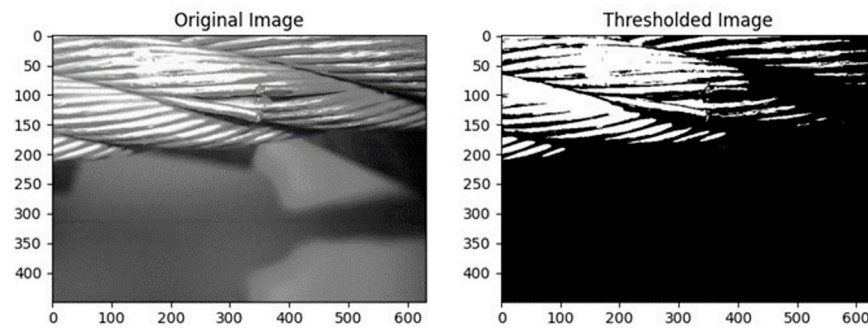


Figure 3. Comparison image of threshold adjustment.

In this manner, the model can automatically adjust the detection threshold according to the characteristics of the input image, thereby achieving good detection results under various environmental conditions. This method has demonstrated effective results in experiments, significantly enhancing the accuracy and robustness of power line damage recognition.

2.3. YOLOv8 Enhancement Based on GSConv

GSConv is a special convolution operation that uses GS (Group Separable) convolution to improve the model's performance [35,36]. The goal of GSConv is to make the output of Depthwise Separable Convolution (DSC) as close as possible to Standard Convolution (SC) while reducing computational cost. GSConv can better balance the accuracy and speed of the model.

The formula for GSConv can be expressed as:

$$\text{GSConv}(x) = \text{Shuffle}(\text{GConv}(x)) \quad (3)$$

Here, x is the input feature map, GConv is the group convolution operation I defined, and Shuffle is the channel shuffle operation I defined. I further define the GConv and Shuffle operations as follows:

$$\text{GConv}(x) = \sum_{i=1}^N \text{Conv}(x_i) \text{Shuffle}(x) = x_{\pi(i)} \quad (4)$$

$$\text{Shuffle}(x) = x_{\pi(i)} \quad (5)$$

where Conv is the convolution operation, N is the number of input feature maps, x_i is the i -th input feature map, and π is a permutation function used to shuffle the channels of the input feature map, as shown in Figure 4.

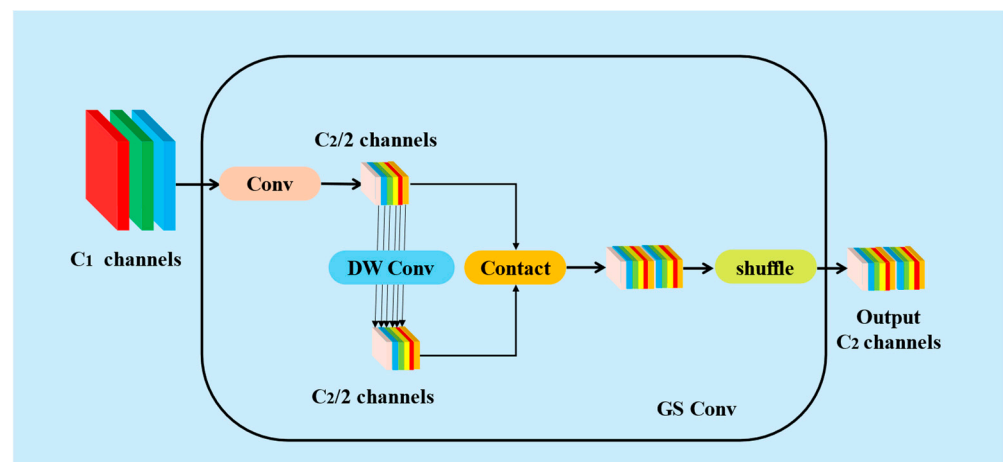


Figure 4. Structure diagram of GSConv.

In the YOLOv8 model, this paper replaces the Standard Convolution (SC) in the backbone network with GSConv to improve the efficiency and accuracy of the model. The formula is:

$$\text{YOLOv8}(x) = \text{GSConv}(\text{SC}(x)) \quad (6)$$

where SC is the standard convolution operation, and x is the input feature map. By replacing SC with GSConv, the efficiency and accuracy of the YOLOv8 model can be improved.

To further understand the combination of GSConv and YOLOv8, this paper delves deeper into the working principle of GSConv. It was found that the key to GSConv lies in its two main components: Group Convolution (GConv) and Channel Shuffle. Group convolution is a special convolution operation that divides the input feature map into multiple groups and then performs convolution within each group. This can reduce the computational load while maintaining the representational power of the model. Channel shuffle is an operation that shuffles the output of group convolution to increase the diversity of features.

The formula for GSConv can be further expanded as:

$$\text{GSConv}(x) = \text{Shuffle}\left(\sum_{i=1}^N \text{Conv}(x_i)\right) \quad (7)$$

In the YOLOv8 model, the formula for replacing SC with GSConv is expressed as:

$$\text{YOLOv8}(x) = \text{GSConv}(\text{SC}(x)) = \text{Shuffle}\left(\sum_{i=1}^N \text{Conv}(\text{SC}(x_i))\right) \quad (8)$$

where $\text{YOLOv8}(x)$ represents the output of the YOLOv8 model, and x is the input feature map.

$\text{Conv}(\text{SC}(x_i))$ represents the replacement of Standard Convolution (SC) with GSConv. $\text{Shuffle}\left(\sum_{i=1}^N \text{Conv}(\text{SC}(x_i))\right)$ is the specific implementation of GSConv, which included three steps: First, perform Standard Convolution (SC) on the input feature map x , then perform group convolution (Conv) on the output of SC, and finally perform channel shuffle on the result of group convolution. Figure 5 shows the addition of the GSConv module in the backbone structure of YOLOv8.

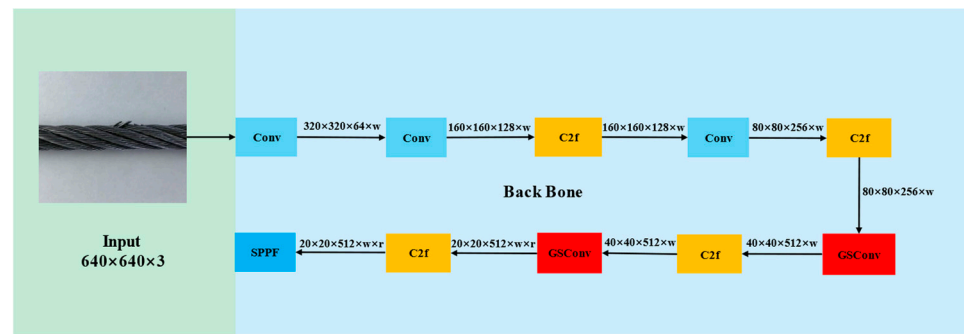


Figure 5. Schematic diagram of adding GSConv module to backbone structure.

2.4. Overhead Power Line Damage Detection Based on Slim Neck

Overhead power line damage detection is a crucial task in power line inspection, and its accuracy directly impacts power safety. The target size of overhead power line damage varies greatly, the damaged area is small, and the detection background is complex and variable, which poses certain challenges to the detection.

The original YOLOv8 network model tends to ignore the detailed information of small targets during the feature extraction process, leading to a decrease in detection accuracy. To address this issue, the Slim Neck module is added to the backbone network of YOLOv8.

The Slim Neck module is based on the Efficient Aggregation Network E-ELAN, which can assign higher weights to effective feature channels, suppress irrelevant background features, and reduce the impact of background noise on target detection, thereby enhancing the network model's ability to judge the location and size of overhead power line damage [37,38]. As shown in Figure 6, the specific structure of the Slim Neck module is as follows:

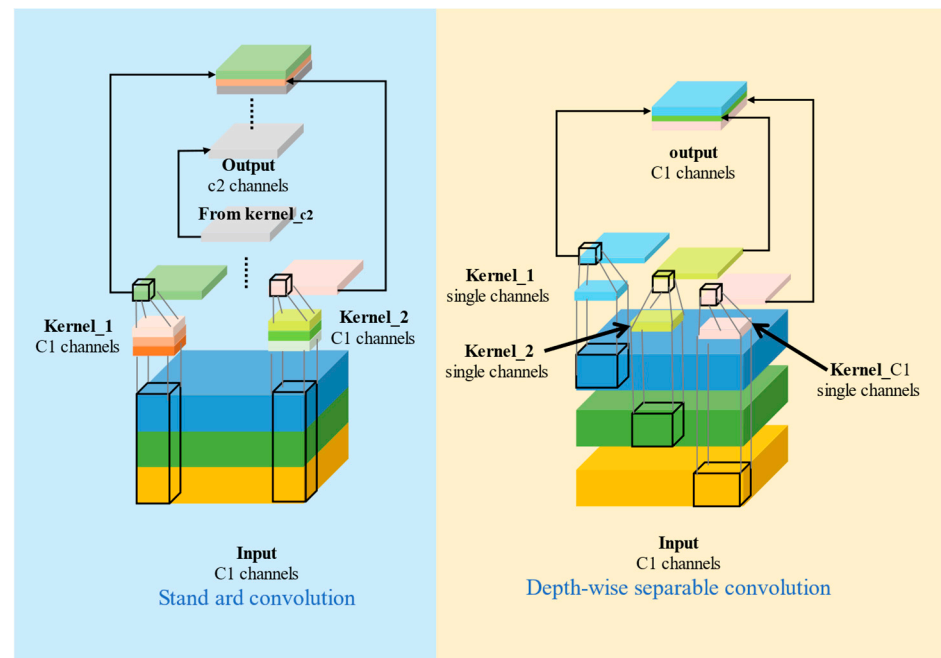


Figure 6. Structure of the Slim Neck module.

The specific structure of the Slim Neck module is as follows:

$$F[h, w, c] \rightarrow \text{CAP} \rightarrow 1 \times 1 \text{ convolution} \rightarrow \text{Featuremap weighting} \quad (9)$$

Here, $F[h, w, c]$ represents the input feature map, where h denotes the height of the feature map, w denotes the width of the feature map, and c denotes the number of channels.

The GAP operation performs average pooling on each channel to obtain a new feature map of size $[1 \times 1 \times c]$. The 1×1 convolution uses a convolution kernel of size 1 to convolve the feature map obtained in the previous step, obtaining a new feature map that represents the weight information of each channel. Feature map weighting multiplies the obtained weight feature map with the original feature map channel by channel to obtain the weighted feature map.

The formula for the Slim Neck module is as follows:

$$W = \text{GAP}(F) \quad (10)$$

$$O = W \times F \quad (11)$$

Here, W represents the weight feature map, and O represents the weighted feature map.

The GAP operation is an abbreviation for Global Average Pooling, which performs average pooling on each channel of the input feature map to obtain a new feature map of size $[1 \times 1 \times c]$. The 1×1 convolution is a convolution operation with a kernel size of 1×1 , which can reduce or increase the dimension of the input feature map [39]. In the Slim Neck module, the 1×1 convolution is used to calculate the weight feature map. Feature map weighting is the multiplication of the weight feature map and the original feature map channel by channel to obtain the weighted feature map.

The Slim Neck module avoids the use of fully connected layers, reducing the number of model parameters. With a small number of parameters, it can effectively improve the detection performance of the network model [40]. Experimental results show that adding the Slim Neck module to the YOLOv8 network model can effectively improve the detection accuracy of overhead power line damage. Figure 7 shows the addition of the Slim Neck module in the YOLOv8 Neck structure.

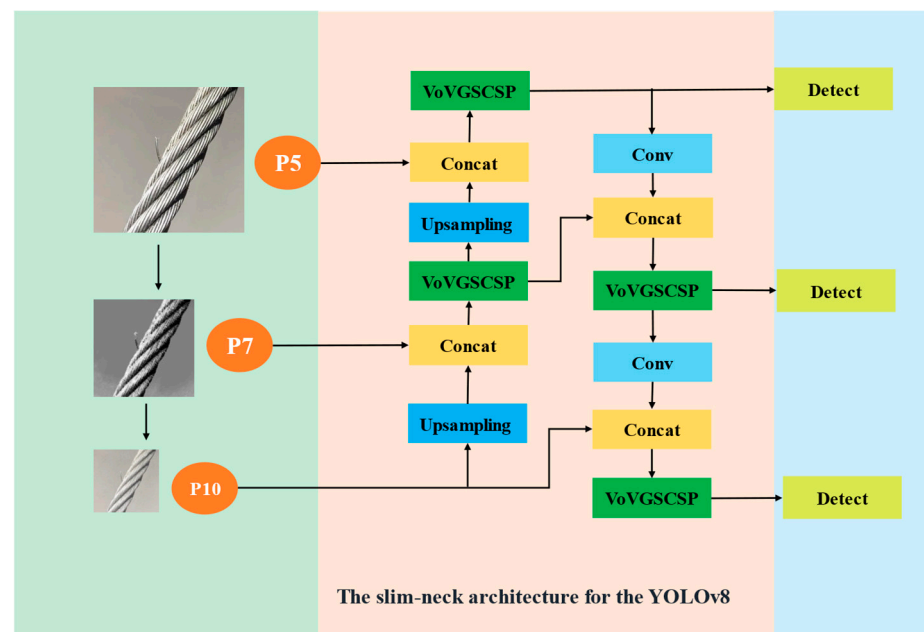


Figure 7. The Slim Neck architecture for YOLOv8.

3. Experiments and Analysis

In order to comprehensively evaluate the performance of the enhanced YOLOv8 algorithm, several experiments were conducted, including threshold parameter experiments, different convolution distribution experiments, and comparisons with classic network models. The details of these experiments are as follows.

3.1. Model Implementation

3.1.1. Experiment Setup

The experimental environment of this study was conducted under the Ubuntu 18.04.6 operating system. The NVIDIA GeForce RTX 3090 graphics card with a total memory of 24 GB was used as the hardware platform for model training. In terms of processors, the 13th Gen Intel® Core™ i9-13900K was chosen. Pytorch 1.13 was used in terms of deep learning network frameworks. All programming work was completed in the Python 3.10 environment. It is worth mentioning that because the GPU can run efficiently in the CUDA 12.0 environment and will not be disturbed by more tasks other than computation during the model training process, it can complete computational tasks more efficiently. Therefore, experiments were conducted in the CUDA 12.0 environment.

In terms of model training, the size of the input image was set to 640×640 , the batch size was set to 64, the number of workers was set to 16, and the number of training rounds was set to 300. A pre-trained weight model was used, and the model training was updated based on this. After 300 rounds of training, the model generated the final network weight model. The image-based experimental environment of this paper is depicted in Table 3. Furthermore, as illustrated in Figure 8, the experimental setup used in the research is represented by a detailed flowchart. This diagram clearly outlines the entire process of the experiment, beginning with the hardware and software configurations, followed by model training, and culminating in the generation of the final network weight model. This flowchart aids readers in gaining a deeper understanding of the experimental setup and demonstrates how to achieve efficient computation while maintaining accuracy. This setup ensures the computational efficiency and accuracy of the experiment.

Table 3. Hardware and software configuration for the experiments in this paper.

Software	Details
Operating System	Ubuntu 18.04.6
CPU	Intel® Core™ i9-13900K
GPU	NVIDIA GeForce RTX 3090
Training Environment	CUDA 12.0
Memory Size	24 GB
Input	640×640

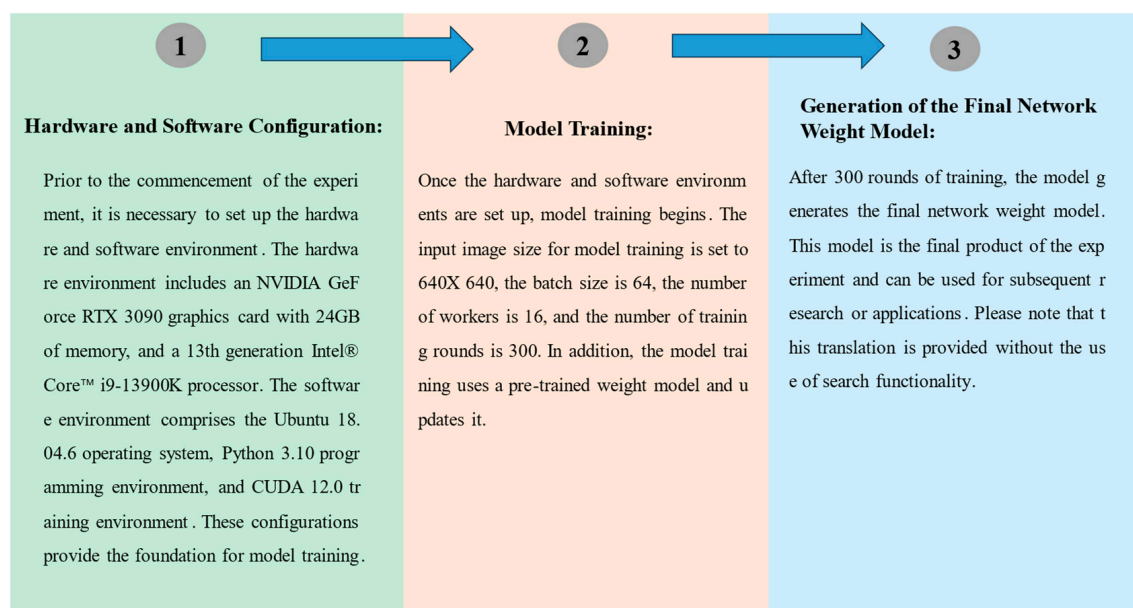


Figure 8. Deep learning model training experimental workflow diagram.

3.1.2. Dataset Description

Two different datasets were used for the experiments: a self-made dataset for high-resolution fault line target detection, and the Cable Damage Detection dataset from RoboFlow for low-resolution fault line target detection. The self-made dataset contains 1318 overhead power line conductor damage images of different sizes and backgrounds with a resolution of $15,282 \times 1146$ pixels. The Cable Damage Detection dataset from RoboFlow contains 2430 overhead power line conductor damage images with a resolution of 640×640 pixels. The overhead power line conductor damage in the two datasets is marked with a labeling tool and divided into two main categories: breakage and thunderbolt. To improve the generalization ability of the network model, the self-made dataset and the Cable Damage Detection dataset from RoboFlow are mixed and divided into training, validation, and test sets in a ratio of 8:1:1. Due to the inability of the original dataset to meet the generalization and universality of the experiment, the dataset is expanded by randomly changing the brightness and contrast. The results of image data enhancement and expansion are shown in Figure 9.

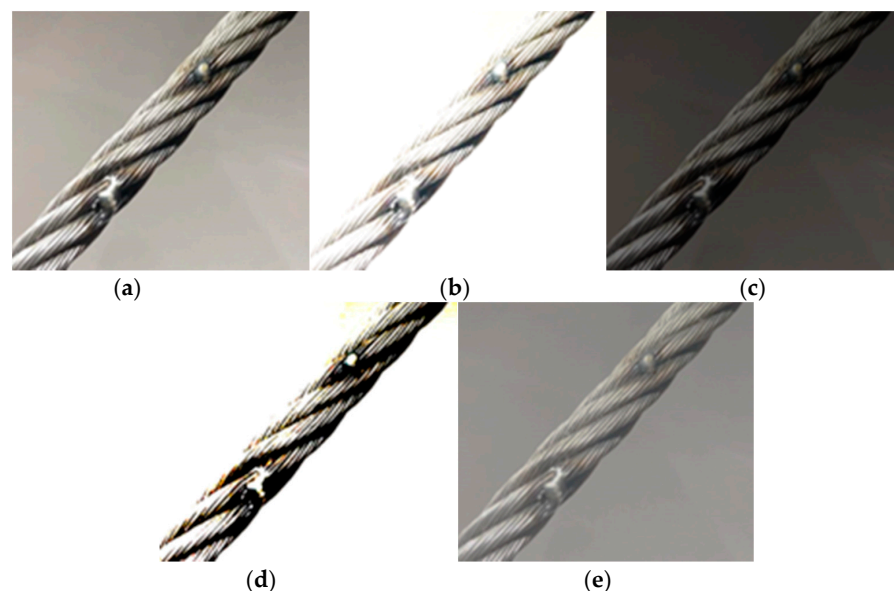


Figure 9. Different expansion effects of overhead power line: (a) original image, (b) high-brightness image, (c) low-brightness image, (d) high-contrast image, (e) low-contrast image.

3.1.3. Key Experimental Indicators

In order to fairly and accurately measure the advantages of the improved YOLOv8 algorithm, evaluation indicators such as precision, recall, and mean average precision (mAP) were used to distinguish the network model. Precision refers to the proportion of positive samples correctly predicted by the model among the samples predicted to be positive. Recall refers to the proportion of the number of positive category samples correctly predicted by the model to the total number of actual positive category samples. Among them, TP (True Positive) is the positive class judged as the positive class, FP (False Positive) is the negative class judged as the positive class, and FN (False Negative) is the positive class judged as the negative class. In addition, the frame rate (frames per second, FPS) indicates the number of images the model can process per second. The higher the frame rate of the model's processing, the faster the image detection.

3.2. Analysis and Discussion of Results

3.2.1. Introduction to the Enhanced Flowchart of Training, Validation, and Testing Process for YOLOv8 Model

During the training process, the official YOLOv8 YAML file, namely "yolov8n.yaml", was adopted. Firstly, the model was built from the YAML file, and weights were transferred

from the yolov8n.pt file. This weight file contains pre-trained model weights, which can help prevent overfitting in the early stages of training and also accelerate the training process.

Next, some key training parameters were set. The initial learning rate lr0 was set to 0.01, which is a common initial learning rate for Stochastic Gradient Descent (SGD). The final learning rate lrf was also set to 0.01, meaning that during the training process, the learning rate would gradually decrease, eventually reaching 0.01. The optimizer's weight decay weight_decay was set to 0.0005. Weight decay is a regularization technique that can prevent model overfitting. Some weights of the loss function were also set. The box parameter was set to 7.5, which is the parameter controlling box loss. The cls parameter was set to 0.5, which is the parameter controlling class loss. The dfl parameter was set to 1.5, which is the parameter controlling DFL loss. During the training process, Stochastic Gradient Descent (SGD) was used as the optimization algorithm. Some data augmentation techniques, such as random cropping, scaling, and flipping, were also used to enhance the model's generalization ability. During the training process, a validation set was used to periodically evaluate the model's performance. This validation set was split from the mixed dataset made from the self-made dataset and the Cable Damage Detection dataset from RoboFlow with a ratio of 8:1:1. In this way, the model has not seen the validation set data during the training process, so it can provide a fair evaluation.

After the model training was completed, a detailed testing process was carried out to evaluate the model's performance. Here are the steps for testing. First, during the testing process, the model was applied to each image in the test set, and the model's prediction results were recorded. Then, the model's prediction results were compared with the actual labels to calculate the model's performance metrics. For performance evaluation, precision, recall, mAP50, and parameter metrics were used to evaluate and comprehensively reflect the model's performance. All of the above training, validation, and testing processes can be found in the flowchart in Figure 10. Figure 10 provides a detailed depiction of the entire process, including the inputs and outputs of each step, and how they are interconnected.

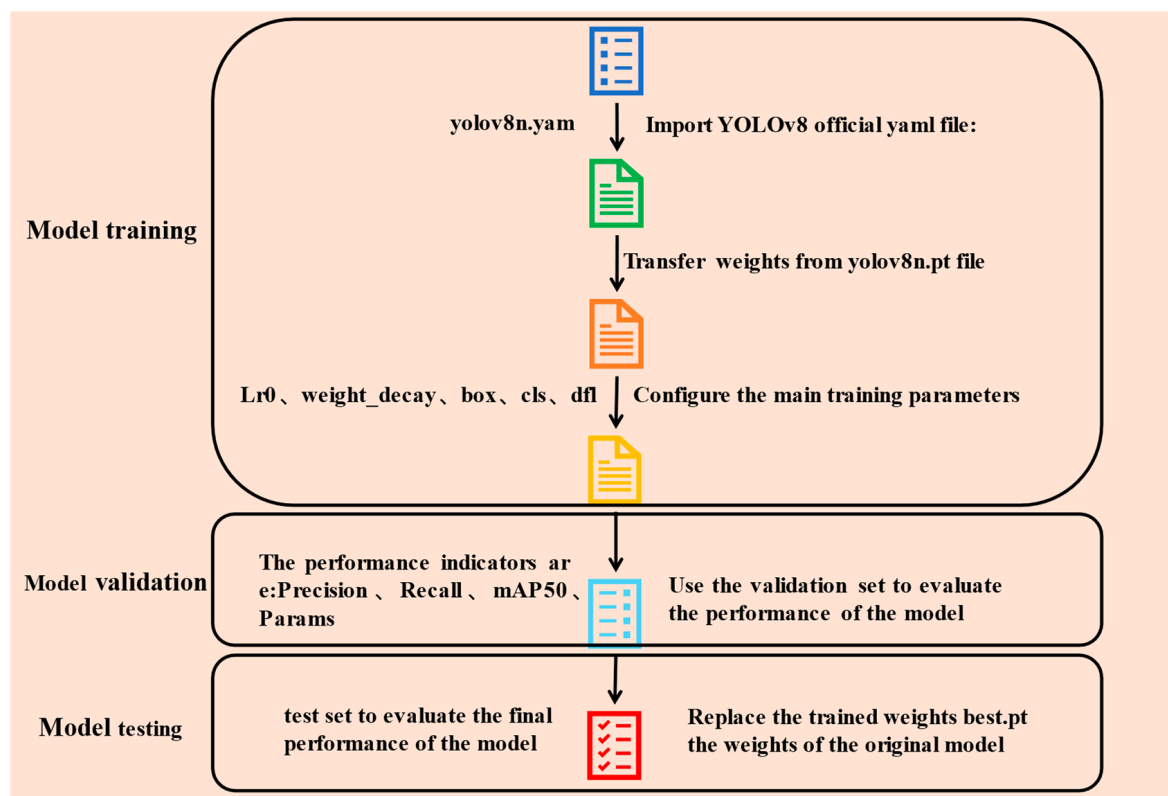


Figure 10. Flowchart of training, testing, and validation process.

3.2.2. Comparison of Experiments with Different Parameters

In the experiments, three key parameters were used: K1, K2, and T0, which are the core of the adaptive threshold mechanism. These parameters allow for the dynamic adjustment of the detection threshold based on the brightness and contrast of the input image. The K1 parameter primarily influences how the detection threshold is adjusted based on the brightness of the input image. If the value of K1 is larger, then the model will be more sensitive to images with higher brightness. Conversely, if the value of K1 is smaller, then the model will be more sensitive to images with lower brightness. The K2 parameter primarily affects how the detection threshold is adjusted based on the contrast of the input image. If the value of K2 is larger, then the model will be more sensitive to images with higher contrast. Conversely, if the value of K2 is smaller, then the model will be more sensitive to images with lower contrast. The T0 parameter is the baseline threshold of the adaptive threshold mechanism. The model will adjust this baseline threshold based on the values of K1 and K2 and the brightness and contrast of the input image.

To better understand how these parameters affect the performance of the model, a series of experiments was conducted and the results under different parameter settings were listed in Table 4. As shown in Table 4, when the parameters are set to K1 = 0.1, K2 = 0.2, and T0 = 0.5, the model performs best in the ‘Thunderbolt’ and ‘Break’ categories. Specifically, the precision, recall, and mAP50 for the ‘Thunderbolt’ category reached 87.7%, 91.5%, and 90.1% respectively, while those for the ‘Break’ category were 86.2%, 85.3%, and 87.6%. This parameter setting not only optimized the performance of the ‘Thunderbolt’ category but also achieved significant improvement in the ‘Break’ category. Therefore, it is reasonable to choose K1 = 0.1, K2 = 0.2, and T0 = 0.5 as the parameter settings, which will help improve the generalization ability and practical potential of the model.

Table 4. Comparison of results obtained with different parameter settings.

Labels	Parameters	Precision	Recall	mAP50
Thunderbolt	/	78.3%	82.6%	83.3%
Break		76.7%	77.1%	79.5%
Thunderbolt	K1 = 0.05; K2 = 0.15; T0 = 0.4	78.1%	85.2%	82.1%
Break		76.3%	82.8%	84.1%
Thunderbolt	K1 = 0.1; K2 = 0.2; T0 = 0.5	87.7%	91.5%	90.1%
Break		86.2%	85.3%	87.6%
Thunderbolt	K1 = 0.15; K2 = 0.25; T0 = 0.6	80.4%	87.1%	89.4%
Break		79.7%	81.5%	84.9%
Thunderbolt	K1 = 0.2; K2 = 0.3; T0 = 0.7	78.8%	87.3%	84.3%
Break		76.9%	80.9%	81.7%

To further demonstrate the effectiveness of the adaptive threshold mechanism in detecting damages to overhead power lines under different brightness and contrast conditions, the custom dataset was expanded to include four types of datasets: low brightness, high brightness, low contrast, and high contrast. These datasets were trained using the best.pt model. As shown in Figures 11 and 12, the detection results indicate that the YOLOv8 model with the adaptive threshold mechanism is more effective and accurate in detecting the damage to the overhead power lines compared to the original YOLOv8 model.

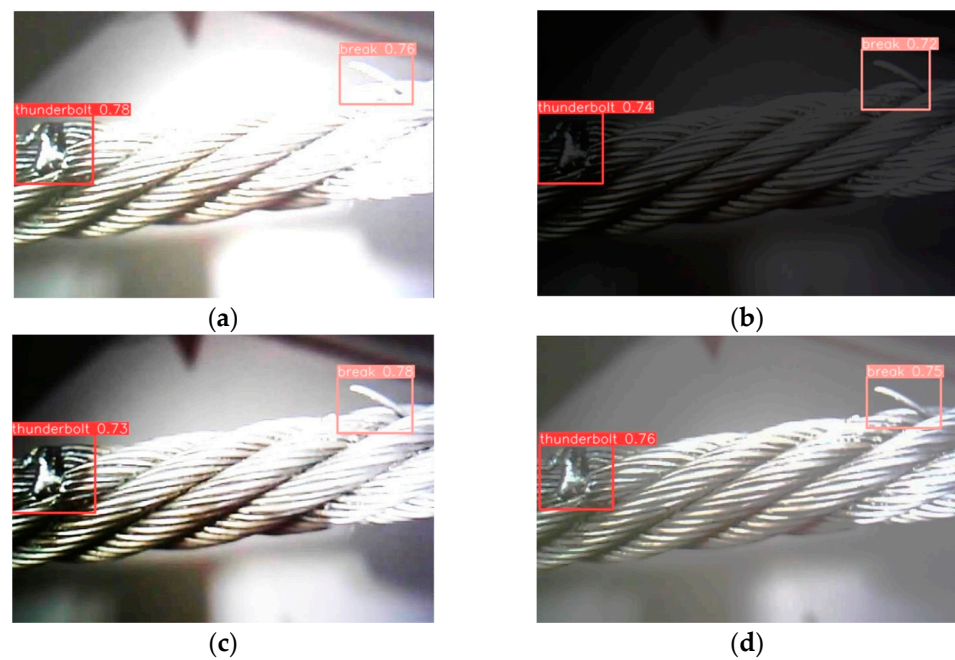


Figure 11. Detection results of the original YOLOv8 model under different environments: (a) High Brightness, (b) Low Brightness, (c) High Contrast, (d) Low Contrast.

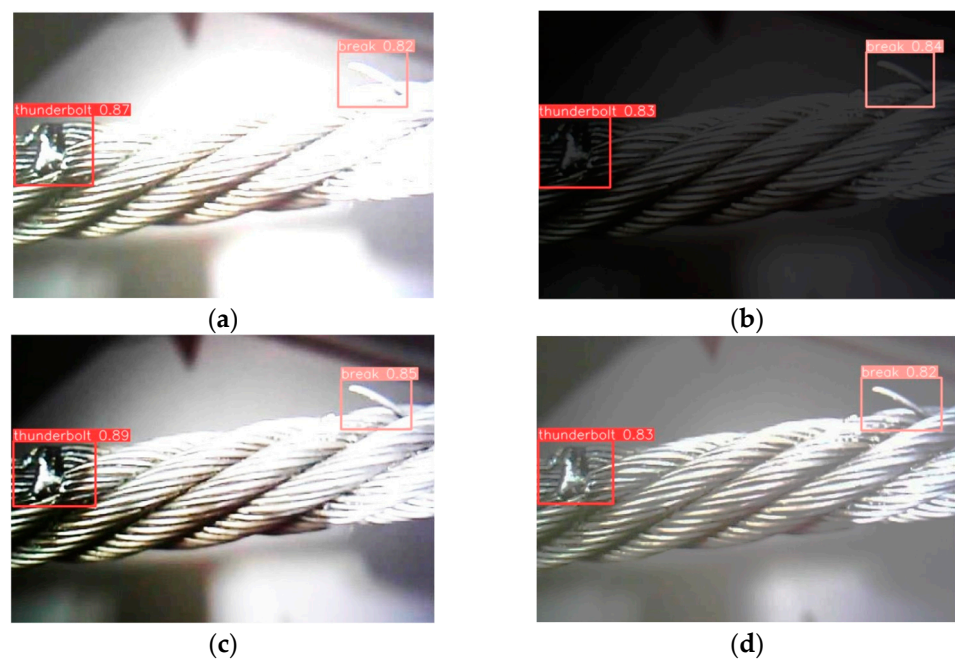


Figure 12. Detection results of the YOLOv8 model with adaptive threshold mechanisms under different environments: (a) High Brightness, (b) Low Brightness, (c) High Contrast, (d) Low Contrast.

3.2.3. Comparison of Different Layer Additions in GSConv Module

In order to evaluate the performance of GSConv in the damage detection of overhead power lines, this study compared it with the traditional convolution (Conv) in the YOLOv8 network model. GSConv is a new lightweight convolution technology that can maintain accuracy while reducing the model. GSConv achieves an excellent balance between model accuracy and speed. In the damage detection of overhead power lines, GSConv can achieve damage detection by extracting features. In this experiment, four sets of experiments were conducted: In the first set, GSConv was added to the fifth layer of the YOLOv8 backbone network; In the second set, GSConv was added to the seventh layer of the

YOLOv8 backbone network; In the third set, GSConv was simultaneously added to both the fifth and seventh layers of the YOLOv8 backbone network; In the fourth set, GSConv was not added to any layer. The purpose of these four sets of experiments was to evaluate the impact of adding GSConv at different layers on model performance.

The data from the Table 5 show that applying GSConv in the fifth and seventh layers can effectively optimize the model's performance. When GSConv is applied in the fifth and seventh layers, the mAP50 of the 'Thunderbolt' category reaches 84.5%, and the mAP50 of the 'Break' category reaches 86.3%. At the same time, the model's parameter size is reduced from 6.3 to 5.7 MB, and GFLOPs are also reduced from 8.1 to 7.5. These results indicate that GSConv can effectively reduce the model's complexity and computational load while maintaining its performance. Therefore, it is believed that applying GSConv in the fifth and seventh layers is an effective optimization strategy, which will help improve the practicality and generalization ability of the model.

Table 5. Analysis of the results of adding GSConv module.

Labels	GSConv Layers	Params	mAP50	GFLOPs
Thunderbolt Break	/	6.3 MB	83.5% 85.5%	8.1
Thunderbolt Break	5th Layer	6.2 MB	83.7% 86.3%	8.0
Thunderbolt Break	7th Layer	6.0 MB	84.5% 86.2%	8.0
Thunderbolt Break	5th and 7th Layers	5.7 MB	84.5% 86.3%	7.5

3.2.4. Comparative Analysis of Adding Slim Neck Module

Adding the Slim Neck module to YOLOv8 can improve the performance of the model while achieving better accuracy and speed. Slim Neck uses GSConv, a lightweight convolution technology that can reduce the burden of the model while maintaining accuracy. When this technology is applied to the damage detection of overhead power lines, it can improve the accuracy and efficiency of detection. This experiment replaces the convolution (C2f) in the YOLOv8 network model with the Slim Neck module and compares it with the original model.

As can be seen from the data in the Table 6, compared with C2f convolution, Slim Neck convolution has a slight increase in the speed of PyTorch, the number of parameters is reduced, mAP50 is almost unchanged in Thunderbolt, there is an increase in break, and GFLOPs are significantly reduced. This shows that Slim Neck convolution can improve the running speed and efficiency while maintaining the performance of the model, and it can also reduce the complexity of the model.

Table 6. Analysis of the results of adding Slim Neck module.

Labels	GSConv Layers	speed_PyTorch	Params	mAP50	GFLOPs
Thunderbolt Break	C2f	0.486	6.3 MB	0.959 0.855	8.1
Thunderbolt Break	Slim Neck	0.566	5.5 MB	0.958 0.902	6.7

3.2.5. Comparison of Ablation Experiments

In order to evaluate the accuracy and effectiveness of the algorithm proposed in this paper, and to examine the impact of each module on the performance indicators of the model, ablation experiments were conducted on the Cable Damage Detection dataset from RoboFlow. The YOLOv8 model was used as the base model, and different modules were

gradually added to evaluate the performance indicators of the network model. Table 7 shows the results of the ablation experiments.

Table 7. Comparison results of ablation experiments.

Labels	Methods	Precision	Recall	mAP50	Params
Thunderbolt Break	YOLOv8	79.3% 76.7%	82.6% 77.1%	83.3% 79.5%	6.3 MB
Thunderbolt Break	YOLOv8 + Adaptive Threshold	87.7% 86.2%	91.5% 85.3%	90.1% 87.6%	6.3 MB
Thunderbolt Break	YOLOv8 + Adaptive Threshold + GSConv	86.9% 87.1%	89.5% 83.7%	88.4% 87.9%	5.8 MB
Thunderbolt Break	YOLOv8 + Adaptive Threshold + GSConv + Slim Neck	87.8% 86.1%	91.6% 84.1%	90.2% 86.5%	5.2 MB

In the research, a series of improvements were made to the YOLOv8 model, and the effectiveness of these improvements was verified through ablation experiments.

Firstly, an adaptive threshold was added to the base YOLOv8 model. This improvement significantly enhanced the model's performance in handling the 'Thunderbolt' category with precision increasing from 79.3% to 87.7%, recall from increasing 82.6% to 91.5%, and mAP50 increasing from 83.3% to 90.1%. For the 'Break' category, precision increased from 76.7% to 86.2%, recall increased from 77.1% to 85.3%, and mAP50 increased from 79.5% to 87.6%.

Then, on the basis of adding the adaptive threshold, GSConv was further added. This improvement led to a slight decrease in precision to 86.9% for the 'Thunderbolt' category, but both recall and mAP50 increased to 89.5% and 88.4%, respectively. For the 'Break' category, precision slightly increased to 87.1%, but recall slightly decreased to 83.7%, and mAP50 slightly increased to 87.9%. Meanwhile, the model's parameter size was reduced from 6.3 to 5.8 MB.

Finally, Slim Neck was added. This resulted in an increase in precision to 87.8%, recall to 91.6%, and mAP50 to 90.2% for the 'Thunderbolt' category. For the 'Break' category, precision slightly decreased to 86.1%, but both recall and mAP50 slightly increased to 84.1% and 86.5%, respectively. At the same time, the model's parameter size was further reduced to 5.2 MB.

The experimental results show that the series of improvements made to the YOLOv8 model significantly enhanced the model's performance, effectively reducing the model's complexity while maintaining high accuracy and recall rate.

3.2.6. Experimental Comparison of Different Network Models

The improved YOLOv8 algorithm proposed in this paper is evaluated by using the Cable Damage Detection dataset from RoboFlow, and its performance is compared with several mainstream object detection network models, such as the single-stage object detection algorithms YOLO v3, YOLO v4, YOLO v5, YOLO7, YOLO8, and SSD and the two-stage object detection algorithm Faster R-CNN. Table 8 shows the relevant performance metrics obtained from the experiments.

As shown in Table 8, the detection results of different network models were compared. In terms of precision, recall, and mAP50, these models performed exceptionally well, achieving 90.3%, 91.6%, and 93.1%, respectively. These results are significantly superior to other models such as YOLOv3, YOLOv4, YOLOv5, YOLOv7, YOLOv8, Fast RCNN, and SSD.

As illustrated in Figure 13, the test results of different models on RoboFlow's Cable Damage Detection dataset indicate that these models are more accurate in detecting breakage and thunderbolt than other models. This demonstrates that these models exhibit

certain efficiency and stability when processing large-scale datasets, giving them certain advantages in practical applications.

Table 8. Detection results of different network models.

Methods	Precision	Recall	mAP50
YOLOv3	78.2%	72.1%	83.2%
YOLOv4	76.6%	74.2%	75.3%
YOLOv5	81.2%	81.6%	83.9%
YOLOv7	83.1%	82.3%	83.6%
YOLOv8	82.6%	81.2%	80.4%
Fast RCNN	63.2%	82.6%	84.3%
SSD	78.4%	78.2%	79.1%
Ours	90.3%	89.6%	91.1%

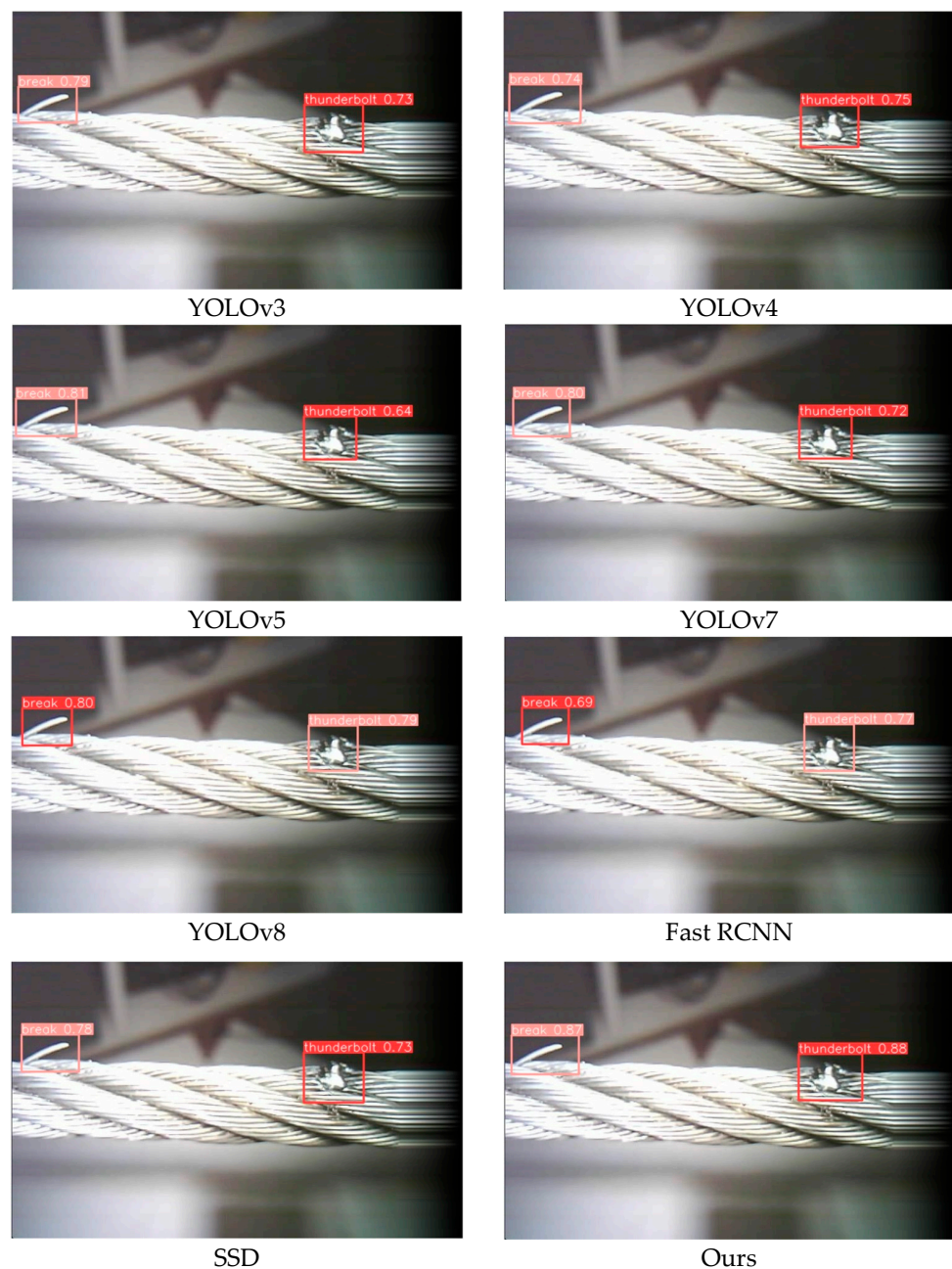


Figure 13. Experimental comparison results of different network models.

Next, the results of the newly introduced models in Table 9, EfficientDet, CenterNet, RetinaNet, and Mask R-CNN, are compared with the improved YOLOv8 algorithm. Among these new models, the precision (88.3%) and mAP50 (90.1%) of EfficientDet are slightly lower than the improved YOLOv8 algorithm, but its recall rate (84.6%) is higher. This may be because EfficientDet's performance is slightly inferior when dealing with some complex scenarios, but it excels in identifying targets in images. The precision (86.4%) and mAP50 (87.7%) of CenterNet are both lower than the improved YOLOv8 algorithm, but its recall rate (91.2%) is the highest among all models. This indicates that CenterNet excels in detecting all positive cases (i.e., real targets), but it may produce more false positives (i.e., erroneous detections). The precision (90.6%) of RetinaNet is slightly higher than the improved YOLOv8 algorithm, which may be because RetinaNet uses a special loss function (i.e., Focal Loss), making it excellent in dealing with class imbalance issues. However, its recall rate (89.2%) and mAP50 (88.6%) are both lower than the improved YOLOv8 algorithm. All indicators of Mask R-CNN are lower than the improved YOLOv8 algorithm, which may be because Mask R-CNN is mainly designed for instance segmentation tasks, not pure object detection, so it may not be as good as dedicated object detection models on some object detection indicators.

Table 9. Performance comparison between the enhanced version of YOLOv8 and other state-of-the-art models.

Methods	Precision	Recall	mAP50
EfficientDet	88.3%	84.6%	90.1%
CenterNet	86.4%	91.2%	87.7%
RetinaNet	90.6%	89.2%	88.6%
Ours	90.3%	89.6%	91.1%

Overall, although newly introduced models surpass the enhanced YOLOv8 algorithm in some performance metrics, the enhanced YOLOv8 algorithm still excels in comprehensive performance. This is primarily due to improvement strategies, including the introduction of an adaptive threshold mechanism to enhance the model's robustness. This mechanism can dynamically adjust the detection threshold based on the brightness, contrast, and other characteristics of the input image. Secondly, a new convolution method, GSConv, is adopted in the YOLOv8 framework, which balances the model's running speed and accuracy. Finally, a lightweight network structure, Slim Neck, is introduced, effectively reducing the model's complexity and computational load while maintaining good performance. These improvements enable the YOLOv8 model to excel in detecting 'thunderbolt' and 'break' types of cable damage. Especially in handling complex scenarios and class imbalance problems, the enhanced YOLOv8 algorithm shows significant advantages. In contrast, other models, although they perform well in some aspects, also expose some weaknesses, such as producing more false positives or performing poorly in pure object detection tasks. Therefore, the enhanced YOLOv8 algorithm remains the optimal choice in terms of overall performance and stability. These factors collectively determine the model's performance, highlighting the superiority of the enhanced YOLOv8 algorithm.

To further validate the model's performance in video processing, the model was deployed on a Windows 11 system and tested with a specific 11th Gen Intel® Core™ i5-1155G7 CPU for video processing. As shown in Table 10, the improved YOLOv8 model performed excellently in video processing, surpassing other network models. It achieved the highest levels in both average frame rate and peak frame rate, which were 13.58 and 18, respectively. This indicates that the model can not only maintain a high frame rate when processing video streams, but it also performs excellently in terms of peak performance.

Table 10. Video test results of different network models.

Methods	Average Frames	Maximum Number of Frames
YOLOv3	9.05	16
YOLOv4	8.52	15
YOLOv5	10.07	17
YOLOv7	8.12	15
YOLOv8	7.99	18
Fast RCNN	8.32	14
SSD	10.03	16
Ours	13.58	18

In the literature [30–33,36–38], it was found that the performance of most existing object detection models tends to decline when processing large-scale datasets. However, this model can not only maintain high accuracy when processing large-scale datasets but also maintain a high frame rate when processing video streams. This result indicates that the model has significant advantages in processing large-scale datasets and video streams.

In summary, the experimental results show that the proposed improved YOLOv8 algorithm outperforms other mainstream object detection network models in object detection tasks. This result validates the effectiveness and superiority of the model in practical applications.

4. Conclusions

In this study, an improved YOLOv8 model has been successfully proposed, which was specifically designed for high-voltage power line damage detection. The model enhances robustness and accuracy by introducing an adaptive threshold mechanism, the GSConv convolution method, and a lightweight network structure, Slim Neck, while reducing the complexity of the model.

However, it is also clearly recognized that despite significant progress in some areas, there are still some potential weaknesses and challenges. Firstly, although the introduction of Slim Neck reduces the complexity of the model, the adaptive threshold mechanism and GSConv convolution method may increase the computational requirements of the model, which could pose a challenge for resource-limited devices or systems. Secondly, as the model needs to learn how to dynamically adjust the detection threshold based on the characteristics of the input image, it may require a longer training time. The model has shown significant accuracy improvements when dealing with “lightning strike” and “break” labels. However, for other types of power line damage, such as corrosion or wear, the detection capability of the model may decline. In addition, if the background of the power line is complex or the lighting conditions are poor, it may affect the detection performance of the model. Despite these challenges, among all compared network models, this model achieved the best results on all indicators, demonstrating its superiority in object detection tasks. Especially in terms of precision, recall, and mAP50, the model achieved 90.3%, 89.6%, and 91.1%, respectively, all of which are the highest. The model has high robustness and accuracy, and it can dynamically adjust the detection threshold based on the characteristics of the input image, thus finding a balance between the running speed and accuracy of the model. Therefore, despite some challenges, this model is very suitable for the real-time detection of high-voltage power line damage. This paper will continue to optimize and improve this model to address these potential weaknesses and challenges in hopes of achieving better results in the future.

Author Contributions: J.G. served as the mentor, providing guidance on techniques and writing methodologies; Y.W. contributed as an author, participating in data analysis and interpretation of results; T.L., F.C., H.Z. and S.O. collaborated as classmates in conducting experiments and writing. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Key Project of Natural Science Foundation of Fujian Province No. 2020J02048, and Xiamen Municipal Natural Science Foundation 3502Z20227215, and Xiamen Ocean and Fisheries Development Special Fund Youth Science and Technology Innovation Project 23ZHZB043QCB37.

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Božiček, A.; Franc, B.; Filipović-Grčić, B. Early Warning Weather Hazard System for Power System Control. *Energies* **2022**, *15*, 2085. [\[CrossRef\]](#)
- Brettschneider, S.; Fofana, I. Evolution of Countermeasures against Atmospheric Icing of Power Lines over the Past Four Decades and Their Applications into Field Operations. *Energies* **2021**, *14*, 6291. [\[CrossRef\]](#)
- Lu, J.; Hu, J.; Fang, Z.; Qiao, X.; Zhang, Z. Electric Field Distribution and AC Breakdown Characteristics of Polluted Novel Lightning Protection Insulator under Icing Conditions. *Energies* **2021**, *14*, 7493. [\[CrossRef\]](#)
- Tarko, R.; Gajdzica, J.; Nowak, W.; Szpyra, W. Study of the Lightning Overvoltage Protection Effectiveness of High Voltage Mixed Overhead Cable Power Lines. *Energies* **2021**, *14*, 2329. [\[CrossRef\]](#)
- Chojnacki, A.L. Assessment of the Risk of Damage to 110 kV Overhead Lines Due to Wind. *Energies* **2021**, *14*, 556. [\[CrossRef\]](#)
- Qiao, K.; Zhu, A.; Wang, B.; Di, C.; Yu, J.; Zhu, B. Characteristics of Heat Resistant Aluminum Alloy Composite Core Conductor Used in Overhead Power Transmission Lines. *Materials* **2020**, *13*, 1592. [\[CrossRef\]](#) [\[PubMed\]](#)
- Esmeryan, K.D. From Extremely Water-Repellent Coatings to Passive Icing Protection—Principles, Limitations and Innovative Application Aspects. *Coatings* **2020**, *10*, 66. [\[CrossRef\]](#)
- Kumar, S.; Yadav, D.; Gupta, H.; Verma, O.P.; Ansari, I.A.; Ahn, C.W. A Novel YOLOv3 Algorithm-Based Deep Learning Approach for Waste Segregation: Towards Smart Waste Management. *Electronics* **2021**, *10*, 14. [\[CrossRef\]](#)
- Alauthman, M.; Al-qerem, A.; Sowan, B.; Alsarhan, A.; Eshtay, M.; Aldweesh, A.; Aslam, N. Enhancing Small Medical Dataset Classification Performance Using GAN. *Informatics* **2023**, *10*, 28. [\[CrossRef\]](#)
- Lim, S.-C.; Huh, J.-H.; Kim, J.-C. Siamese Trackers Based on Deep Features for Visual Tracking. *Electronics* **2023**, *12*, 4140. [\[CrossRef\]](#)
- Yang, L.; Fan, J.; Liu, Y.; Li, E.; Peng, J.; Liang, Z. A Review on State-of-the-Art Power Line Inspection Techniques. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 9350–9365. [\[CrossRef\]](#)
- Xu, C.; Li, Q.; Zhou, Q.; Zhang, S.; Yu, D.; Ma, Y. Power Line-Guided Automatic Electric Transmission Line Inspection System. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 3512118. [\[CrossRef\]](#)
- Zhou, Y.; Xu, C.; Dai, Y.; Feng, X.; Ma, Y.; Li, Q. Dual-View Stereovision-Guided Automatic Inspection System for Overhead Transmission Line Corridor. *Remote Sens.* **2022**, *14*, 4095. [\[CrossRef\]](#)
- Wang, L.; Wan, H.; Huang, D.; Liu, J.; Tang, X.; Gan, L. Sustainable Analysis of Insulator Fault Detection Based on Fine-Grained Visual Optimization. *Sustainability* **2023**, *15*, 3456. [\[CrossRef\]](#)
- 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. [\[CrossRef\]](#)
- Wang, Y.; Chen, P. A Lightweight Multi-Scale Quadratic Separation Convolution Module for CNN Image-Classification Tasks. *Electronics* **2023**, *12*, 4839. [\[CrossRef\]](#)
- Yang, H.; Li, X.; Guo, Y.; Jia, L. RT-GAN: GAN Based Architecture for Precise Segmentation of Railway Tracks. *Appl. Sci.* **2022**, *12*, 12044. [\[CrossRef\]](#)
- Gao, Y.; Liu, J.; Li, W.; Hou, M.; Li, Y.; Zhao, H. Augmented Grad-CAM++: Super-Resolution Saliency Maps for Visual Interpretation of Deep Neural Network. *Electronics* **2023**, *12*, 4846. [\[CrossRef\]](#)
- Sun, X.; Zhang, Q.; Wei, Y.; Liu, M. Risk-Aware Deep Reinforcement Learning for Robot Crowd Navigation. *Electronics* **2023**, *12*, 4744. [\[CrossRef\]](#)
- Hütten, N.; Meyes, R.; Meisen, T. Vision Transformer in Industrial Visual Inspection. *Appl. Sci.* **2022**, *12*, 11981. [\[CrossRef\]](#)
- Zhao, D.; Shao, F.; Yang, L.; Luo, X.; Liu, Q.; Zhang, H.; Zhang, Z. Object Detection Based on an Improved YOLOv7 Model for Unmanned Aerial-Vehicle Patrol Tasks in Controlled Areas. *Electronics* **2023**, *12*, 4887. [\[CrossRef\]](#)
- Peng, P.; Wang, Q.; Feng, W.; Wang, T.; Tong, C. An SAR Imaging and Detection Model of Multiple Maritime Targets Based on the Electromagnetic Approach and the Modified CBAM-YOLOv7 Neural Network. *Electronics* **2023**, *12*, 4816. [\[CrossRef\]](#)
- Xu, H.; Li, B.; Zhong, F. Light-YOLOv5: A Lightweight Algorithm for Improved YOLOv5 in Complex Fire Scenarios. *Appl. Sci.* **2022**, *12*, 12312. [\[CrossRef\]](#)
- Ruibin, S.; Kui, Q.; Weimin, X.; Hong, L. Adaptive Recognition of Complex Invoices Based on Tesseract-OCR. *Nanjing Xinxing Gongcheng Daxue Xuebao* **2021**, *13*, 349–354.
- Liu, L.; Huang, Z. Infrared Small Target Detection Technology Based on OpenCV. In Proceedings of the Signal Processing, Sensor Fusion, and Target Recognition XXII, Baltimore, MD, USA, 29 April–2 May 2013; Volume 8745, pp. 578–596.

26. Zhao, X.; Song, Y. Improved Ship Detection with YOLOv8 Enhanced with MobileViT and GSConv. *Electronics* **2023**, *12*, 4666. [CrossRef]
27. Hu, J.; Wang, Z.; Chang, M.; Xie, L.; Xu, W.; Chen, N. PSG-Yolov5: A Paradigm for Traffic Sign Detection and Recognition Algorithm Based on Deep Learning. *Symmetry* **2022**, *14*, 2262. [CrossRef]
28. Zhang, Y.; Shen, S.; Xu, S. Strip Steel Surface Defect Detection Based on Lightweight YOLOv5. *Front. Neurorobot.* **2023**, *17*, 1263739. [CrossRef]
29. Wen, G.; Li, M.; Luo, Y.; Shi, C.; Tan, Y. The Improved YOLOv8 Algorithm Based on EMSPConv and SPE-Head Modules. *Multimed. Tools Appl.* **2024**. [CrossRef]
30. Chen, K.; Wang, Y.; Wang, G.; He, J. The Real-Time Detection Method for Coal Gangue Based on YOLOv8s-GSC. *Res. Sq.* **2023**.
31. Niu, Y.; Cheng, W.; Shi, C.; Fan, S. YOLOv8-CGRNet: A Lightweight Object Detection Network Leveraging Context Guidance and Deep Residual Learning. *Electronics* **2024**, *13*, 43. [CrossRef]
32. Wang, Z.; Yuan, G.; Zhou, H.; Ma, Y.; Ma, Y. Foreign-Object Detection in High-Voltage Transmission Line Based on Improved YOLOv8m. *Appl. Sci.* **2023**, *13*, 12775. [CrossRef]
33. Yu, X.; Kuan, T.W.; Zhang, Y.; Yan, T. YOLO v5 for SDSB Distant Tiny Object Detection. In Proceedings of the 2022 10th International Conference on Orange Technology (ICOT), Shanghai, China, 10–11 November 2022; pp. 1–4.
34. Jiang, T.; Xue, Y. An Improved ORB Feature Point Extraction Algorithm Based on Image Enhancement and Adaptive Threshold. In Proceedings of the 2023 IEEE 7th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 15–17 September 2023; Volume 7, pp. 1233–1238.
35. Chen, G.; Sun, S.; Sun, Y.; Chen, H.; Zhang, W.; Su, X. CAGn: High-Order Coordinated Attention Module for Improving Fall Detection Models. In Proceedings of the IECON 2023—49th Annual Conference of the IEEE Industrial Electronics Society, Singapore, 16–19 October 2023; pp. 1–6.
36. Xiong, G.; Qi, J.; Wang, M.; Wu, C.; Sun, H. GCCE-YOLO: Improved YOLOv5s Algorithm for Object Detection in UAV Images. In Proceedings of the 2023 42nd Chinese Control Conference (CCC), Tianjin, China, 24–26 July 2023; pp. 7723–7728.
37. Zhao, Y.; Wang, H.; Xie, X.; Xie, Y.; Yang, C. An Enhanced YOLOv5-Based Algorithm for Metal Surface Defect Detection. *Appl. Sci.* **2023**, *13*, 11473. [CrossRef]
38. Li, J.; Pan, H.; Li, J. ESD-YOLOv5: A Full-Surface Defect Detection Network for Bearing Collars. *Electronics* **2023**, *12*, 3446. [CrossRef]
39. Al-Sabaawi, A.; Ibrahim, H.M.; Arkah, Z.M.; Al-Amidie, M.; Alzubaidi, L. Amended Convolutional Neural Network with Global Average Pooling for Image Classification. In *Proceedings of the Intelligent Systems Design and Applications*; Abraham, A., Piuri, V., Gandhi, N., Siarry, P., Kaklauskas, A., Madureira, A., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 171–180.
40. Zhang, Z.; Lv, G.; Zhao, G.; Zhai, Y.; Cheng, J. BS-YOLOv5s: Insulator Defect Detection with Attention Mechanism and Multi-Scale Fusion. In Proceedings of the 2023 IEEE International Conference on Image Processing (ICIP), Kuala Lumpur, Malaysia, 8–11 October 2023. Available online: <https://ieeexplore.ieee.org/document/10222163> (accessed on 4 January 2024).

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.