An Improved Algorithm for Wind Turbine Blade Defect Detection

XIUKANG RAN¹,², SHANG ZHANG²,³, HENGTAO WANG²,³, AND ZHAOYANG ZHANG¹,²
¹College of Electrical Engineering & New Energy, China Three Gorges University, Yichang, Hubei 443002, China
²Hubei Province Engineering Technology Research Center for Construction Quality Testing Equipments, China Three Gorges University, Yichang, Hubei 443002, China
³College of Computer and Information Technology, China Three Gorges University, Yichang, Hubei 443002, China
Corresponding author: Shang Zhang (wetoo@sina.com)

ABSTRACT With the increase in wind power generation, wind turbine blades require regular inspections to ensure they continue to operate safely. You only look once (YOLO) is one of the most widely used object detection algorithms and is easy to deploy into drone devices. To enhance the real-time detection of small target defects in wind turbine blades, this paper proposes an improved attention and feature balanced YOLO (AFB-YOLO) algorithm based on YOLOv5s. Specifically, AFB-YOLO improves the feature pyramid network by using weighted feature fusion and cross-scale connections. The improved feature pyramid network solves the problem that most previous feature pyramid networks treat all input features equally, and obtains more feature information. Furthermore, the coordinate attention (CA) module is introduced into the network to augment the representations of the objects of interest. Finally, the paper redesigned the loss function through efficient intersection over union (EIoU) loss to make the model obtain a better localization effect. The experimental results on the imagery of wind turbine blade defects indicate that our method shows significant gains in performance. The mean average precision (mAP50) of AFB-YOLO is 83.7% and the detection accuracy is improved by 4.0% compared to the original YOLOv5s model. The experiments in this paper demonstrate that AFB-YOLO is more effective and robust than state-of-the-art detectors.

INDEX TERMS Wind turbine blade, YOLOv5s, feature fusion, coordinate attention, EIoU loss.

I. INTRODUCTION To promote the positive development of the world’s energy-saving and emission-reduction economy, wind power has developed rapidly. However, the lifespan of the earliest wind turbines is nearing the end, and some wind turbines are even out of warranty. Defect detection is of great significance to the normal operation of wind turbines. Yang et al. [1] proposed a two-stage detection framework to detect bolt loosening, combining traditional manual torque methods commonly used in engineering with deep learning models to reduce manual detection costs and missed detection rates. This work discusses the defect detection of wind turbine blade. As the most critical and core component of a wind turbine, the integrity of the blade is an important factor in ensuring the stable operation of the wind turbine. Wind turbine blade are a component that causes many failures and high maintenance costs[2]. Due to the long-term harsh open-air working environment of wind turbine blades, various damage will inevitably occur on the surface of the wind turbine blades. Failure to detect and repair in time will lead to serious failure consequences [3], [4], [5]. Taking effective detection methods to find surface defects of wind turbine blades as soon as possible and repair them in time is the most effective way to reduce accident risks and economic losses. Before unmanned aerial vehicles (UAVs) are widely used, most of the research on wind turbine blade detection mainly focuses on the acquisition and processing of sensor signals. The commonly used methods are: vibration detection [6], acoustic emission technology [7], ultrasonic flaw detection [8], strain detection [9], infrared thermal imaging [10], etc. With the maturity of UAV technology equipped with high-definition cameras and the wide application of deep learning technology in the field of object detection [11]. A deep learning-based defect detection method for wind turbine blades emerges as the times require.

The task of object detection is to identify all interested targets in the image and determine their categories and positions, which is one of the core issues in the field of
computer vision. Object detection technology has reduced the burden on human beings to some extent and changed the way of life of human beings. Its accuracy and real-time performance are important criteria for measuring the entire computer vision system. Research on detection methods is divided into two categories: Traditional object detection algorithms [12], [13] and deep learning algorithms [14]. In the field of computer vision, traditional object detection algorithms are still popular. Traditional object detection algorithms extract artificial features based on prior knowledge and then use machine learning algorithms to classify them. The histogram of oriented gradients (HOG) is a commonly used feature extraction method and still performs well in object detection. Kassani et al. [15] improved the HOG feature and proposed soft HOG. Soft HOG is a more discriminative histogram that fully utilizes symmetric shapes. Xiao et al. [16] integrated HOG and convolutional neural networks to achieve the highest recall rate. Although traditional object detection algorithms are relatively mature, it is difficult to design robust features manually. With the continuous breakthrough of deep learning-based algorithms in the field of image recognition, many representative object detection algorithms have been proposed and widely used in industry, agriculture, medicine, and other fields [17], [18], [19], [20], [21]. Target detection algorithms based on deep learning can be divided into two categories: “two-stage detection” and “one-stage detection”. The two-stage deep learning algorithm generates candidate regions through selective search; and then combines the candidate regions with convolutional neural networks to extract features and perform regression classification. This kind of detection algorithm needs to be completed in two steps, so it is called two-stage deep learning algorithm [22], [23]. Two-stage deep learning algorithm has the characteristics of high accuracy but relatively slow speed. Typical detection algorithms based on candidate regions include the regions with CNN features (R-CNN) [24], fast region-based convolutional network (Fast R-CNN) [25], faster region-based convolutional network

![Diagram of object detection algorithms](image_url)
(Faster R-CNN) [26], and mask region-based convolutional network (Mask R-CNN) [27]. The above two-stage deep learning algorithms have the disadvantage of poor real-time performance, while the one-stage detection algorithm greatly improves the running speed of the algorithm. The one-stage detection algorithm reconstructs object detection as a single regression problem, directly from image pixels to bounding box coordinates and class probabilities [22], [23]. Typical one-stage detection algorithms include single shot multibox detector (SSD) [28], [29] series, and you only look once (YOLO) [30], [31], [32], [33], [34] series. Figure 1 illustrates some representative object detection algorithms.

The main contribution of this paper is to modify the operating mechanism of YOLOv5s and propose AFB-YOLO on the basis of YOLOv5s. AFB-YOLO draws inspiration from bidirectional feature pyramid network (BiFPN) and Coordinate Attention Module. The improved feature pyramid network enables the network to learn more important input features. The CA attention mechanism acquires more details about the target and suppresses other useless information to enhance the representation of objects of interest. The improved model can detect low-resolution and unclear wind turbine blade features, greatly improving detection of small target defects. Although the CIoU loss of YOLOv5s considers the aspect ratio of the predicted box and the ground truth box, it can speed up the regression of the predicted box to a certain extent. However, in the regression process of the predicted box, once the aspect ratio of the predicted box and the real box is linearly proportional, the and h of the predicted box cannot be increased or decreased at the same time, and the regression cannot continue to be optimized. AFB-YOLO redesigns the loss function through the Efficient Cross-Union (EIoU) loss, which enables the model to obtain better localization results.

The structure of this paper is as follows. Section 2 introduces the advantages of YOLOv5s and various techniques to improve the performance of YOLOv5s. Section 3 presents AFB-YOLO, including the motivation and structure of AFB-YOLO. In Section 4, experimental results and discussions are presented, where Precision, Recall, F1-Score, mean average precision (mAP), real-time performance in frames per second (FPS), and the model size are compared. Furthermore, Section 4 presents the comparison results of AFB-YOLO with other state-of-the-art lightweight models. Finally, the conclusion is drawn in Section 5.

II. RELATED WORKS
A. WHY YOLOV5S

The YOLOv5 algorithm has good network portability and is one of the most widely used models. YOLOv5 model is the most extensive lightweight object recognition network available today. YOLOv5 has four versions, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. YOLOv5s is the smallest depth and width network, which is only 14.4MB. Among them, YOLOv5s is very easy to deploy in drones due to its small size and fast computing speed [35]. The network structure of the YOLOv5s is shown in Figure 2. Convolution-batch normalization-sigmoid weighted liner unit (CBS) module, cross stage partial (CSP) module, spatial pyramid pooling fusion (SPPF) module, and other modules make up the network structure. The CBS module sequentially performs convolution computation, batch normalization, and SiLU activation function. The green CSP modules are CSP modules with residual structures. The green CSP module evenly divides the input tensor into two branches for convolution operations: one branch passes through a CBS module and then passes through residual units [36]; the other branch directly convolves and then concatenates the two branches. The orange CSP differs from the green CSP by replacing the residual unit with multiple CBS modules. The spatial pyramid pooling fusion (SPPF) module concatenates four paths to improve the receptive field of the network.

B. MULTISCALE FEATURE REPRESENTATIONS

Processing multiscale features is one of the main difficulties in object detection. Table 1 shows the related work of feature pyramid networks in recent years. The feature pyramid network (FPN) method proposed by Lin et al. makes the information of the shallow network and the deep network complement each other and improves the expressive ability of the deep convolutional neural network [37]. The feature layers of the deep convolutional neural network correspond to different information. The shallow network has a higher resolution and can learn the details of the target’s texture and shape; the deep network can learn the semantic information of the target. However, since FPN does not fully utilize the fusion of the detailed information of the shallow network and the semantic information of the deep network, Liu et al. [38] proposed the path aggregation network (PANet) method. PANet adds a bottom-up path to the FPN structure to further aggregate feature information of the shallow and deep layers and enhance the structure of the feature layer of the entire neural network. Ghiasi et al. [39] proposed neural architecture search feature pyramid network (NAS-FPN), which was the first to use NAS techniques to find the optimal FPN structure in the search space. The NAS-FPN search process uses reinforcement learning to train the controller. The controller uses the accuracy of the submodel in the search space as an excitation signal to update its parameters. The controller learns by trial and error to find the optimal FPN structure, and the discovered new FPN structure is called the NAS-FPN structure. Although NAS-FPN achieves better performance, NAS-FPN requires thousands of GPU hours to search. Recently, BiFPN has addressed the problem that input features have different resolutions and contribute unevenly to output features. BiFPN introduces learnable weights that enable the network to learn input features of varying importance [40].

C. ATTENTION MECHANISM

The importance of attention in human perception cannot be overstated. An important property of the human visual system is that it does not attempt to analyze the entire image at...
once, our attention is first focused on outstanding features and salient parts. The attention mechanism is a model proposed by Bengio et al. that simulates the attention mechanism of the human brain [41]. It can be regarded as a combination function, and the influence of a key input on the output is highlighted by calculating the probability distribution of attention. In a range of object detection tasks, attention mechanisms have proven helpful [42], [43], [44]. Channel attention and spatial attention are the two main types of attention mechanisms. The channel attention module focuses on channel information, but the spatial attention module focuses on positional information. Squeeze-and-excitation (SE) block [38] is a well-known channel attention mechanism that allows the model to learn the importance of different channel focuses on positional information. Squeeze-and-excitation [38] is a well-known channel attention mechanism that allows the model to learn the importance of different channel characteristics. Later works, the convolutional block attention module
CBAM) [45], [46], [47], [48] combines spatial and channel attention mechanism modules, using convolution to compute spatial attention. However, convolution can only capture local relations, but not the long-range dependencies required for object detection. In this work, the coordinate attention (CA) [49], [50] is introduced into the network. The CA attention mechanism decomposes the channel attention into two features for encoding, which capture long-range dependencies and precise location information along different spatial directions.

### III. IMPROVED OBJECT DETECTION ALGORITHM

#### A. BALANCE OF FEATURES

The sizes of the wind turbine blade vary widely in the images captured by drones, so handling multiscale features is critical. Traditional top-down FPN does not fully utilize the fusion of detailed information of shallow networks and semantic information of deep networks. To solve this problem, the PANet structure of YOLOv5s adds a bottom-up path, as shown in Figure 3(a). In this work, AFB-YOLO achieves two optimizations for cross-scale connectivity by improving an efficient weighted bidirectional feature pyramid network, as shown in Figure 3(b). First, the feature pyramid network of AFB-YOLO adds extra edges between input and output nodes in a similar way to residual connections, while borrowing the bidirectional path proposed by the PANet structure as the base layer for a more efficient feature fusion. Secondly, AFB-YOLO feature pyramid network introduces learning weight to learn the importance of different input features. AFB-YOLO is not simply adding or concatenating input features, which leads to feature mismatch. We help the model achieve more competitive performance by using fast normalized fusion with learnable weights. The improved weighted feature pyramid network of AFB-YOLO can better balance the feature information of wind turbine blades, so that the network can learn more important input features.

Taking into account the different importance of feature maps on different scales, bidirectional concat (BiConcat) adopts weighted fusion to fuse feature layers. The formulas of the two-branch weighted fusion method and the three-branch weighted fusion method are respectively defined as follows:

\[
P_{\text{out}}^{i} = \frac{\omega^1 \cdot p_{\text{in}}^1 + \omega^2 \cdot p_{\text{in}}^2}{\epsilon + \omega^1 + \omega^2} \quad (1)
\]

\[
P_{\text{out}}^{i} = \frac{\omega^1 \cdot p_{\text{in}}^1 + \omega^2 \cdot p_{\text{in}}^2 + \omega^3 \cdot p_{\text{in}}^3}{\epsilon + \omega^1 + \omega^2 + \omega^3} \quad (2)
\]

where \(\omega^i\) represents the input weight of the \(i\) branch, \(p_{\text{in}}^i\) represents the input feature of the \(i\) branch, and \(p_{\text{out}}^i\) represents the output feature. To avoid numerical instability, \(\epsilon\) is a small value. At the same time, by applying the ReLu activation function [51] to each weight, make sure that the weight value is greater than zero.

#### B. INTRODUCE ATTENTION MECHANISM

The attention module originates from the processing of image information by the human brain. By observing the global information of the image, humans can use attention to lock the focus area and automatically ignore some redundant background information. We applied the CA attention mechanism to the model. The improved model can obtain more detailed information about defects in the wind turbine blades and suppress otherwise useless information.

The CA attention mechanism is a novel attention mechanism. The CA attention mechanism can capture not only information about channel relationships, but also dependencies with precise location information. The CA attention mechanism can help the model locate and identify important information more accurately. As shown in Figure 4, the CA module is mainly divided into two steps: coordinate information embedding and coordinate attention generation. After input of the image information \(X\), the pooling layer encodes the channels along the horizontal and vertical coordinates, resulting in a feature map of the two coordinates. Generating a pair of orientation-aware feature maps enables the module to capture dependencies along one spatial path and preserve accurate location information along the other. Global pooling compresses global spatial information into channel descriptors, enabling the embedding of coordinate information. The output of the \(c\)-th channel with height \(h\) and width \(\omega\) is represented as:

\[
Z_{h}^{c}(h, i) = \frac{1}{W} \sum_{0 \leq i < W} x_{c}(h, i) \quad (3)
\]

\[
Z_{w}^{c}(\omega) = \frac{1}{H} \sum_{0 \leq j < H} x_{c}(j, \omega) \quad (4)
\]

The CA attention mechanism first transposes the output dimension of the horizontal dimension, then concatenates the two feature maps, and uses a \(1 \times 1\) convolution function to obtain the intermediate feature map. After the appeal operation, use BatchNorm and Non-linear to encode spatial information in vertical and horizontal directions. The features are then split into two separate tensors along the spatial dimension, where the output dimension of the horizontal dimension is transposed. Finally, convolve the two tensors on the horizontal and vertical coordinates respectively and use the sigmoid activation function to obtain two attention weights. Get the output \(Y\) of the coordinate attention block. Coordinate attention generation can accurately capture regions of interest and effectively capture inter-channel relationships. Through experiments, using the CA module embedded in YOLOv5s is

---

**TABLE 1.** The related work discussion on feature pyramid networks.

| Paper | Technique | Work Goals |
|-------|-----------|------------|
| 2017 Lin et al. [37] | FPN | Feature enhancement |
| 2018 Liu et al. [38] | PANet | Feature enhancement |
| 2019 Ghiassi et al. [39] | NAS-FPN | Search for optimal FPN |
| 2020 Tan et al. [40] | BiFPN | Feature enhancement |

---
beneficial to the extraction of network feature information.

\[ f = \delta(F_1([Z^h, Z^ω])) \] (5)

where \([\cdot, \cdot]\) represents the connection operation along the spatial dimension, \(F_1\) is the \(1 \times 1\) convolution transformation function, and \(\delta\) is the nonlinear activation function. After the above operations, the intermediate feature map \(f\) is obtained. Then decompose \(f \in R^{C/r \times (H+W)}\) into two separate tensors \(f_h \in R^{C/r \times H}\), \(f_w \in R^{C/r \times W}\) along the spatial dimension. Here, \(r\) is the reduction ratio for controlling the block size in the SE block.

\[ g^h = \sigma(F_h(f^h)) \] (6)

\[ g^ω = \sigma(F_ω(f^ω)) \] (7)

\[ y_c(i, j) = x_c(i, j) \times g^h_c(i) \times g^ω_c(j) \] (8)

Here \(\sigma\) is the sigmoid activation function, and \(F_h\) and \(F_ω\) are the other two \(1 \times 1\) convolution transformation functions. Then the outputs \(g^h\) and \(g^ω\) are expanded and used as attention weights, and finally, the output \(Y\) of the coordinate attention block is obtained.

To accurately identify and locate defects in wind turbine blades, this work uses a CA module to replace the CSP structure. The CA module can improve the network’s global receptive field and its ability to precisely pinpoint the target.

The improved network accommodates large changes in the frame size of wind turbine blade image detection by introducing a CA module. The whole improved detection architecture is shown in Figure 5.

C. IMPROVED LOSS FUNCTION

Bounding box regression is a key step in predicting bounding boxes to locate wind turbine blade defects, and the loss function of bounding box regression enables faster iterative convergence of the network model. In the YOLOv5s training process, the loss function and backpropagation are used to continuously update the model parameters to improve the detection accuracy. The loss function of YOLOv5s consists of three parts: localization loss, confidence loss, and class loss [18].

\[ \text{Loss}_{\text{total}} = \text{Loss}_{\text{CIoU}} + \text{Loss}_{\text{conf}} + \text{Loss}_{\text{cls}} \] (9)

Among them, \(\text{Loss}_{\text{CIoU}}\) is the localization loss of YOLOv5s. \(\text{Loss}_{\text{CIoU}}\) solves the inclusion problem of ground truth box and predicted box by calculating the Euclidean distance between boxes. \(\text{Loss}_{\text{conf}}\) is the confidence loss. \(\text{Loss}_{\text{conf}}\) will use cross-entropy to represent whether the anchor box contains objects or not. \(\text{Loss}_{\text{cls}}\) represents the loss of the object category. When the anchor box is predicted to be a
real object, the generated box will calculate the class loss. YOLOv5s adopts CIoU loss as the regression of bounding box, and considers the overlapping area, center point distance and aspect ratio of bounding box, which has high detection accuracy. However, once the aspect ratios of the predicted and ground truth boxes are linearly proportional, the predicted boxes ω, and h cannot be increased or decreased at the same time.

\[ \text{Loss}_{\text{CIoU}} = \text{Loss}_{\text{iou}} + \text{Loss}_{\text{dis}} + \text{Loss}_{\text{asp}} \]
\[ = 1 - \text{IoU} \frac{\rho^2(b, h^\text{gt})}{c^2} + \alpha \upsilon \]
\[ \alpha = \frac{1}{(1 - \text{IoU}) + \upsilon} \]
\[ \upsilon = \frac{4}{\pi^2} \left( \arctan \frac{\omega^\text{gt}}{h^\text{gt}} - \arctan \frac{\omega}{h} \right)^2 \]

where IoU represents the proportion of intersection on the union of bounding box, where b and b^gt represent the center point of the predicted box and the ground truth box, respectively; \(\rho(*)\) represents the Euclidean distance. c represents the diagonal distance of the smallest rectangle formed by the two bounding boxes, and \(\upsilon\) is a weight function that measures the similarity of the aspect ratio.

In the loss function of AFB-YOLO, the class loss and confidence loss are the same as YOLOv5s, except that the localization loss uses the EIoU loss [52] instead of the CIoU loss [53]. Although CIoU loss considers the overlapping area, center point distance, and aspect ratio of bounding box regression. However, the aspect ratio difference reflected by CIoU loss is not the real width and height difference, which sometimes hinders effective optimization of the model. In response to this problem, the EIoU loss divides the aspect ratio based on the CIoU loss. The EIoU loss of confidence is as follows:

\[ \text{Loss}_{\text{EIoU}} = \text{Loss}_{\text{iou}} + \text{Loss}_{\text{dis}} + \text{Loss}_{\text{asp}} \]
\[ = 1 - \text{IoU} \frac{\rho^2(b, h^\text{gt})}{c^2} + \frac{\rho^2(\omega, \omega^\text{gt})}{c_\omega} \]
\[ + \frac{\rho^2(h, h^\text{gt})}{c_h} \]

where b, \(\omega\), h represent the center point, width, and height of the predicted box, respectively; \(b^\text{gt}\), \(\omega^\text{gt}\), \(h^\text{gt}\) represent the center point, width, and height of the ground truth box, respectively. \(\rho(*)\) indicates the Euclidean distance. c is the diagonal length of the smallest enclosing box covering the two boxes. \(c_\omega\) and \(c_h\) are the width and height of the minimum bounding rectangle of the predicted box and ground truth box.

The EIoU loss divides the bounding box regression (BBR) loss into three parts: the overlap loss between the predicted box and the ground truth box (\(\text{Loss}_{\text{iou}}\)), the center distance loss between the predicted and ground truth boxes (\(\text{Loss}_{\text{dis}}\)), and the width and height loss of the predicted and ground truth boxes (\(\text{Loss}_{\text{asp}}\)). The first two parts of the EIoU loss continue the method in the CIoU loss, while the localization loss directly minimizes the difference between the width and height of the predicted box and the ground truth box, making the convergence faster. The figure 6 shows the iterative process of the CIoU loss and EIoU loss predicted box. The red boxes and the green boxes are the regression process of the predicted boxes, the blue boxes is the target box, and the black boxes is the preset anchor boxes.

The CIoU loss of YOLOv5s cannot effectively measure the difference between bounding boxes, resulting in slow model convergence and inaccurate positioning. The \(\text{Loss}_{\text{asp}}\) of CIoU loss has the measure of the aspect ratio of the predicted box and the real box, which can speed up the regression of the predicted box to a certain extent. However, in the regression process of the predicted box, once the aspect ratio of the predicted box and the real box is in a linear ratio, the \(\omega\) and h of the predicted box cannot be increased or decreased at the same time, and the regression optimization cannot be continued.

AFB-YOLO redesigns the EIoU loss to minimize the width and height difference between the target box and the anchor box for better localization results. The EIoU loss splits the loss term of the aspect ratio into the difference between the predicted box width and height and the minimum bounding box width and height, which accelerates the convergence of the predicted box and improves the regression accuracy of the predicted box. The improvement of the loss function effectively improves the convergence speed and accuracy of the model training.

### IV. EXPERIMENTS RESULTS

#### A. EXPERIMENTAL PLATFORM AND DATASET

Table 2 lists the software and hardware platforms and parameters used in this work. The operating system is Windows 10, and the neural network framework is Pytorch. Pytorch is an open-source python machine learning toolkit based on Torch, which is widely used in various machine learning algorithms. The processor we used is an Intel Core i5-11400H @ 2.70GHz, and the graphics card is an NVIDIA GeForce RTX 3050.

In applied research on wind turbine blade defect detection, there is a lack of publicly available datasets suitable for deep learning training. In this work, we used a drone to take different angles around the wind turbine blades, and collected a total of 559 images. Since the insufficient number of samples

| Software and hardware platform | Parameters |
|--------------------------------|------------|
| Operating system              | Windows 11 |
| Deep learning framework       | Pytorch    |
| CPU                           | Intel Core i5-11400H @ 2.70GHz |
| GPU                           | NVIDIA GeForce RTX 3050 |
| Hardware acceleration         | CUDA11.3   |

In the loss function of AFB-YOLO, the class loss and confidence loss are the same as YOLOv5s, except that the localization loss uses the EIoU loss [52] instead of the CIoU loss [53]. Although CIoU loss considers the overlapping area, center point distance, and aspect ratio of bounding box regression. However, the aspect ratio difference reflected by CIoU loss is not the real width and height difference, which sometimes hinders effective optimization of the model. In response to this problem, the EIoU loss divides the aspect ratio based on the CIoU loss. The EIoU loss of confidence is as follows:

\[ \text{Loss}_{\text{EIoU}} = \text{Loss}_{\text{iou}} + \text{Loss}_{\text{dis}} + \text{Loss}_{\text{asp}} \]
\[ = 1 - \text{IoU} \frac{\rho^2(b, h^\text{gt})}{c^2} + \frac{\rho^2(\omega, \omega^\text{gt})}{c_\omega} \]
\[ + \frac{\rho^2(h, h^\text{gt})}{c_h} \]

where b, \(\omega\), h represent the center point, width, and height of the predicted box, respectively; \(b^\text{gt}\), \(\omega^\text{gt}\), \(h^\text{gt}\) represent the center point, width, and height of the ground truth box, respectively. \(\rho(*)\) indicates the Euclidean distance. c is the diagonal length of the smallest enclosing box covering the two boxes. \(c_\omega\) and \(c_h\) are the width and height of the minimum bounding rectangle of the predicted box and ground truth box.

The EIoU loss divides the bounding box regression (BBR) loss into three parts: the overlap loss between the predicted box and the ground truth box (\(\text{Loss}_{\text{iou}}\)), the center distance loss between the predicted and ground truth boxes (\(\text{Loss}_{\text{dis}}\)), and the width and height loss of the predicted and ground truth boxes (\(\text{Loss}_{\text{asp}}\)). The first two parts of the EIoU loss continue the method in the CIoU loss, while the localization loss directly minimizes the difference between the width and height of the predicted box and the ground truth box, making the convergence faster. The figure 6 shows the iterative process of the CIoU loss and EIoU loss predicted box. The red boxes and the green boxes are the regression process of the predicted boxes, the blue boxes is the target box, and the black boxes is the preset anchor boxes.

The CIoU loss of YOLOv5s cannot effectively measure the difference between bounding boxes, resulting in slow model convergence and inaccurate positioning. The \(\text{Loss}_{\text{asp}}\) of CIoU loss has the measure of the aspect ratio of the predicted box and the real box, which can speed up the regression of the predicted box to a certain extent. However, in the regression process of the predicted box, once the aspect ratio of the predicted box and the real box is in a linear ratio, the \(\omega\) and h of the predicted box cannot be increased or decreased at the same time, and the regression optimization cannot be continued.

AFB-YOLO redesigns the EIoU loss to minimize the width and height difference between the target box and the anchor box for better localization results. The EIoU loss splits the loss term of the aspect ratio into the difference between the predicted box width and height and the minimum bounding box width and height, which accelerates the convergence of the predicted box and improves the regression accuracy of the predicted box. The improvement of the loss function effectively improves the convergence speed and accuracy of the model training.
TABLE 3. Dataset used in this work.

| Dataset          | Training dataset | Test dataset |
|------------------|------------------|--------------|
| Number of Images | 2396             | 599          |
| Number of Labels | 7513             | 1836         |

will lead to problems such as overfitting of the model training, this work uses a random cropping method to expand the initial wind turbine blade dataset. After removing the background image from the cropped image, 2995 images are obtained. Data augmentation by random cropping helps to improve the robustness of our feature detector in the conv network while removing a large amount of background images. The image has two categories of defect: damage and dirt. The training input image size is $640 \times 640$, and the batch size is set to 8.

We roughly divide the training data set and the test data set by 4:1, and the specific data set division is shown in Table 3.

### B. PERFORMANCE EVALUATION FOR DEFECT DETECTION

If the IoU of the predicted box and the ground truth box is greater than 0.5, the two boxes are matched. For binary classification problems, examples can be divided into TP, FP, and FN according to their predicted categories, and their concepts are provided below.

(i) True positives (TP) are the number of samples that the model correctly predicted.
(ii) False positive (FP) is the number of samples whose background area is mistakenly identified as a defect.
(iii) False negative (FN) is the number of defects that the model has not yet predicted.

Precision, Recall, F1-Score, average precision (AP), and mean average precision (mAP) are commonly used metrics to evaluate model performance in object detection tasks [54]. The calculation formula of Precision, Recall, F1-Score, AP, and mAP are respectively defined as follows:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$  \hspace{1cm} (14)

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$  \hspace{1cm} (15)

$$F_1\text{-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$  \hspace{1cm} (16)

$$\text{AP} = \int_0^1 P(R)dR$$  \hspace{1cm} (17)

$$\text{mAP} = \frac{1}{C} \sum_{i=1}^{C} \text{AP}_i$$  \hspace{1cm} (18)

where $\text{AP}_i$ represents the AP value of the ith class and $C$ represents the total number of classes. The mAP reflects the average classification performance of the model for all classes. In addition, F1-Score is a metric for classification problems and is often used as the final metric for multi-classification problems. It is the harmonic mean of precision and recall.

### C. ABLATION EXPERIMENT

We pool a variety of tricks to improve YOLO’s performance. The improved feature pyramid network enables the network to learn more important input features. The CA attention mechanism acquires more details about the target and suppresses other useless information. The EIoU loss achieves precise regression of the bounding box. To verify the three improvement tricks proposed in this work for YOLOv5s, ablation experiments were carried out on the dataset of wind turbine blade defects to judge the effectiveness of each improvement point. The weighted feature pyramid network, CA attention mechanism, and loss function are added to the original model in turn. On our experimental platform, 300 epochs of training were performed with the same parameter configuration, and the results are shown in Table 4.

After introducing the weighted feature pyramid network, it can be seen that the recall rate increased by 3.7% compared to YOLOv5s. At the same time, the average accuracy is improved by 2.5%. However, after introducing the weighted feature pyramid network and the attention mechanism, the performance improvement is weak when the EIoU loss is introduced. The analysis shows that the introduction of the weighted feature pyramid network and the attention mechanism enhance the ability of the network to extract features, paying attention to much semantic information that is easily submerged. Compared with YOLOv5s, the proposed algo-
TABLE 4. Abortion experiment results.

| Model       | Weighted Feature Pyramid network | CA attention mechanism | EIoU loss | Precision (%) | Recall (%) | F1-Score (%) | mAP50 (%) |
|-------------|----------------------------------|------------------------|-----------|---------------|------------|--------------|------------|
| YOLOv5s     | -                                | -                      | -         | 82.9          | 79.0       | 80.9         | 80.5       |
| 1           | √                                | -                      | -         | 81.1          | 81.9       | 81.5         | 82.5       |
| 2           | -                                | √                      | -         | 83.2          | 81.7       | 82.4         | 82.1       |
| 3           | -                                | -                      | √         | 83.2          | 80.3       | 81.7         | 81.5       |
| 4           | -                                | √                      | -         | 82.6          | 80.5       | 81.5         | 82.4       |
| 5           | √                                | -                      | √         | 83.0          | 79.9       | 81.4         | 82.6       |
| 6           | √                                | √                      | -         | 82.2          | 81.1       | 81.6         | 83.5       |
| 7           | √                                | √                      | √         | 83.4          | 82.0       | 82.7         | 83.7       |

TABLE 5. Comparison of the speed and accuracy of different object detectors.

| Algorithm    | Precision (%) | Recall (%) | F1-Score (%) | mAP50 (%) | FPS | Weight(MB) |
|--------------|---------------|------------|--------------|-----------|-----|------------|
| Faster R-CNN | 41.1          | 70.0       | 51.8         | 58.3      | 4.9 | 108.0      |
| Mask R-CNN   | 59.7          | 66.0       | 62.7         | 62.0      | 5.1 | 123.1      |
| SSD          | 85.2          | 49.6       | 62.7         | 61.4      | 47.6| 91.1       |
| YOLOv4-tiny  | 67.7          | 41.1       | 51.1         | 51.4      | 85.9| 22.4       |
| YOLOv4       | 76.2          | 63.1       | 69.0         | 69.1      | 24.6| 244.0      |
| YOLOv3       | 80.8          | 66.1       | 72.7         | 75.6      | 35.8| 235.0      |
| YOLOv5s      | 82.9          | 79.0       | 80.9         | 80.5      | 64.5| 11.4       |
| AFB-YOLO     | 83.4          | 82.0       | 82.7         | 83.7      | 63.4| 11.6       |

![Figures](image_url)

**FIGURE 7.** Wind turbine blade defect detection results. (a) Faster R-CNN detection effect, (b) SSD detection effect, (c) YOLOv3 detection effect, (d) YOLOv5s detection effect, (e) AFB-YOLO detection effect.

Algorithm AFB-YOLO improves the performance with gains of 3.8% Recall, 2.2% F1-Score, and 4.0% mAP50. The results show that the detection results of AFB-YOLO are relatively comprehensive, and it has a good detection effect on the image of wind turbine blade defects.

D. PERFORMANCE COMPARISON OF THE DIFFERENT MODELS

Table 5 shows the comparative performance of AFB-YOLO with other state-of-the-art models. Compared with the two-stage deep learning algorithms FasterR-CNN and MaskRCNN, AFB-YOLO has a great improvement in accuracy and the speed is nearly 12 times faster. Compared with the classic one-stage object detection detector SSD, AFB-YOLO has a great improvement in various indicators. Compared with YOLOv3 and YOLOv4, AFB-YOLO also has great improvements in accuracy and speed. AFB-YOLO is 24.1%, 13.8%, and 10.7% higher than YOLOv3 in the detection accuracy indicators Recall, F1-Score, and mAP50, respectively. When considering only speed, YOLOv4-tiny has the best detection speed. However, YOLOv4-tiny prunes the network structure, which greatly reduces the accuracy. In this work, the improved AFB-YOLO achieves higher performance compared to YOLOv5s. AFB-YOLO improves...
F1-Score by 2.2% and mAP50 by 4.0% while maintaining the detection speed.

Small object detection has always been a hot and difficult point in the field of object detection. The main challenge is that small target defects have few pixels and it is difficult to extract effective feature information [55]. Figure 7 shows the results of different algorithms on the wind turbine blade defect dataset. On large target defects, the target detection algorithms all perform well. However, compared to the detection performance of large targets, there is still a large gap in the detection performance of small targets. In the second and third rows of images, most algorithms miss small target defects seriously. Small target defects carry few features, so it is difficult to extract the effective features of defects. AFB-YOLO achieves good detection performance in small target defects by improving the weighted feature pyramid network, attention mechanism, and loss function. For example, in (e2), YOLOv5s detected small target defects, but with low confidence. AFB-YOLO improves the appealing problem and achieves better detection accuracy. Furthermore, AFB-YOLO detected small target defects in (e3) that were not detected in (d3). The results show that the proposed model has a good detection effect on small target defects.

V. CONCLUSION

Inspired by BiFPN and the coordinate attention module. In this work, a YOLOv5s-based defect detection algorithm for wind turbine blades is proposed. Through experimental verification, AFB-YOLO has the characteristics of high real-time performance and stable performance in the wind turbine blade dataset. The accuracy of AFB-YOLO is similar to the two-stage detection algorithm Faster R-CNN, but the detection speed is significantly improved. Compared with YOLOv3, AFB-YOLO improves the detection accuracy indicators Recall, F1-Score, and mAP50 by 24.1%, 13.8%, and 10.7%, respectively. Compared with the latest one-stage object detection algorithm YOLOv5s, the detection accuracy is still greatly improved. The good performance of the proposed model can be attributed to learning important feature information. The improved model is capable of detecting low resolution and unclear features of wind turbine blades, greatly improving the detection of small target defects. In future work, we plan to use more powerful matching strategies to improve our detection results.

REFERENCES

[1] X. Yang, Y. Gao, C. Fang, Y. Zheng, and W. Wang, “Deep learning-based bolt loosening detection for wind turbine towers,” Struct. Control Health Monit., vol. 29, no. 6, Jun. 2022, Art. no. e2943.
[2] Y. Du, S. Zhou, X. Jing, Y. Peng, H. Wu, and N. Kwok, “Damage detection techniques for wind turbine blades: A review,” Mech. Syst. Signal Process., vol. 141, Jul. 2020, Art. no. 106445.
[3] G. Lichtenegger, A. A. Rentizelas, N. Triviya, and S. Siegl, “Offshore and onshore wind turbine blade waste material forecast at a regional level in Europe until 2050,” Waste Manag., vol. 106, pp. 120–131, Apr. 2020.
[4] A. Norgren, A. Carpenter, and G. Heath, “Design for recycling principles applicable to selected energy technologies: Crystalline-silicon photovoltaic modules, electric vehicle batteries, and wind turbine blades,” J. Sustain. Metall., vol. 6, no. 4, pp. 761–774, Dec. 2020.
[5] A. Rentizelas, N. Triviya, S. Oswald, and S. Siegl, “Reverse supply network design for circular economy pathways of wind turbine blades in Europe,” Int. J. Prod. Res., vol. 60, no. 6, pp. 1795–1814, Mar. 2022.
[6] A. Abouhnik and A. Albarbar, “Wind turbine blades condition assessment based on vibration measurements and the level of an empirically decomposed feature,” Energy Convers. Manage., vol. 64, pp. 606–613, Dec. 2012.
[7] Q. Hongwu, S. Haixin, C. Wei, and D. Mengcong, “Structural health monitoring WTB using the effectiveness of graphical programming packages analysis on acoustic emission data,” in Proc. IEEE 5th Int. Conf. Big Data Cloud Comput., Aug. 2015, pp. 207–212.
[8] K. Yang, J. A. Rongong, and K. Worden, “Damage detection in a laboratory wind turbine blade using techniques of ultrasonic NDT and SHM,” Strain, vol. 54, no. 6, Dec. 2018, Art. no. e12290.
[9] K. Lee, A. Aihara, G. Puntsagdash, T. Kawaguchi, and M. Okuma, “Feasibility study on a strain based deflection monitoring system for wind turbine blades,” Mech. Syst. Signal Process., vol. 82, pp. 117–129, Jan. 2017.
[10] C. Q. G. Muñoz, F. P. G. Márquez, and J. M. S. Tomás, “Ice detection using thermal infrared radiometry on wind turbine blades,” Measurement, vol. 93, pp. 157–163, Nov. 2016.
[11] A. Shihaudevvin, M. R. A. Rashid, M. H. Maruf, M. A. Hasan, M. A. U. Haq, R. H. Ashique, and A. A. Mansur, “Image based surface damage detection of renewable energy installations using a unified deep learning approach,” Energy Rep., vol. 7, pp. 4566–4576, Nov. 2021.
[12] D. Forsyth, “Object detection with discriminatively trained part-based models,” Comput. Graph., vol. 47, no. 2, pp. 6–7, Feb. 2014.
[13] S. Huang, N. Cai, P. Pucheco, S. Narandres, Y. Wang, and W. Xu, “Applications of support vector machine (SVM) learning in cancer genomics,” Cancer Genomics Proteomics, vol. 15, no. 1, pp. 41–51, Jan./Feb. 2018.
[14] S. Li, Y. Li, Y. Li, M. Li, and X. Xu, “YOLO-FIRE: Improved YOLOv5 for infrared image object detection,” IEEE Access, vol. 9, pp. 148161–148175, 2021.
[15] P. H. Kassani and A. B. J. Teoh, “A new sparse model for traffic sign classification using soft histogram of oriented gradients,” Appl. Soft Comput., vol. 52, pp. 231–246, Mar. 2017.
[16] Z. Xiao, Z. Yang, L. Geng, and F. Zhang, “Traffic sign detection based on histograms of oriented gradients and Boolean convolutional neural networks,” in Proc. Int. Conf. Mach. Vis. Inf. Technol. (CMVIT), Feb. 2017, pp. 111–115.
[17] J. Yan and Z. Wang, “YOLO V3 + VGG16-based automatic operations monitoring and analysis in a manufacturing workshop under industry 4.0,” J. Manuf. Syst., vol. 63, pp. 134–142, Apr. 2022.
[18] R. Shi, T. Li, and Y. Yamaguchi, “An attribution-based pruning method for real-time mango detection with YOLO network,” Comput. Electron. Agric., vol. 169, Feb. 2020, Art. no. 105214.
[19] L. Shiqi, G. Jia, W. Jiang, and M. Xuan, “Recognition algorithm for small packages of traditional Chinese medicine based on improved YOLOV4,” in Proc. Int. Conf. Cyber-Phys. Social Intell. (ICCSI), Dec. 2021, pp. 1–4.
[20] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, “Traffic-sign detection and classification in the wild,” in Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 2110–2118.
[21] J. Yu and W. Zhang, “Face mask wearing detection algorithm based on improved YOLO-v4,” Sensors, vol. 21, no. 9, p. 3263, May 2021.
[22] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, “Object detection with deep learning: A review,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 11, pp. 3212–3232, Nov. 2019.
[23] Z. Zou, Z. Shi, Y. Guo, and J. Ye, “Object detection in 20 years: A survey,” 2019, arXiv:1905.05055.
[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.
[25] R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1440–1448.
[26] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in Proc. Adv. Neural Inf. Process. Syst., vol. 28, 2015, pp. 1–9.
[27] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in IEEE Int. Conf. Comput. Vis., Oct. 2017, pp. 2961–2969.
[28] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in Proc. Eur. Conf. Comput. Vis. Springer, 2016, pp. 21–37.
[29] X. Xu, H. Zheng, Z. Guo, X. Wu, and Z. Zheng, “SDD-CNN: Small data-driven convolution neural networks for subtle roller defect inspection,” Appl. Sci., vol. 9, no. 7, p. 1364, 2019.
[30] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 779–788.

[31] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 7263–7271.

[32] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018, arXiv:1804.02767.

[33] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” 2020, arXiv:2004.10934.

[34] C.-Y. Wang, A. Bochkovskiy, and H.-Y.-M. Liao, “Scaled-YOLOv4: Scaling cross stage partial network,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 13029–13038.

[35] Z. Zhao, J. Han, and L. Song, “YOLO-highway: An improved highway center marking detection model for unmanned aerial vehicle autonomous flight,” Math. Problems Eng., vol. 2021, Jun. 2021, Art. no. 1205153.

[36] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

[37] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 2117–2125.

[38] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, “Path aggregation network for instance segmentation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 8759–8768.

[39] G. Ghiasi, T.-Y. Lin, and Q. V. Le, “NAS-FPN: Learning scalable feature pyramid architecture for object detection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 7036–7045.

[40] M. Tan, R. Pang, and Q. V. Le, “EfficientDet: Scalable and efficient object detection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 10781–10790.

[41] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” 2014, arXiv:1409.0473.

[42] X. Li, H. Zhang, L. Feng, J. Hu, R. Zhang, and Q. Qiao, “Edge-aware image outputting with attentional generative adversarial networks,” IET Image Process., vol. 16, no. 7, pp. 1807–1821, May 2022.

[43] J. Chen, L. Wan, J. Zha, G. Xu, and M. Deng, “Multi-scale spatial and channel-wise attention for improving object detection in remote sensing imagery,” IEEE Geosci. Remote Sens. Lett., vol. 17, no. 4, pp. 681–685, Apr. 2020.

[44] Z. Nan, J. Peng, J. Jiang, H. Chen, B. Yang, J. Xin, and N. Zheng, “A joint object detection and semantic segmentation model with cross-attention and inner-attention mechanisms,” Neurocomputing, vol. 463, pp. 212–225, Nov. 2021.

[45] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional block attention module,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 3–19.

[46] S.-H. Wang, Q. Zhou, M. Yang, and Y.-D. Zhang, “ADVIAN: Alzheimer’s disease VGG-inspired attention network based on convolutional block attention module and multiple way data augmentation,” Frontiers Aging Neurosci., vol. 13, p. 313, Jan. 2021.

[47] B. Chen, Z. Zhang, N. Liu, Y. Tan, X. Liu, and T. Chen, “Spatiotemporal convolutional neural network with convolutional block attention module for micro-expression recognition,” Information, vol. 11, no. 8, p. 380, Jul. 2020.

[48] S. Du, B. Zhang, P. Zhang, P. Xiang, and H. Xue, “FA-YOLO: An improved YOLO model for infrared occlusion object detection under confusing background,” Wireless Commun. Mobile Comput., vol. 2021, pp. 1–10, Nov. 2021.

[49] Q. Hou, D. Zhou, and J. Feng, “Coordinate attention for efficient mobile network design,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 13713–13722.

[50] M. Zha, W. Qian, W. Yi, and J. Hua, “A lightweight YOLOv4-based forestry pest detection method using coordinate attention and feature fusion,” Entropy, vol. 23, no. 12, p. 1587, Nov. 2021.

[51] X. Glorot, A. Bordes, and Y. Bengio, “Deep sparse rectifier neural networks,” in Proc. 14th Int. Conf. Artif. Intell. Statist., 2011, pp. 315–323.

[52] Y.-F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, and T. Tan, “Focal and efficient IOU loss for accurate bounding box regression,” 2021, arXiv:2101.08158.

[53] Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, and W. Zhao, “Enhancing geometric factors in model learning and inference for object detection and instance segmentation,” 2020, arXiv:2005.05572.

F. Xie, B. Lin, and Y. Liu, “Research on the coordinate attention mechanism in a YOLOv5 deep learning detector for the SAR ship detection task,” Sensors, vol. 22, no. 9, p. 3370, Apr. 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/9/3370

Q. Xu, R. Lin, H. Yue, H. Huang, Y. Yang, and Z. Yao, “Research on small target detection in driving scenarios based on improved YOLO network,” IEEE Access, vol. 8, pp. 27574–27583, 2020.

**XIUKANG RAN** was born in Jingzhou, Hubei, China, in 1999. He received the B.S. degree in electrical engineering and automation from the Wuhan Institute of Technology, Hubei, in 2021. He is currently pursuing the master’s degree in electrical engineering with China Three Gorges University. His research interests include computer vision, image processing, and detection of electrical equipment.

**SHANG ZHANG** was born in Yichang, Hubei, China, in 1997. He received the Ph.D. degree from the China University of Geosciences, in 2014. In 2019, he was an Associate Professor with the School of Computer and Information, Three Gorges University. His research interests include image processing and artificial intelligence.

**HENGTAO WANG** was born in Shandong, China, in 1996. He received the B.S. degree from the Shandong Youth University of Political Science, China, in 2020. He is currently pursuing the M.S. degree with China Three Gorges University. His current research interests include computer vision, image processing, and deep learning.

**ZHAOYANG ZHANG** was born in Shangqiu, Henan, China, in 2004. He received the B.S. degree from the University of Science and Technology of China, in 2021. He is currently pursuing the M.S. degree with China Three Gorges University. His current research interests include image processing and artificial intelligence.