applied sciences

MDPI

Article

Surface Crack Detection Method for Coal Rock Based on Improved YOLOv5

Xinlin Chen [1], Qingwang Lian [1,*], Xuanlai Chen [2,*] and Jin Shang [1]

[1] Key Laboratory of In-Situ Property-Improving Mining of Ministry of Education, School of Mining Engineering, Taiyuan University of Technology, Taiyuan 030024, China
[2] School of Engineering, The University of Western Australia, Perth, WA 6009, Australia
* Correspondence: lianqingwang@tyut.edu.cn (Q.L.); xuanlai.chen@research.uwa.edu.au (X.C.)

Abstract: Coal mine safety may be able to be ensured via the real-time identification of cracks in rock and coal surfaces. Traditional crack identification methods have the disadvantages of slow speed and low precision. This work suggests an improved You Only Look Once version 5 (YOLOv5) detection model. In this study, we improved YOLOv5 from the perspective of three aspects: a Ghost module was introduced into the backbone network to lighten the model; a coordinate attention mechanism was added; and ECIOU_Loss is proposed as a loss function in this paper to achieve the co-optimization of crack detection speed and accuracy and to meet the deployment requirements in the embedded terminal. The results demonstrate that the improved YOLOv5 has a 92.8% mean average precision (mAP) with an 8 MB model size, and the speed of recognition was 103 frames per second. Compared to the original method, there was a 53.4% reduction in the number of parameters, a detection speed that was 1.9 times faster, and a 1.7% improvement in the mAP. The improved YOLOv5 can effectively locate cracks in real time and offers a new technique for the early warning of coal and rock dynamic hazards.

Keywords: coal rock cracks; YOLOv5; Ghost module; CA; ECIOU

check for updates

Citation: Chen, X.; Lian, Q.; Chen, X.; Shang, J. Surface Crack Detection Method for Coal Rock Based on Improved YOLOv5. *Appl. Sci.* **2022**, *12*, 9695. https://doi.org/10.3390/app12199695

Academic Editor: Giuseppe Lacidogna

Received: 9 August 2022
Accepted: 24 September 2022
Published: 27 September 2022

1. Introduction

A typical coal rock dynamic catastrophe experienced during coal mining is rock burst. When rock burst happens, coal and rocks are flung out abruptly, damaging equipment, obstructing traffic, and resulting in injuries [1]. The issue of rock burst has become worse in coal mines, as mining depths have progressively increased, presenting a major danger to coal mine crew safety and underground production operations [2,3]. The beginning and progression of cracks to penetration often accompany the instability failure process of coal and rock mass under impact loads [4,5]. In addition to being the most obvious indicator of the coal and rock mass's present condition, the cracks that are produced on their surfaces are also a sign of their impending breakdown under dynamic impact [6–8].

With the quick advancement of computer vision technology, several traditional crack detection methods have been developed, with the main ones including segmentation, region growth, edge detection, support vector machines, and k-nearest neighbor, in digital image correlation technology [9]. By analyzing cracks in coal CT images, Li et al. [10] established a crack segmentation technique and crack profile evolution model based on contour evolution and gradient direction consistency, resulting in improved crack segmentation. By using the partial differential method and the improved C-V model during image processing, Chen et al. [11] improved the coal rock picture and estimated parameters such as the length, area, and spacing of the cracks. Li et al. [12] suggested a technique for detecting cracks on the surface of coal bodies based on the vibration loading failure test and SVM. Miao et al. [13] used acoustic emissions and digital image processing technologies to conduct uniaxial compression experiments on sandstone with various dip angles and to identify and categorize the crack shape. These techniques are straightforward and intuitive, and they can quickly and precisely display parameters such as crack propagation, direction, and length. However, they are generally difficult to use, expensive, and unable to be applied in some situations in which high detection accuracy and no contact are required. Future developments will focus on the quick, extensive, and automated analysis of a large number of pictures transmitted on-site against the backdrop of current intelligent mine construction.

Different from traditional methods, which need to design and extract features manually, deep learning obtains features through automatic learning, which can improve the accuracy of model recognition [14,15]. At present, deep learning algorithms are the mainstream technology for image recognition, with examples including the two-stage target detection algorithm based on candidate frames and the one-stage regression-based target detection algorithm [16–18]. The two-stage algorithm detection method, which uses algorithms such as RCNN [19], Fast-RCNN [20], and Faster-RCNN [21], involves the target item being detected and then classified. End-to-end neural networks that can anticipate item categories and bounding boxes are known as one-stage detection algorithms. Examples include the SSD [22] and YOLO series [23–26]. Compared to other networks, YOLO is extensively used in a variety of sectors and has great generalization capabilities while also showing better performance as well as superior stability. At present, it is one of the most used frameworks for object detection. Cui et al. [27] improved the YOLOv3 algorithm to identify concrete damage, and the model's accuracy increased to 75.68% as a result. Zhao et al. [28] proposed a bauxite sorting model that enhanced YOLOv3 by integrating the SE attention mechanism [29] with the K-means algorithm to enhance the detection model's feature extraction and propagation capabilities. In order to discover fabric flaws, Yue et al. [30] clustered the prediction framework using the K-means algorithm, added the prediction layer, adjusted the loss function, and suggested an enhanced YOLOv4 algorithm. Liu et al. and Yan et al. [31,32] provided technological help for automated fruit harvesting by determining the maturity of fruits using the enhanced YOLOv5 model. To find flaws on the surface of aero engine components, Li et al. [33] suggested the YOLOv5s-KEB model, which combines the representation capabilities of improved networks such as ECA and BiFPN. Breast cancers were identified using the YOLOv5 model by Aqsa Mohiyuddin et al. [34].

Deep learning can automatically extract high-level semantic data from photos, opening up a new method for intelligent crack detection. With the development of deep learning, tunnel and bridge cracks have received extensive attention in terms of research and development, but at present, the use of YOLOv5 to identify coal and rock cracks has not been studied by scholars. Through the research presented in this paper, we can improve the information extraction of crack features and enrich crack identification research.

This study introduces an improved YOLOv5 coal rock surface crack detection method that takes into account that the model should have high flexibility, quick detection speed, compact size, cheap deployment cost, and great applicability in practical applications. In the first phase, we replaced the primary network layer with the Ghost [35] lightweight feature extraction network to minimize the number of network parameters and computations and to speed up the identification of crack targets. In order to strengthen attention to the foreground area, enrich the semantic information of the shallow feature map, and increase the target detection accuracy, we introduced a coordinate attention (CA) [36] module in the second step. This addressed the issue of the shallow feature map containing more background noise and insufficient semantic information. The final phase involved the use of the enhanced loss function ECIOU_Loss to hasten model convergence and achieve precise crack categorization and location.

2. Materials

2.1. Coal Sample Selection

The coal samples used in this experiment were selected from the #11 coal seam of the Xinliangyou Coal Industry in Pu County, Linfen, and the #9 coal seam of the Shuozhou Xiayao Coal Industry.

The Linfen #11 coal seam's macroscopic coal rock features mostly include brilliant coal and black coal, with a tiny proportion of mirror coal and silk coal also present. This coal sample is from the semi-bright–semi-dark briquette category. The coal sample has a metamorphic degree of II–III with the following being present in thirds: coking coal, gas coal, and fat coal. The coal has the following features: black in color, pitch-vitreous sheen, hard and brittle, staggered cracks, well-developed cracks, strip-like structure, layered, massive structure, and pyrite nodules in certain places. Existing research demonstrates that when the coal seam is hard and when black coal predominates, rock burst is more likely to occur.

With the exception of bright and dark coal, the Shuozhou #9 coal seam's macro-coal characteristics are mostly semi-bright coal and dark coal, with the black coal having a high degree of hardness. The coal samples' degree of metamorphism falls within the I–II metamorphic stage, which is characterized by thick, hard cracks with angular or irregular cracks that are filled with calcite veins.

2.2. Sample Preparation

Coal samples were obtained from the working faces of two coal mines along the trough. The working face's flat coal wall was where the coal samples were collected. Coal samples were wrapped in plastic wrap and were then placed in a foam box during transit to avoid weathering and collision. Experiments were carried out in the in situ modified mining mechanics laboratory of the Taiyuan University of Technology.

Requirements for sample preparation were as follows: A CNC sand wire cutting machine was used for coal sample processing and production. This machine was used to take two types of test pieces of various sizes: the first were 50 mm × 100 mm coal pillar test pieces, and the second were 50 mm × 25 mm coal pillar specimens.

2.3. Data Collection

Since there was no existing dataset pertaining to coal rock cracks, data collection activities needed to be completed before conducting the study. WDW-100KN microcomputer-controlled electronic universal testing equipment was used in this study to conduct tests on the physical and mechanical parameters of coal samples [37]. The loading method adopted displacement control, and the loading rate was set to 0.001 mm/s. As soon as a crack appeared on the specimen's surface, the dataset gathering process was started for the cracks. In total, more than 1600 photographs of coal rock cracks were obtained; 100 of these photos were chosen as the test set, and the other photos were randomly split into training and validation shots in a 7:3 ratio. Table 1 shows the divide.

Table 1. Dataset division.

Dataset	Quantity
train	1103
val	458
test	100

2.4. Data Annotation

The LabelImg [38] software, which is available for Windows, was used. In this experiment, the YOLO data format was used, and each picture's label information was immediately stored in a text file with the same name as the image. Table 2 displays a portion of the text file's content that was saved following data annotation.

Table 2. Crack callout information sheet.

Class	X [2]	Y [3]	W [4]	H [5]
0 [1]	0.500000	0.322266	0.988281	0.251953
0	0.380859	0.685547	0.421875	0.605469
0	0.499023	0.507324	0.992188	0.366211
0	0.495117	0.555176	0.976562	0.184570
0	0.519043	0.061523	0.032227	0.072266

In the table, [1] indicates the labeling category (in this paper, coal and rock cracks), [2,3] represent the center coordinates of the label box, and [4,5] represent the relative width and height of the label box. X, Y, H, and W all normalize the data.

3. Method

3.1. Principle of Coal Rock Crack Recognition

YOLO takes in the full picture and utilizes complete image information to make predictions. YOLO has a greater field of vision through which it can identify targets than the sliding window approach since it may use the whole picture information during training and prediction.

The principle of coal rock crack image detection is shown in Figure 1.

(1) The input picture was divided into N × N grid cells.

(2) Each grid consisted of five a priori boxes (anchor boxes) to predict the crack bounding boxes, and a reasonable a priori box shape and number were selected as the initial a priori boxes. According to the offset value of the predicted coordinates, the position of the target center point and the width and height of the predicted frame were calculated.

(3) By calculating the classification probability and crack location of the objects in the current boundary box, the confidence is obtained. The confidence is used to express the class probability of the target in the prior box and the performance of matching the target.

(4) The confidence threshold was set to screen the bounding box, and the repeatedly detected cracks were removed using non-maximum suppression (NMS), thereby realizing the precise location of the cracks.

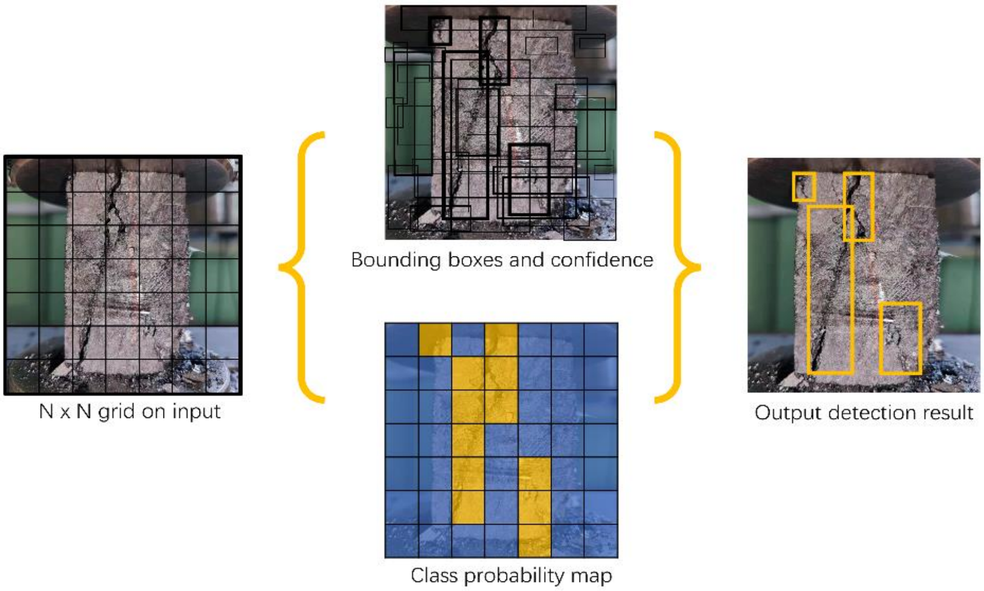


Figure 1. Crack detection and identification.

3.2. YOLOv5 Network Structure

The input, backbone, neck, and prediction components made up the fundamental foundation of the YOLOv5-6.0 network model used in this paper. An image preparation step, such as image scaling, normalization, mosaic data augmentation, or adaptive anchor box computation, often takes place as part of the input. Common feature representations, which are primarily separated into the Conv module, C3 module, and SPPF module, are often extracted using the backbone component. The Conv module encapsulates three functions: convolution, batch normalization, and SiLU activation. The C3 module draws on the idea of CSPNet and includes three standard convolutional layers and multiple bottleneck modules. The shortcut parameter controls whether or not the residual connection is performed. The SPPF module uses serial maxpool to perform multi-scale fusion to expand the feature map of the receptive field. The neck part can further improve the diversity and robustness of the features. This part adopts FPN and PAN structures to strengthen the network feature fusion ability. The FPN layer conveys strong semantic features from top to bottom, while the feature pyramid conveys strong localization features from the bottom up. The two combine to perform parameter aggregation for different detection layers from different backbone layers. Prediction is used to complete the output of target detection results. CIOU_Loss [39,40] is used as the loss function of the bounding box, and NMS non-maximum suppression is used to screen the multi-object box and output the predicted image. The YOLOv5-6.0 network structure is shown in Figure 2.

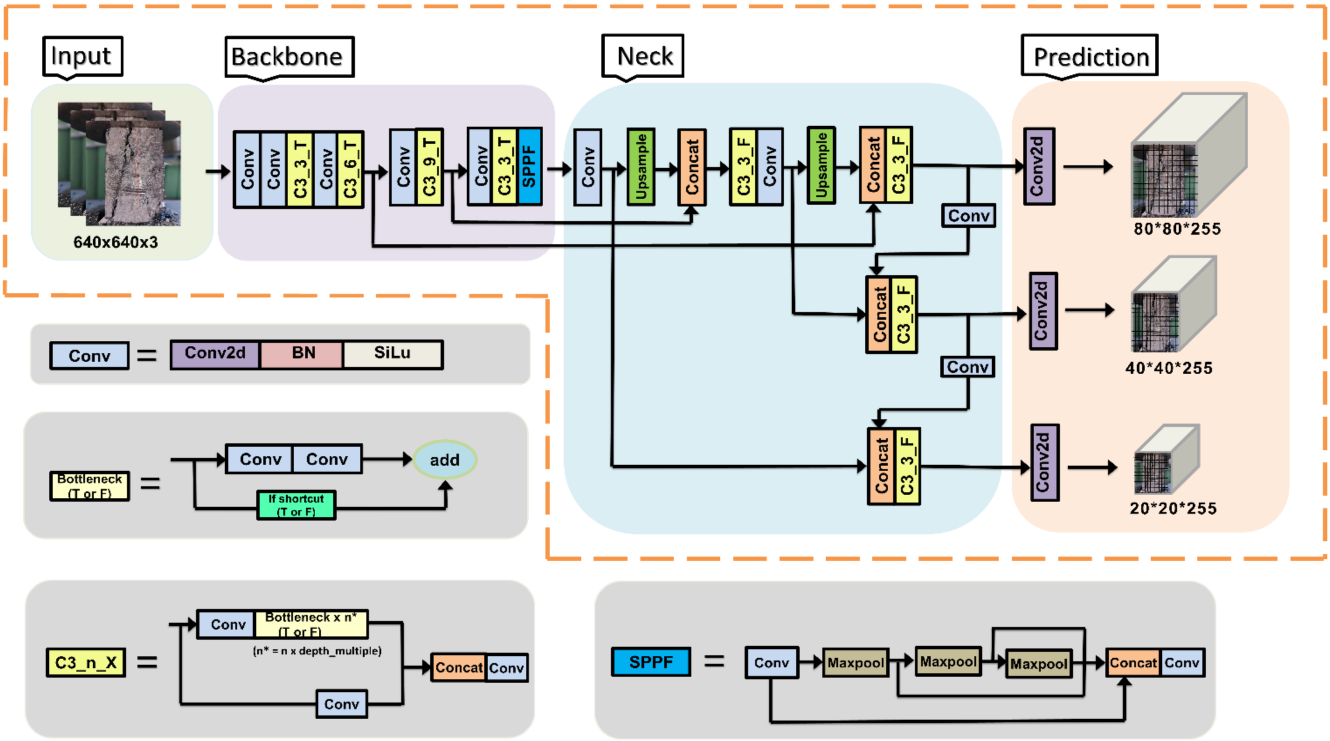


Figure 2. Network structure diagram of YOLOv5-6.0.

3.3. Model Improvement and Optimization

This research offered an enhanced lightweight network YOLOv5-G model based on YOLOv5 to increase the operational efficiency of the model and decrease hardware costs while maintaining accuracy. First, in order to make the network lighter, reduce the number of parameters, and accelerate computation, we introduced the Ghost module; second, to boost feature extraction and increase network accuracy, a coordinate attention mechanism was included in the backbone network; finally, quick crack localization was accomplished using border regression with ECIOU_Loss as the loss function. The improved YOLOv5-G network structure is shown in Table 3.

Table 3. The network structure of YOLOv5-G.

Num [1]	From [2]	Params [4]	Module [5]	Arguments [6]
0	−1 [3]	3520	Conv	[3, 32, 6, 2, 2]
1	−1	10,144	GhostConv	[32, 64, 3, 2]
2	−1	11,344	CAC3Ghost	[64, 64, 1]
3	−1	38,720	GhostConv	[64, 128, 3, 2]
4	−1	43,512	CAC3Ghost	[128, 128, 2]
5	−1	151,168	GhostConv	[128, 256, 3, 2]
6	−1	177,872	CAC3Ghost	[256, 256, 3]
7	−1	597,248	GhostConv	[256, 512, 3, 2]
8	−1	614,944	CAC3Ghost	[512, 512, 1]
9	−1	656,896	SPPF	[512, 512, 5]
10	−1	69,248	GhostConv	[512, 256, 1, 1]
11	−1	0	Upsample	[None, 2, 'nearest']
12	[−1, 6]	0	Concat	[1]
13	−1	208,608	C3Ghost	[512, 256, 1, False]
14	−1	18,240	GhostConv	[256, 128, 1, 1]
15	−1	0	Upsample	[None, 2, 'nearest']
16	[−1, 4]	0	Concat	[1]
17	−1	53,104	C3Ghost	[256, 128, 1, False]
18	−1	75,584	GhostConv	[128, 128, 3, 2]
19	[−1, 14]	0	Concat	[1]
20	−1	143,072	C3Ghost	[256, 256, 1, False]
21	−1	298,624	GhostConv	[256, 256, 3, 2]
22	[−1, 10]	0	Concat	[1]
23	−1	564,672	C3Ghost	[512, 512, 1, False]

In the table, [1] represents the network layer serial number; [2] represents which layer the input of the current network comes from, [3] represents the output from the previous layer, [4] represents the size of the parameter, [5] is the network module, and [6] represents the information of the module parameters (including the number of input channels, the number of output channels, the size of the convolution kernel, and the step size information).

3.3.1. Ghost Lightweight Model

The traditional feature extraction approaches used for convolutional neural networks include the use of convolution mapping operations on all input channels using a number of convolution kernels. However, stacking several convolutional layers results in rich, often redundant feature maps and requires a significant amount of processing and parameters. Figure 3 shows a crack picture from the dataset after a typical convolution. The feature network extraction layer's initial convolution layer may be seen to have several redundant pictures that are quite similar to one another. Although these redundant feature maps are useful for network training as well, creating them requires a lot of computation and is costly. We also anticipate that the concept will be portable to traditional hardware platforms or mobile devices in terms of actual use. This study enhanced model lightweight operation for the original YOLOv5 method based on the two reasons mentioned above.

Huawei Noah's Ark Lab proposed the GhostNet lightweight model at the 2020 CVPR conference, and its core module is the Ghost module. The purpose of the Ghost module is to substitute portions of the standard convolution for feature extraction via linear transformation. Some redundant feature maps may be modified from the conventional convolution feature maps that are obtained to collect comparable feature maps. This greatly reduces the number of convolution calculations and parameters, and it can be easily integrated into other networks. The difference between the ordinary convolution module and the Ghost convolution module is shown in Figure 4.

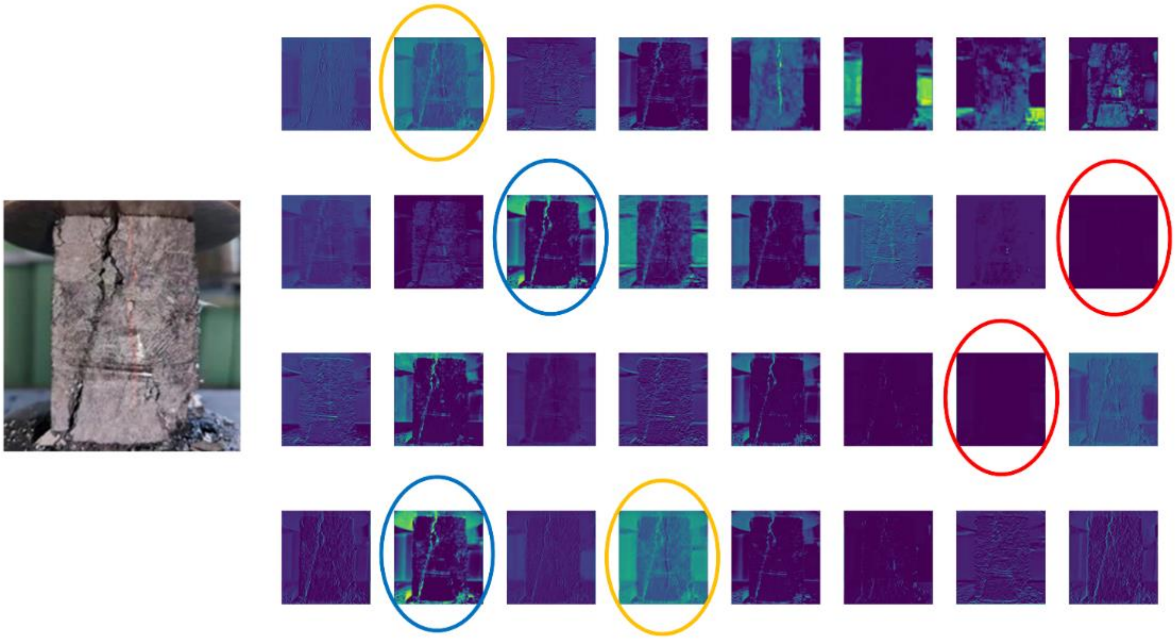


Figure 3. Convolutional layer feature map.

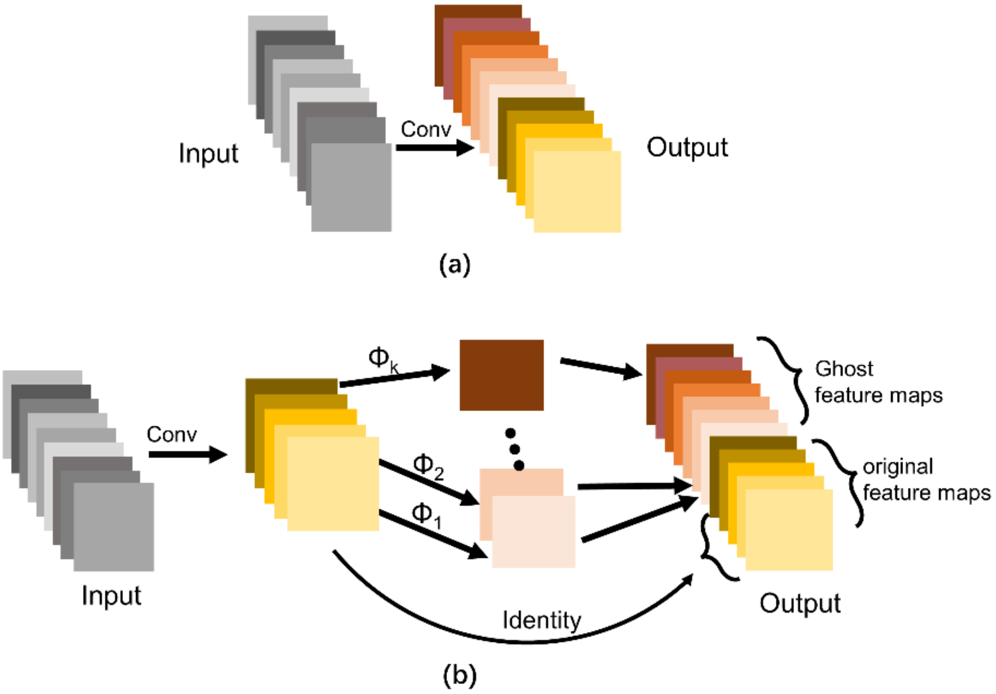


Figure 4. Ordinary convolution and Ghost convolution process. (**a**) Ordinary convolution; (**b**) Ghost convolution.

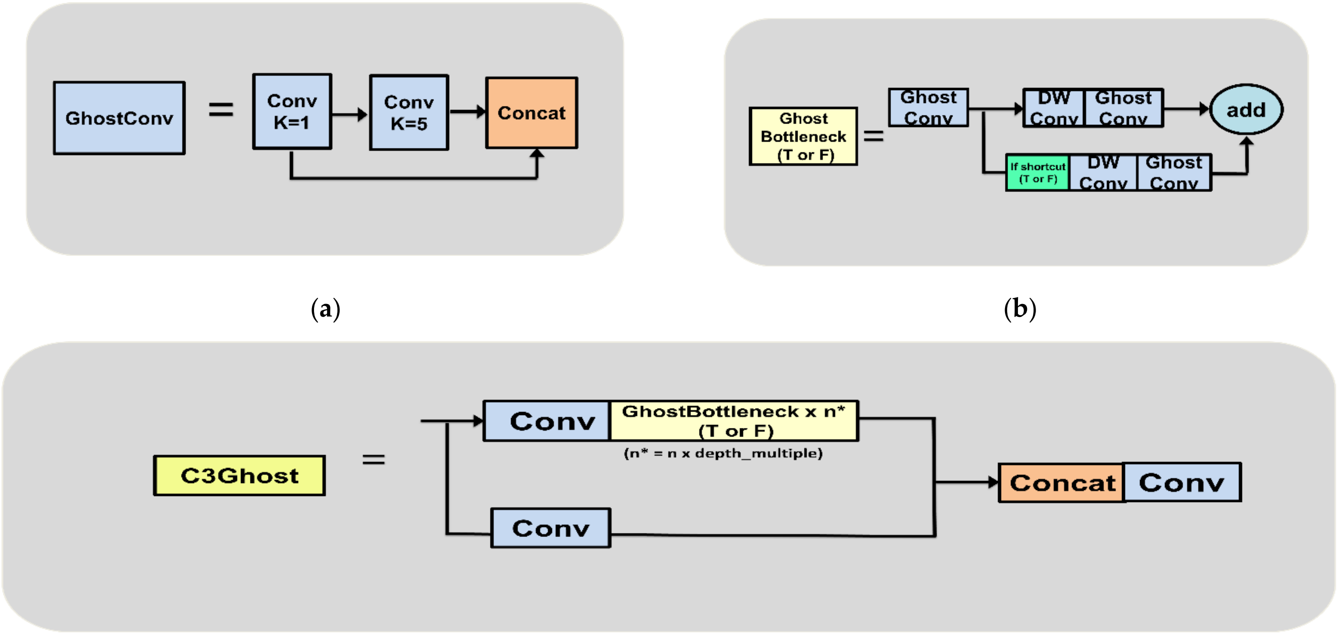
Suppose $X = [x_1, x_2, \ldots x_c,] \in R^{c\times h\times w}$ is the size of the input feature map, $f \in R^{c\times k\times k\times n}$ is the size of the convolution kernel of the layer, and $Y = [y_1, y_2, \ldots y_n,] \in R^{n\times h\times w}$ is the size of the output feature map, where c and n represent the number of channels of the input and output feature maps, h/h' and w/w' represent the height and width of the feature maps, and k represents the size of the convolution kernel. The number of standard convolution output channels in the first step of the host module is m, and the second step includes identity mapping and $m(s-1) = \frac{n}{s}\cdot(s-1)$ linear transformation operations. The size of each operation kernel is $d \times d$, and the theoretical speedup of the Ghost module upgrading ordinary convolution is calculated by the following Equation (1):

$$r_s = \frac{n\cdot w'\cdot h'\cdot c\cdot k\cdot k}{\frac{n}{s}\cdot n\cdot w'\cdot h'\cdot c\cdot k\cdot k + \frac{n}{s}\cdot(s-1)\cdot w'\cdot h'\cdot d\cdot d} \approx \frac{s\cdot c}{s+c-1} \approx s \tag{1}$$

where $d \times d$ is similar in magnitude to $k \times k$, $s \ll c$; then, the theoretical parameter compression ratio is:

$$r_c = \frac{n\cdot c\cdot k\cdot k}{\frac{n}{s}\cdot c\cdot k\cdot k + \frac{n}{s}\cdot(s-1)\cdot d\cdot d} \approx \frac{s\cdot c}{s+c-1} \approx s \tag{2}$$

The streamlined outcome allows for the conclusion that the amount of computation required for Ghost convolution is about s times less than that for normal convolution. Similar to that, s times fewer parameters may also be used. Ghost convolution is a lighter and faster module. Based on this, this research replaced portions of the generic convolutions in YOLOv5 with Ghost convolution. Figure 5 depicts the Ghost construction that was replaced.



(**a**)

(**b**)

(**c**)

Figure 5. Replaced Ghost series modules. (**a**) Ghostconv; (**b**) GhostBottleneck; (**c**) C3Ghost.

3.3.2. Introduction of Coordinate Attention Mechanism

The accuracy of model identification is impacted because the coal and rock photos in this research include many microscopic cracks that are difficult to see and are quickly obstructed by the backdrop. As a result, some minor target information is lost during the detection process. Deep learning has recently included attention mechanisms, the main goal of which is to have the network to focus on the areas that require greater attention. At CVPR2021, Hou et al. proposed a coordinate attention (CA) mechanism that can embed location information into channel attention. By simultaneously encoding features along the horizontal and vertical axes of space, CA can obtain channel information, position information, and spatial dependencies. Consequently, this article increased the model's capability to precisely locate and identify target features with almost no extra computing expense by incorporating the CA method. The structure diagram of the CA mechanism is shown in Figure 6.

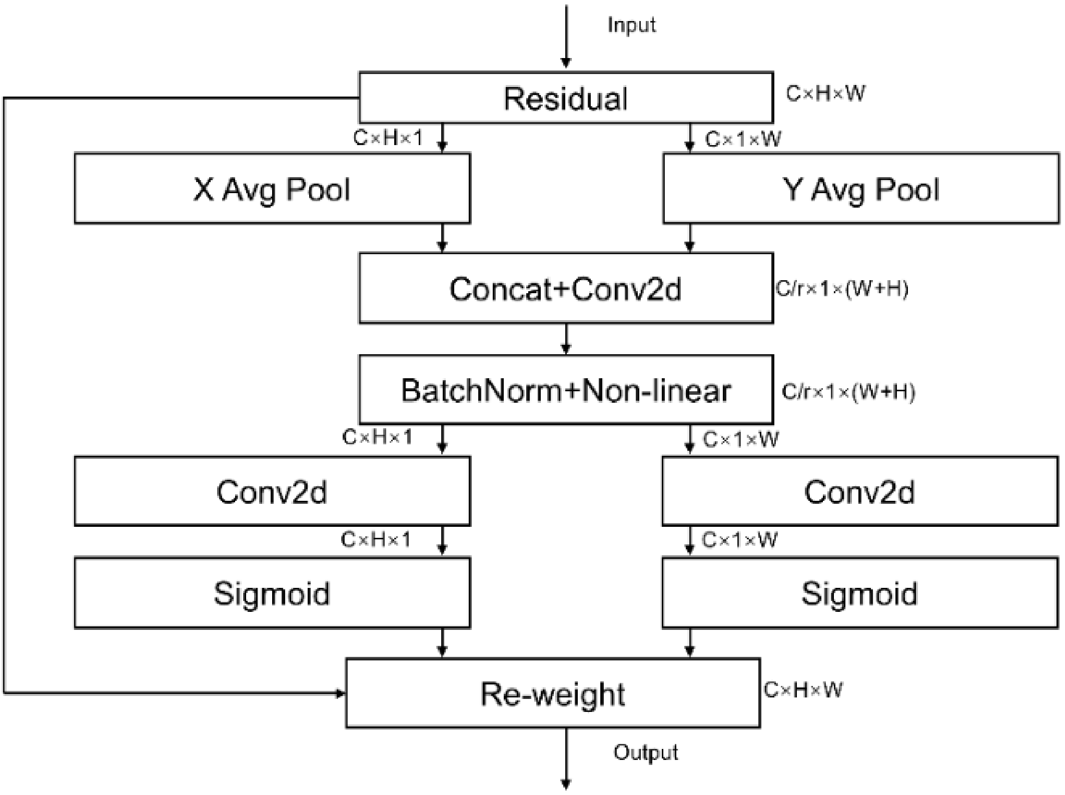


Figure 6. Coordinate attention mechanism.

CA encoded the input feature maps with the size $C \times H \times W$ using pooling kernels with the size $H \times 1$ horizontally and $1 \times W$ vertically for each channel. After encoding, a pair of feature maps Z^h and Z^w demonstrating the orientation perception were generated. When the height is h, the output formula of the cth channel of Z^h and Z^w is as follows:

$$Z_c^h(h) = \frac{1}{W}\sum_{i=0}^{W} X_c(h,i) \tag{3}$$

$$Z_c^w(w) = \frac{1}{H}\sum_{i=0}^{H} X_c(j,w) \tag{4}$$

Then, the feature maps Z^h and Z^w were cascaded, and the F and δ nonlinear activation functions were transformed through 1×1 convolution. The operation formula is as follows:

$$f = \delta\left(F\left[Z^h, Z^w\right]\right) \tag{5}$$

The intermediate feature map $f \in R^{C/r \times 1 \times (H+W)}$ was obtained, and r represents the downsampling ratio.

Next, the intermediate feature map f was decomposed into two separate tensors $f^h \in R^{C/r \times H}$ and $f^w \in R^{C/r \times W}$ along the spatial dimension. Using two 1×1 convolution operations, F_h and F_w, and the sigmoid activation function, f^h and f^w were transformed into the same number of channels as the input feature map. The formula is as follows:

$$g^h = \sigma\left(F_h\left(f^h\right)\right) \tag{6}$$

$$g^w = \sigma(F_w(f^w)) \tag{7}$$

Finally, the attention weight maps g^h and g^w in the two spatial directions were extended and output. The final output formula of the CA module is as follows:

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j) \tag{8}$$

A plug-and-play module that can be installed behind any feature map is the attention mechanism. Adding it to the backbone portion may enable the model to enhance the initial feature extraction of the network and may increase the prediction impact since the features extracted from the backbone component serve as the foundation for further processing. Therefore, this paper embedded the attention mechanism behind the C3Ghost module in the backbone.

3.3.3. Improvement of Loss Function

In YOLOv5, complete intersection over union loss (CIOU_Loss) was used as the frame loss function. The formula is as follows:

$$CIOU\_Loss = 1 - IOU + \frac{\rho^2\left(b^{gt}, b\right)}{c^2} + \alpha v \tag{9}$$

$$IOU \ = \frac{A \cap B}{A \cup B} \tag{10}$$

$$\alpha = \frac{v}{(1 - IOU) + v} \tag{11}$$

$$v = \frac{4}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right) \tag{12}$$

where *IOU* represents the intersection ratio of the predicted box and the real box area. $\frac{\rho^2\left(b^{gt},b\right)}{c^2}$ represents the distance between the predicted frame and the actual frame center point, α is the weight function used to balance the scale, and ν is used to measure the consistency of the aspect ratio.

The width and height of the predicted frame cannot be increased or decreased at the same time during regression because av is not the real difference between the width and height and its confidence; as a result, once it converges to the line-to-line ratio between the width and height of the predicted frame and the real frame, it sometimes prevents the model from effectively optimizing for similarity. The loss function *EIOU_Loss* [41] splits the aspect ratio influence factor αv on the basis of *CIOU_Loss* and calculates the length and width of the predicted frame and the real frame to solve the problem of *CIOU_Loss*. *EIOU_Loss* is calculated by Equation (13).

$$EIOU\_Loss = 1 - IOU + \frac{\rho^2\left(b^{gt}, b\right)}{c^2} + \frac{\rho^2\left(h^{gt}, h\right)}{c_h^2} + \frac{\rho^2\left(w^{gt}, w\right)}{c_w^2} \tag{13}$$

where $\frac{\rho^2\left(h^{gt},h\right)}{c_h^2}$ represents the difference between the height of the predicted box and the real box; $\frac{\rho^2\left(w^{gt},w\right)}{c_w^2}$ represents the difference between the width of the predicted box and the real box.

When there is an edge that is far away, only *EIOU _Loss* computation will become slower without early convergence. This research suggests a novel enhanced loss function, *ECIOU*, which can increase the prediction frame adjustment and accelerate the frame regression rate.

Combining the two loss functions of *CIOU* and *EIOU* is the foundation of *ECIOU*. The prediction frame's aspect ratio is first changed by *CIOU* until it converges to a suitable range, and then each edge is carefully tweaked by *EIOU* until it converges to the right value. *ECIOU_Loss* is calculated by Equation (14). Figure 7 depicts the loss comparison curve for *CIOU_Loss*, *EIOU_Loss*, and *ECIOU_Loss* throughout the training procedure.

$$ECIOU\_Loss = 1 - IOU + \alpha v + \frac{\rho^2(b^{gt}, b)}{c^2} + \frac{\rho^2(h^{gt}, h)}{c_h^2} + \frac{\rho^2(w^{gt}, w)}{c_w^2} \quad (14)$$

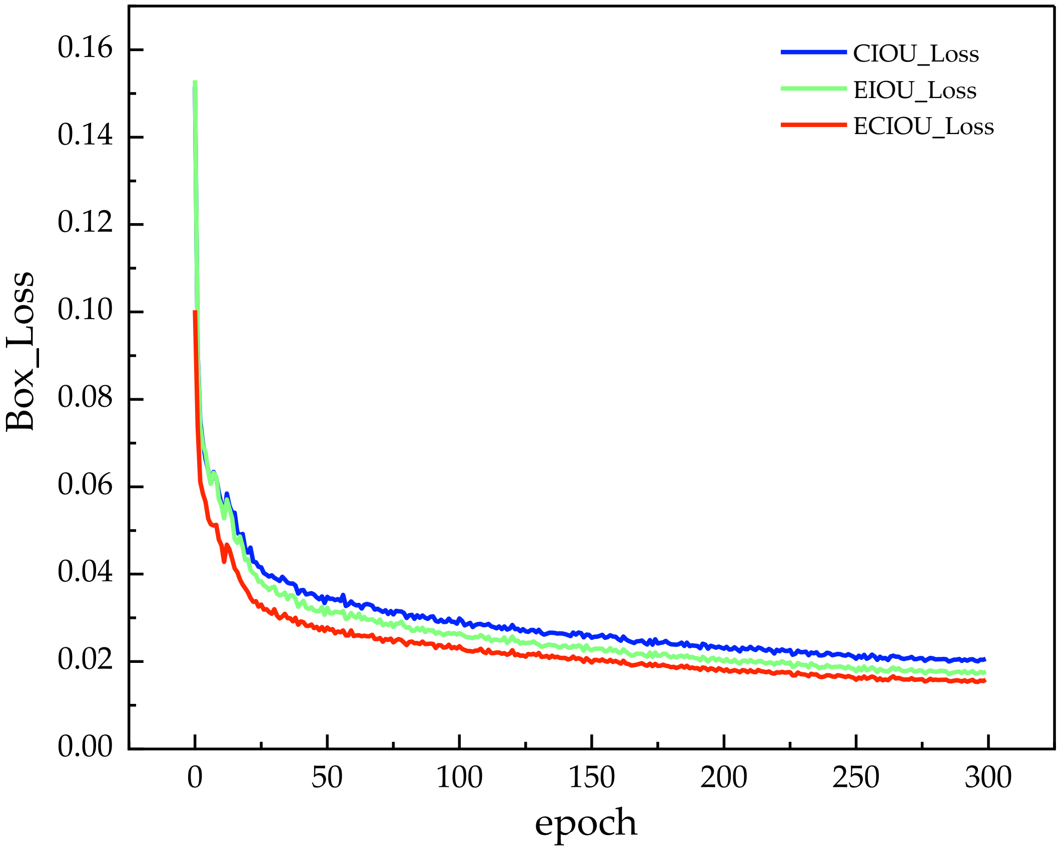


Figure 7. The loss comparison curve for *CIOU_Loss*, *EIOU_Loss*, and ECIOU_Loss.

Figure 7 shows that *ECIOU_Loss* converges the quickest, and the loss value drops to less than 0.03 after 50 rounds. After 300 rounds, the loss values for *ECIOU_Loss*, *EIOU_Loss*, and *CIOU_Loss* are around 0.015, 0.017, and 0.020, respectively. Both *CIOU_Loss* and *EIOU_Loss* are inferior to *ECIOU_Loss*.

3.3.4. Algorithm Flow

The overall flow chart of the algorithm is shown in Figure 8 and is mainly divided into two parts: model training and testing. In the training phase, the model is trained according to the algorithm steps described in the process, and the trained model is obtained and saved. In the testing stage, the trained model is used to detect the test set to obtain the final result.

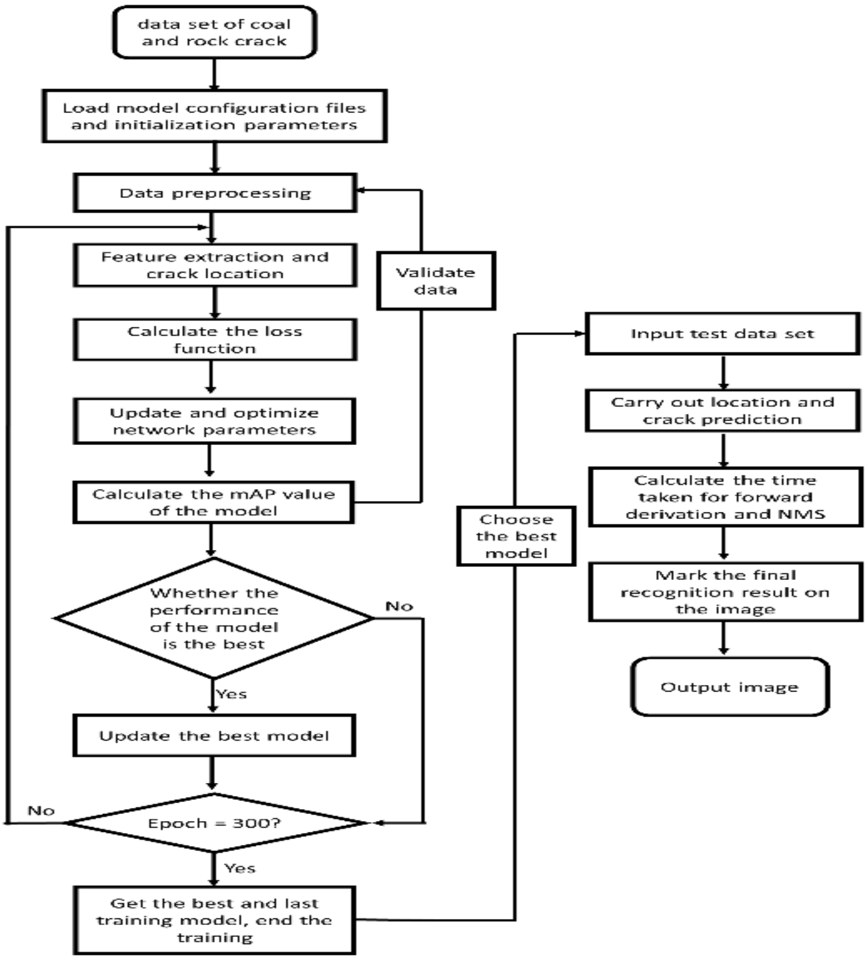

data set of coal and rock crack
Load model configuration files and initialization parameters
Data preprocessing
Feature extraction and crack location
Calculate the loss function
Update and optimize network parameters
Calculate the mAP value of the model
Validate data
Whether the performance of the model is the best
No
Yes
Update the best model
No
Epoch = 300?
Yes
Get the best and last training model, end the training
Choose the best model
Input test data set
Carry out location and crack prediction
Calculate the time taken for forward derivation and NMS
Mark the final recognition result on the image
Output image

Figure 8. Algorithm flow chart.

4. Results

4.1. Experimental Environment

The computer central processing unit (CPU model) used in this experiment was an Intel(R) Core (TM) i7-12700F CPU @ 2.10 GHz, and the running memory was 16 GB. The graphics processor (GPU model) was an NVIDIA GeForce RTX 3060 discrete graphics card, and the video memory size was 12 GB. The 64-bit Windows 10 operating system was used as the software environment, PyCharm was used as the development platform, PyTorch was used as the deep learning framework, Python was the chosen programming language, and the CUDA 11.3 version of the parallel computing framework was used with the CuDNN 8.0 version of the deep neural network acceleration library. The parameter settings are shown in Table 4.

Table 4. Hyperparameter settings.

Hyperparameter	Value
Epochs	300
Imgsz	640
batch_size	16
learning_rate	0.01
Weight_decay	0.0005
Momentum	0.937

4.2. Evaluation Indicators

In this experiment, the model will be thoroughly assessed from the perspectives of model correctness and model complexity. As for the model accuracy measurement index, the precision (P), the recall rate (R), and the mean average precision (mAP) were selected. The calculations for precision, recall, and mAP are shown as Equations (15)–(18). The computation amount (GFLIOP), the number of pictures detected per second (FPS), the model size, and the number of parameters were employed as indications to determine the model complexity as it relates to time and space.

The ratio of the accurately anticipated positive classes to all of the expected positive classes is known as precision. The recall rate is the ratio of the positively anticipated classes that really occur to all of the positively occurring classes. In the formula below, true positives (TP) refer to the number of correctly predicted cracks; false positives (FP) refer to the number of incorrectly predicted cracks; false negatives (FN) refer to the number of non-cracks that are not correctly predicted. mAP is obtained by taking the average of the average precision (average precision) of all of the categories, where AP refers to the area of the curve with the recall rate as the horizontal axis and the precision as the vertical axis.

$$Precision = \frac{TP}{TP + FP} \tag{15}$$

$$Recall = \frac{TP}{TP + FN} \tag{16}$$

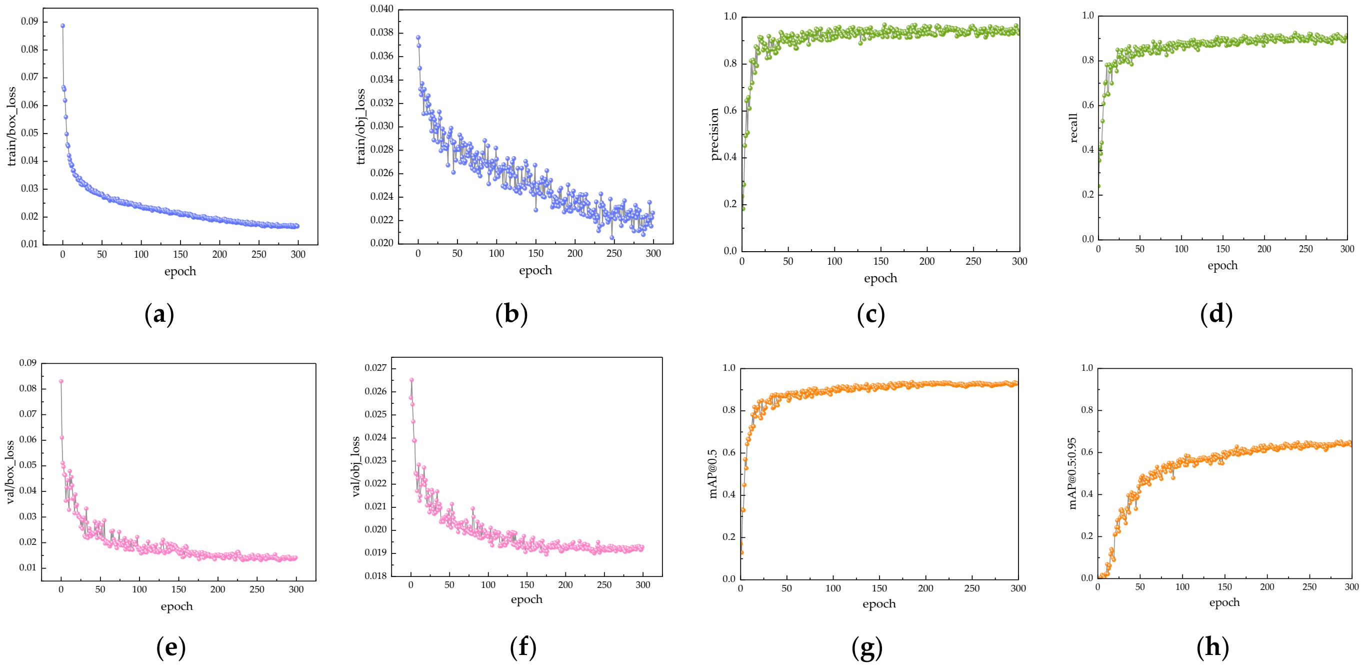
$$AP = \int_0^1 P(R)dR \tag{17}$$

$$mAP = \frac{1}{c}\sum_{i=1}^{c} AP_i \tag{18}$$

4.3. Experimental Results and Analysis

As can be seen in Figure 9, the YOLOv5-G model's parameters are calculated using the log data stored during training. The training loss is minimal, and the result value of the train/box_loss is around 0.015, suggesting that the training outcomes are more accurate; the value of train/obj_loss is around 0.022, indicating that the detection accuracy of the target is high. The values of val/box_loss and val/obj_loss were about 0.014 and 0.0194, respectively, showing that the model fitting effect is good. The precision is around 94%, indicating that the accuracy of detecting the target object is relatively high; the recall rate is about 89%, indicating that the accuracy of finding the correct objects is relatively good. The proportions of mAP@0.5 and mAP@0.5:0.95 are approximately 92 and 65, respectively.

The same dataset and parameters were used to train both the YOLOv5 model and the YOLOv5-G model, and a comparison chart of the two models was created based on the training outcomes, as shown in Figure 10 and in Table 5. In Figure 10, although the recall rate R has dropped by 2.2%, the accuracy rates P, mAP@0.5, and mAP@0.5:0.95 have all improved compared to the previous ones. Specifically, mAP@0.5:0.95 has increased by 5.6%. Due to its higher IOU threshold requirement, mAP@0.5:0.95 is mostly utilized to represent the positioning impact and border regression capability. This demonstrates that improving the model enhances the network's ability to regress to the target boundary. Table 5 shows that YOLOv5-G's parameters, GFLIOP, and size have been decreased, and its detection speed is 1.9 times faster than that of YOLOv5, producing the lightweight effect.



(**a**) (**b**) (**c**) (**d**)

(**e**) (**f**) (**g**) (**h**)

Figure 9. Comparison of model accuracy. (**a**) train/box_loss; (**b**) train/obj_loss; (**c**) precision; (**d**) recall; (**e**) val/box_loss; (**f**) val/obj_loss; (**g**) mAP@0.5; (**h**) mAP@0.5:0.95.

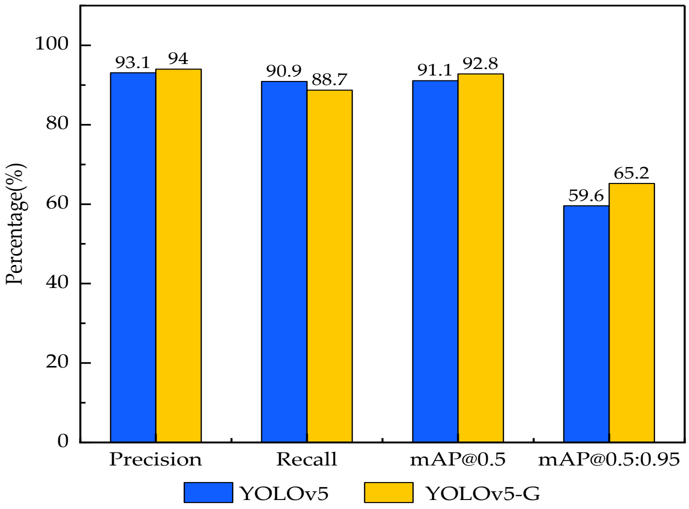


Figure 10. Comparison of model accuracy.

Table 5. Comparison of model complexity.

Model	Params	GFLIOP	Size	FPS
YOLOv5	7015,519	15.8	14.5	55
YOLOv5-G	3746,583	8.2	8.0	103

4.4. Ablation Experiments

We created five sets of ablation experiments based on YOLOv5, each of which employed identical hyperparameters and training methods in order to further investigate the usefulness of each upgraded approach and to confirm the detection performance of the algorithm suggested in this paper. In order to lighten the weight of the model, Model 1 substituted Ghost convolution for the original network's regular convolution. Model 2 augmented the backbone network with an attention mechanism. Model 3 represented the adjustment of the loss function, whereas Model 4 reflected the advancement of Model 1 by including an attention mechanism. YOLOv5-G used Model 1, Model 2, and Model 3 at the same time. The ablation experiments are shown in Table 6.

Table 6. Ablation experiments.

Model	Ghost	CA	ECIOU	mAP	Params	GFLIOP	FPS	Size
YOLOv5	-	-	-	91.1%	7,015,519	15.8	55	14.5
1	√ [1]	-	-	87.5%	3,678,423	8.1	123	7.9
2	-	√	-	95.2%	7,028,855	15.9	51	14.6
3	-	-	√	91.6%	7,015,519	15.8	61	14.5
4	√	√	-	92.4%	3,746,583	8.2	98	8.0
YOLOv5-G	√	√	√	92.8%	3,746,583	8.2	103	8.0

[1] stands for importing module.

According to the statistics in Table 6, the number of parameters, GLOPS, and FPS all dramatically dropped when the Ghost module was added to Model 1, although the mAP fell by 3.6%. This demonstrated that the Ghost module produced successful lightweight model results. Model 2's salient feature extraction of the crack target is clearer, with nearly no increase in the number of parameters or calculations with the addition of the attention mechanism, and the mAP is increased by 4.1 percentage points to 95.2%. The mAP was raised by 0.5% when the loss function of Model 3 was changed, and the speed was 5% quicker than that of YOLOv5, demonstrating that ECIOU_Loss was successful in enhancing the target frame's capacity to fit. Model 4 adds CA on the basis of being lightweight, and the map was increased by 1.3%. When YOLOv5-G incorporated these three enhancements into the model, the mAP was 92.8% higher than it was with the prior model. The detection speed was 103 frames per second; the number of parameters, GFLOPS, and model weights were all halved; and the method had been enhanced in terms of accuracy, speed, and complexity.

4.5. Comparative Experiments

In order to comprehensively evaluate the performance of the YOLOv5-G model, this paper selected the two-stage Faster-RCNN algorithm and the one-stage SDD YOLOv4 and YOLOv4-tiny algorithms and used the same training techniques and parameter settings to conduct experiments on a self-made dataset. The comparative experiments are shown in Table 7.

Table 7. Comparative experiments.

Model	Backbone	mAP	Size	FPS
Faster-RCNN	ResNet	84.3%	108.3	6
SSD	VGG16	82.1%	92	27
YOLOv4	DarkNet53	88.7%	244	34
YOLOv4-tiny	DarkNet53	78.6%	22.6	118
YOLOv5-G	GhostNet	92.8%	8.0	103

Table 7 shows that the mAP of the YOLOv5-G algorithm is 92.8%, which is 8.5%, 10.7%, 4.1%, and 14.2% greater than the mAPs of the Faster-RCNN, SSD, YOLOv4, and YOLOv4-tiny algorithms, respectively. YOLOv5-G has a higher average accuracy than other algorithms. YOLOv5-G has a strong generalization ability and can learn target generalization information. It has certain universality. When it is migrated, the model has high robustness, so YOLO is much more accurate than other target detection methods. The detection rate of YOLOv5-G is higher than that of the Faster-RCNN, SSD, and YOLOv4 algorithms in terms of detection speed FPS. YOLOv5-G is 3.8 times and 3.1 times quicker than the SSD and YOLOv4 algorithms, respectively, and has a detection rate that is 17.2 times faster than that of Faster-RCNN. Compared to the YOLOv4-tiny algorithm, YOLOv5-G is 1.2 ms slower, but in return, the mAP is greatly improved by 14.2%. In terms of model size, YOLOv5-G is only 8 MB, which is the lightest among the above models and is more suitable for porting to embedded devices.

5. Conclusions

This study suggested an enhanced target recognition approach for the YOLOv5-G model, with a focus on the issue of coal and rock dynamic catastrophes, by taking into account the real demands of engineering in situations in which there is low model detection accuracy, sluggish speed, and poor portability. Initially, the model employed the Ghost module to lighten the network as a whole, significantly reducing the number of model parameters and speeding up the network. Then, the CA attention mechanism was embedded in the C3Ghost module of the backbone network to improve the model accuracy and learning ability; Finally, through the ECIOU_Loss function, the regression positioning ability of the model was improved. According to experiments, the mAP of YOLOv5-G is up 1.7% from the original model, the detection speed is 1.9 times quicker, and the model weight is just 8 MB. The strong generalization, high detection accuracy, and quick speed of the model make it useful for future hardware replacement and also indicate some practical value in the area of real-time applications.

Given the limits of the detection technique provided in this research, we will continue to develop the model as well as expand its function to extract physical characteristics such as the crack length, width, and propagation speed. Furthermore, the detection network design suggested in this research can also be utilized for other fields, such as for the detection of industrial equipment defects, building damage, and road and bridge cracks.

Author Contributions: Conceptualization, X.C. (Xinlin Chen); methodology, X.C. (Xinlin Chen) and J.S.; software, X.C. (Xinlin Chen); validation, X.C. (Xinlin Chen) and J.S.; formal analysis, X.C. (Xinlin Chen); investigation, Q.L. and X.C. (Xinlin Chen); resources, Q.L.; data curation, X.C. (Xinlin Chen); writing—original draft preparation, X.C. (Xinlin Chen); writing—review and editing, X.C. (Xuanlai Chen); visualization, X.C. (Xinlin Chen); supervision, Q.L. and X.C. (Xuanlai Chen); project administration, X.C. (Xinlin Chen); funding acquisition, Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of Shanxi Province (grant number 201901D111071) and the UWA-China Joint Scholarships and the China Scholarship Council (grant number 202106930014).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding authors upon request.

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

References

1. Zhou, H.; Meng, F.; Zhang, C.; Hu, D.; Yang, F.; Lu, J. Analysis of rockburst mechanisms induced by structural planes in deep tunnels. *Bull. Eng. Geol. Environ.* **2014**, *74*, 1435–1451. [CrossRef]
2. Xie, H.; Zhou, H.; Xue, D.; Wang, H.; Zhu, R.; Gao, F. Research and consideration on deep coal mining and critical mining depth. *J. China Coal Soc.* **2012**, *37*, 535–542.
3. Qian, M. On sustainable coal mining in China. *J. China Coal Soc.* **2010**, *35*, 529–534.
4. Yu, M.-H. Advances in strength theories for materials under complex stress state in the 20th Century. *Appl. Mech. Rev.* **2002**, *55*, 169–218.
5. Li, Y.-P.; Chen, L.-Z.; Wang, Y.-H. Experimental research on pre-cracked marble under compression. *Int. J. Solids Struct.* **2005**, *42*, 2505–2516. [CrossRef]
6. Hao, X.; Du, W.; Zhao, Y.; Sun, Z.; Zhang, Q.; Wang, S.; Qiao, H. Dynamic tensile behaviour and crack propagation of coal under coupled static-dynamic loading. *Int. J. Min. Sci. Technol.* **2020**, *30*, 659–668. [CrossRef]
7. Ai, D.; Zhao, Y.; Wang, Q.; Li, C. Crack propagation and dynamic properties of coal under SHPB impact loading: Experimental investigation and numerical simulation. *Theor. Appl. Fract. Mech.* **2020**, *105*, 102393. [CrossRef]

8. Li, F.; Dong, X.; Wang, Y.; Liu, H.; Chen, C.; Zhao, X.; Wang, G.-D. The Dynamic Response and Failure Model of Thin Plate Rock Mass under Impact Load. *Shock Vib.* **2021**, *2021*, 9998558. [CrossRef]
9. Nguyen, T.L.; Hall, S.A.; Vacher, P.; Viggiani, G. Fracture mechanisms in soft rock: Identification and quantification of evolving displacement discontinuities by extended digital image correlation. *Tectonophysics* **2011**, *503*, 117–128. [CrossRef]
10. Li, Z.; Zhang, G. Fracture Segmentation Method Based on Contour Evolution and Gradient Direction Consistency in Sequence of Coal Rock CT Images. *Math. Probl. Eng.* **2019**, *2019*, 2980747. [CrossRef]
11. Chen, Y.; Zhang, H. An Improved C-V Model and Application to the Coal Rock Mesocrack Images. *Geofluids* **2020**, *2020*, 1–11.
12. Li, C.; Ai, D. Automatic crack detection method for loaded coal in vibration failure process. *PLoS ONE* **2017**, *12*, e0185750. [CrossRef] [PubMed]
13. Miao, S.; Pan, P.-Z.; Wu, Z.; Li, S.; Zhao, S. Fracture analysis of sandstone with a single filled flaw under uniaxial compression. *Eng. Fract. Mech.* **2018**, *204*, 319–343. [CrossRef]
14. Shinde, P.P.; Shah, S. A review of machine learning and deep learning applications. In Proceedings of the 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 16–18 August 2018; pp. 1–6.
15. Sharma, N.; Sharma, R.; Jindal, N. Machine learning and deep learning applications—A vision. *Glob. Transit. Proc.* **2021**, *2*, 24–28. [CrossRef]
16. Pathak, A.R.; Pandey, M.; Rautaray, S. Application of deep learning for object detection. *Procedia Comput. Sci.* **2018**, *132*, 1706–1717. [CrossRef]
17. Wu, Q.; Liu, Y.; Li, Q.; Jin, S.; Li, F. The application of deep learning in computer vision. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 6522–6527.
18. Yang, J.; Li, J. Application of deep convolution neural network. In Proceedings of the 2017 14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 15–17 December 2017; pp. 229–232.
19. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
20. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
21. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
22. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot Multibox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany; pp. 21–37.
23. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
24. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
25. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
26. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
27. Cui, X.; Wang, Q.; Dai, J.; Zhang, R.; Li, S. Intelligent recognition of erosion damage to concrete based on improved YOLO-v3. *Mater. Lett.* **2021**, *302*, 130363. [CrossRef]
28. Zhao, P.; Luo, Z.; Li, J.; Liu, Y.; Zhang, B. Machine Learning Sorting Method of Bauxite Based on SE-Enhanced Network. *Appl. Sci.* **2022**, *12*, 7178. [CrossRef]
29. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
30. Yue, X.; Wang, Q.; He, L.; Li, Y.; Tang, D. Research on Tiny Target Detection Technology of Fabric Defects Based on Improved YOLO. *Appl. Sci.* **2022**, *12*, 6823. [CrossRef]
31. Liu, X.; Li, G.; Chen, W.; Liu, B.; Chen, M.; Lu, S. Detection of Dense Citrus Fruits by Combining Coordinated Attention and Cross-Scale Connection with Weighted Feature Fusion. *Appl. Sci.* **2022**, *12*, 6600. [CrossRef]
32. Yan, B.; Fan, P.; Lei, X.; Liu, Z.; Yang, F. A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5. *Remote Sens.* **2021**, *13*, 1619. [CrossRef]
33. Li, X.; Wang, C.; Ju, H.; Li, Z. Surface Defect Detection Model for Aero-Engine Components Based on Improved YOLOv5. *Appl. Sci.* **2022**, *12*, 7235. [CrossRef]
34. Mohiyuddin, A.; Basharat, A.; Ghani, U.; Peter, V.; Abbas, S.; Naeem, O.B.; Rizwan, M. Breast Tumor Detection and Classification in Mammogram Images Using Modified YOLOv5 Network. *Comput. Math. Methods Med.* **2022**, *2022*, 1359019. [CrossRef]
35. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1580–1589.
36. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.

37. Liu, X.; Zhu, W.; Zhang, P.; Li, L. Failure in rock with intersecting rough joints under uniaxial compression. *Int. J. Rock Mech. Min. Sci.* **2021**, *146*, 104832. [CrossRef]
38. Lin, T. LabelImg. 2015. Available online: https://github.com/tzutalin/labelImg (accessed on 14 September 2016).
39. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 516–520.
40. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 12993–13000.
41. Zhang, Y.-F.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and efficient IOU loss for accurate bounding box regression. *arXiv* **2021**, arXiv:2101.08158. [CrossRef]