fire

MDPI

Article

RepVGG-YOLOv7: A Modified YOLOv7 for Fire Smoke Detection

Xin Chen [1,*], Yipeng Xue [1], Qingshan Hou [1], Yan Fu [2] and Yaolin Zhu [1]

[1] School of Electronics and Information, Xi'an Polytechnic University, Xi'an 710048, China; xyp_xpu@163.com (Y.X.); hqs2759991833@gmail.com (Q.H.); fz_zyl@126.com (Y.Z.)
[2] Shaanxi Architectural Design Research Institute Co., Ltd., Xi'an 710018, China; fuyan029@outlook.com
* Correspondence: chenxin@xpu.edu.cn

Abstract: To further improve the detection of smoke and small target smoke in complex backgrounds, a novel smoke detection model called RepVGG-YOLOv7 is proposed in this paper. Firstly, the ECA attention mechanism and SIoU loss function are applied to the YOLOv7 network. The network effectively extracts the feature information of small targets and targets in complex backgrounds. Also, it makes the convergence of the loss function more stable and improves the regression accuracy. Secondly, RepVGG is added to the YOLOv7 backbone network to enhance the ability of the model to extract features in the training phase, while achieving lossless compression of the model in the inference phase. Finally, an improved non-maximal suppression algorithm is used to improve the detection in the case of dense smoke. Numerical experiments show that the detection accuracy of the proposed algorithm can reach about 95.1%, which contributes to smoke detection in complex backgrounds and small target smoke.

Keywords: smoke detection; target detection; YOLOv7; RepVGG

check for updates

Citation: Chen, X.; Xue, Y.; Hou, Q.; Fu, Y.; Zhu, Y. RepVGG-YOLOv7: A Modified YOLOv7 for Fire Smoke Detection. *Fire* **2023**, *6*, 383. https://doi.org/10.3390/fire6100383

Academic Editor: Washington Rocha, António Vieira and Marcos Francos

Received: 7 September 2023
Revised: 1 October 2023
Accepted: 2 October 2023
Published: 7 October 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

As one of the most harmful natural disasters today, fire causes tens of billions of economic and property losses to our world every year, and seriously threatens the lives and properties of people [1]. After entering the 21st century, with the advancement of urbanization and industrialization in the world, buildings such as factories, residential areas, and schools have become increasingly denser, and the risk of fires has also increased. Once a fire breaks out, it will cause unrecoverable economic and environmental losses. As an early phenomenon of fire, smoke can be more observable than flame [2]. Therefore, exploiting this feature of fire to detect early smoke can give timely warnings of fire. Nowadays, with the advancement of science and technology, the fire alarm system is also rapidly changing every day. From the initial smoke alarm to the smoke detection based on ordinary cameras, technological innovation has brought new directions and ideas to fire smoke detection. The means of applying new technologies to smoke detection are innovative and play a vital role for the prevention and control of fires [3].

Existing smoke detection algorithms are mainly classified into two categories: traditional smoke detection algorithms and deep learning-based smoke detection algorithms [4]. The current traditional smoke detection algorithms are based on extracting various features of smoke, such as color, texture, motion and shape irregularities. The traditional feature-based smoke detection is mainly based on framing the candidate region of smoke, then extracting the features of smoke and finally classifying and detecting smoke [5]. Smoke detection consists of two tasks: smoke identification and localization. The core step of smoke identification is to extract and classify smoke features in images, and the core step of smoke localization is to locate the smoke regions that appear in images and provide marks [6]. Based on this, many researchers have proposed smoke detection methods; Zhou et al. [7] proposed a smoke detection algorithm combining dynamic and static features, which can

have a fast detection of the presence of smoke, but it has a high false alarm rate and cannot distinguish between real smoke and fake smoke. Gomez-Rodriguez et al. [8] used a wavelet decomposition and optical flow method for smoke detection in wildfire, which is suitable for the extraction of multiple smoke features, but its drawback is the large computational cost. Filonenko et al. [9] proposed a smoke detection method based on smoke edge roughness and edge density. Liu et al. [10] proposed a video smoke detection algorithm based on contrast, wavelet analysis and color segmentation, and in order to use shape information, they proposed an RGB contrast image and shape which is constrained to improve the smoke detection algorithm. In the initial research of deep learning algorithms, some researchers and scholars started from deep learning networks and kept digging deeper into the layers of the network structure, and were devoted to extracting more expressive smoke features, from CNN at the beginning to the one-stage algorithm and two-stage algorithm. Nowadays, the network model is getting bigger and bigger; both the feature extraction ability and the generalization of the algorithm are getting stronger. Frizzi et al. [11] also proposed a CNN-based algorithm for the amount of smoke detection, which is based on the technical support of early target detection, by using LeNet5 [12] to extract the smoke features in the image, and using the sliding window method to detect the smoke image after chunking. Although this method adds deeper deep convolution to the original CNN framework, the extra deep convolution leads to problems such as high computational load and slow detection speed. After applying the target detection algorithm, based on the CNN framework, directly to the smoke detection task, the smoke detection algorithm began an epoch-making development, and researchers began to make improvements to the existing target detection algorithms, in which the main research objectives are the one-stage algorithms and two-stage algorithms. One-stage algorithms are characterized by high speed, simple models and easy deployment, and their representative algorithms are the YOLO series and SSD series [13,14]. For example, Saponara et al. [15] designed a real-time fire and smoke detection algorithm based on YOLOv2 and met the requirements of embedded platforms. Chen et al. [16] proposed a smoke recognition algorithm based on YOLOv5 combined with a Gaussian mixed background modeling algorithm, which effectively reduced the interference of objects in the background for smoke detection. Wang et al. [17] proposed an improved YOLOv5-based smoke detection model by adding the dynamic anchor frame mechanism, channel attention and spatial attention to improve the detection performance of the algorithm. Yazan Al-Smadi et al. [18] proposed a new framework that can reduce the sensitivity of various YOLO detection models and also compared this with the performance of different YOLO detection models. The fast speed of the single-stage algorithm has led to a significant increase in the detection speed of the smoke detection algorithm, but the detection accuracy is not high. The two-stage algorithm, on the other hand, has an incremental improvement in algorithm accuracy, but it requires a greater computational cost. Yuan et al. [19] designed a new multi-scale convolutional structure, DMCNN, which can fully extract multi-scale smoke features and improve the accuracy of smoke detection. Zhang et al. [20] used the Faster R-CNN target detection network to obtain specific coordinate information of smoke in smoke images and used rectangular boxes to frame the smoke, to achieve the recognition of fire smoke in forests. Gagliardi et al. [21] proposed a method based on Kalman filtering and CNN for real-time video smoke detection, which automatically selects smoke regions in images by moving the generated object bounding box. Lin et al. [22] developed a joint detection framework based on Faster-RCNN and 3D-CNN, which achieves smoke target localization with static spatial information by Faster R-CNN, and then uses 3D-CNN combined with dynamic spatio-temporal information to achieve smoke identification.

In summary, with the development of deep learning technology, significant progress has been made in using deep learning technology for smoke detection, but there are still severe open challenges. On the one hand, there is no complete dataset for the current deep learning-based video smoke detection, and the existing fire smoke dataset types and scenes are simple, leading to low generalization of the trained algorithms. On the other hand, increasing the detection speed of fire smoke will reduce the accuracy of smoke

recognition. When the recognition accuracy is improved, it will result in high model complexity, high computational cost and low portability. What is more, it is difficult to detect fire smoke quickly and accurately in the face of complex scenes and environments with many interference factors. In particular, the detection rate for small target smoke or obscured smoke needs to be improved. Therefore, the main contributions and novelties of this paper are reflected in the following aspects:

1. New fire smoke datasets: The current commonly used public fire smoke dataset has a relatively single scene and a simple smoke type, and is of low resolution. This study, therefore, builds a multi-scene and multi-type smoke dataset with a large amount of negative sample data.
2. RepVGG-YOLOv7 model: The principle is to achieve complex training and simple inference through the RepVGG network, which is fused into the YOLOv7 backbone network to achieve several results, including lossless compression of the model, a reduction in the number of model parameters, and improvement in model inference speed and performance of the fire smoke detection model.
3. Improved loss function: The coordinate loss function in the original YOLOv7 network is improved by the SIoU loss function. The SIoU regression loss function re-describes the distance through the angle cost. With the increase in the angle cost, the loss function can be more fully expressed and the probability of the penalty term being zero is reduced at the same time, which makes the convergence of the loss function more stable and improves the regression accuracy to reduce the prediction error.

2. Materials and Methods

2.1. Dataset Acquisition and Preprocessing

2.1.1. Image Acquisition

At present, the widely used public fire smoke datasets have some disadvantages, e.g., single scene, simple smoke type and low resolution. Moreover, the datasets are mainly single-frame pictures, almost all of which are smoke targets and lack fire smoke image data in real environments. Therefore, this study builds a multi-scene, multi-type smoke dataset with many negative samples through network resources and existing public datasets. The dataset contains 9005 photos, including 6605 smoke photos and 2400 non-smoke photos; the dataset also contains 45 videos of various scenes, including 35 smoke videos and 10 non-smoke videos. The negative sample information in the dataset mainly provides fog, lights, white clouds and dark clouds. Several typical samples in the dataset are shown in Figure 1.

Figure 1. Typical samples of the smoke dataset.

2.1.2. Image Preprocessing

In deep learning, many data samples are usually required for model training to reduce over-fitting problems. During the training phase of the model, the more sufficient and comprehensive the type of data collected and processed, the better the recognition effect of the trained model and the better the generalization ability of the model would be. Therefore, in the early stage of model training, the number of samples is enlarged by appropriate data enhancement techniques. The data enhancement technique used in this study includes morphological operations such as translation, saturation adjustment, angle adjustment and image flipping up and down. At the same time, the Mosaic data enhancement method is used on the input side of the algorithm to perform random scaling, random cropping and random layout stitching of the four images. This works by applying a uniform colour space transformation to the image to change the hue, exposure and saturation of the image, which can achieve the effect of reducing the over-fitting of the network and improving the generalization ability of the model. After the above operations, the input of the model is shown in Figure 2.

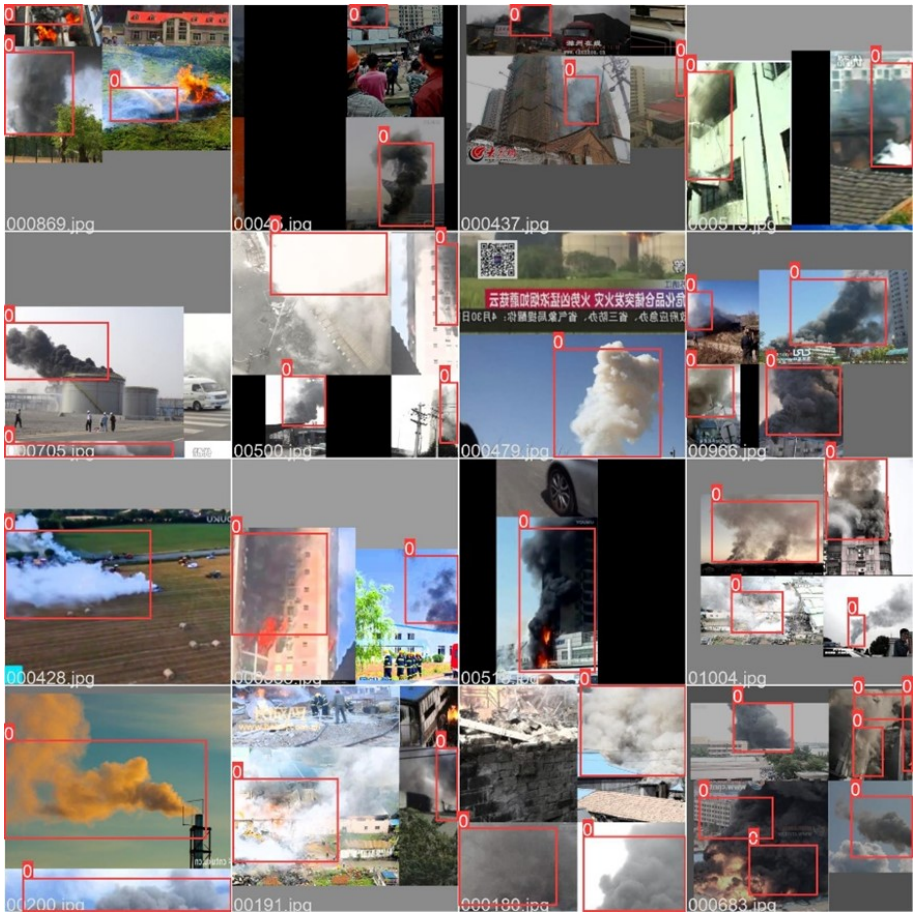


Figure 2. Data enhancement renderings.

In addition to the regularization techniques mentioned above, this paper uses standard methods such as early stopping, weight regularization, batch normalization and dropout during the training phase to prevent model over-fitting.

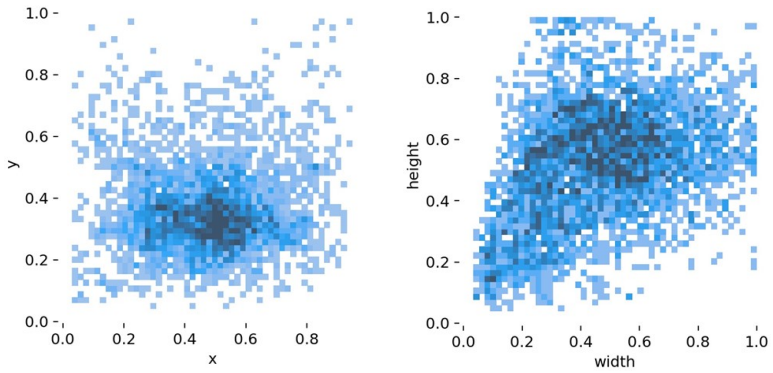
2.1.3. Image Database and Label Database

The dataset used in this paper was produced according to the VOC format by using LabelImg software, with the labeled category of smoke, and then the dataset was divided into three sets: training set, validation set and test set. There is a total of 9005 samples in

the experiment. The number of specific experimental divisions is shown in Table 1, and the distribution of labels in the dataset is shown in Figure 3.

Table 1. Experimental data.

Dataset	Smoke Image	Non-Smoke Image	Total
training set	4175	1000	5175
validation set	1230	700	1930
test set	1200	700	1900
total	6605	2400	9005



(**a**) Label location (**b**) Label size

Figure 3. Label distribution.

Figure 3a illustrates the label location. It is shown that the abscissa x is the ratio of the abscissa of the label center to the image width, and the ordinate y is the ratio of the abscissa of the label center to the image height. From the Figure 3a, we can see that the data are widely distributed and concentrated in the middle of the image. In Figure 3b, it can be seen that the abscissa width is the ratio of the label width to the image width, and the ordinate height is the ratio of the label height to the image height. The dataset contains data of various sizes, which is dominated by the more realistic small and medium target data.

2.2. Proposed Method

The detection framework diagram of the proposed method in this paper is shown in Figure 4. Firstly, the real labeled frames based on the smoke dataset can generate anchor frames of different sizes through the clustering algorithm, so that the initial anchor frame size of the model can accurately match the smoke target size. Secondly, the YOLOv7 network structure is improved by adding the RepVGG structure and the ECA attention mechanism to the backbone network. The basic idea of the RepVGG structure is the complex training and simple reasoning, so a multi-branch network structure like ResNet is used in the training phase of the network to enhance the feature extraction ability of the model. In the inference phase of the model, a structural reparameterization technology is used to compress the model and improve the detection speed in the inference phase of the model, which works by fusing the multi-branch network structure into a single convolutional structure, and it can improve the fire smoke detection performance to a certain extent. The ECA attention module uses 1×1 convolution to capture information between different channels to avoid channel dimension shrinking and reduce the number of parameters when channel attention information is learning. At the same time, it can make the network pay more attention to the target to be detected and improve the detection accuracy. What is more, the improved loss function is exploited to speed up the convergence of the model. Finally, the Soft-NMS method is used to improve the non-maximum value suppression process, which mainly reduces the missed detection and false detection problems

in the case of dense smoke targets and occlusions, and makes the model detection results more accurate.

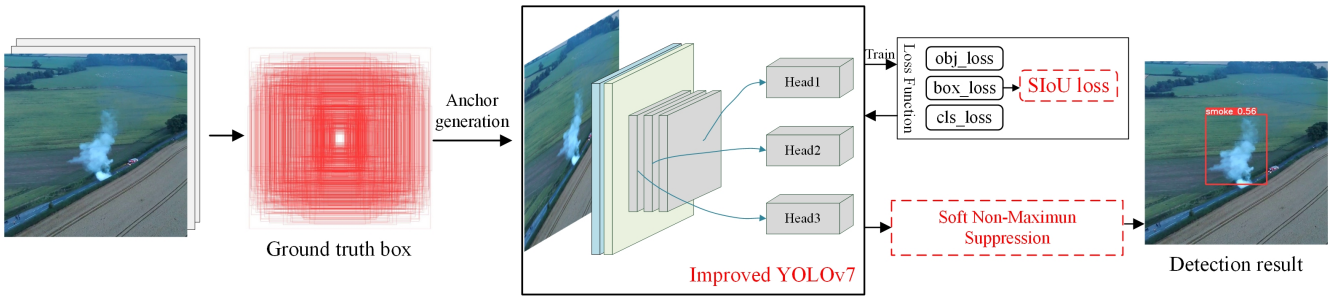


Figure 4. Detection framework of the proposed method.

2.3. Backbone Network

YOLOv7 is the latest model in the YOLO series, which outperforms most target detection networks in terms of speed as well as accuracy in the detection range of 5–160 FPS. There are currently three main models, YOLOv7-tiny, YOLOv7 and YOLOv7-W6. Specifically, strategies such as the Extended Efficient Long-range Attention Network (E-ELAN), the modeling scaling for Concatenation-Based models, convolution reparameterization and others are proposed in the overall architecture [23,24]. As shown in Figure 5, the YOLOv7 network consists of four parts: input, backbone, neck and head.

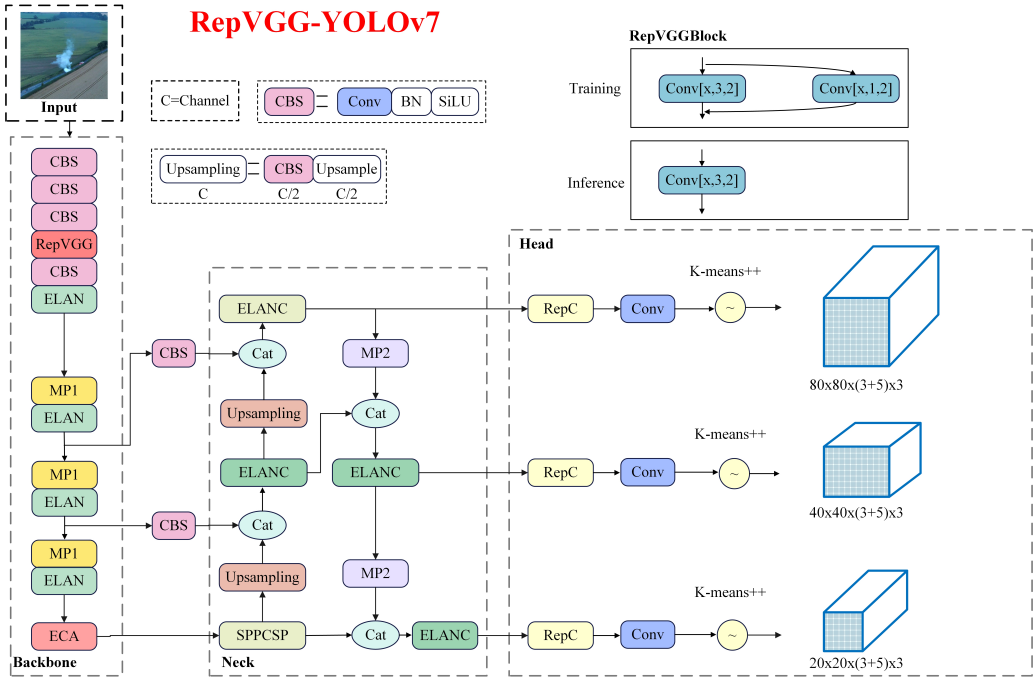


Figure 5. Improved YOLOv7 network.

Firstly, the input performs data enhancement operations on the image, such as Mosaic, random cropping and scaling to avoid over-fitting, and scales the input image to a uniform pixel size to meet the input requirements of the backbone network while the operations are being performed. The backbone mainly includes the ELAN module, the CBS module, consisting of regular convolution, batch normalization and activation functions, and the MP module, consisting of the maxpooling and CBS modules, which differs from the traditional CNN network in that the activation function of the Leaky ReLU is replaced by the SiLU. The ELAN module can control the shortest and longest gradient paths to learn more diverse features by directing blocks of computation to different feature groups, and improve the

learning capability of the network without destroying the original gradient path. The ELAN module is also composed of several CBS modules. The input feature map does not change the size of the feature map after passing through the ELAN module, but only the number of final output channels. The improved network adds a RepVGG module between the third and fourth CBS modules of the backbone network. An ECA attention mechanism is added between the original backbone network and the neck region. The improved network can make the model pay more attention to the valuable features in the input image samples, and it can effectively extract feature information and improve detection accuracy for small targets and targets with complex backgrounds. The neck is composed of the Path Aggregation Network (PAN) and the Feature Pyramid Network (FPN). The channel dimensions change from 1024 to 512 after the 32 times downsampling feature diagram of the backbone network output passes through the SPP CSP module. Then, feature maps are fused according to a top-down strategy and a bottom-up strategy, and the PA-FPN structure effectively infuses the different levels of feature mapping. Compared with YOLOv5, YOLOv7 replaces the CSP module with the ELANC module, and the downsampling becomes the MP2 layer [25]. After the PA-FPN network, the output of the network is three layers of feature maps of different sizes. Finally, the network outputs prediction results through the RepC and Conv modules in the head.

2.3.1. RepVGG

The RepVGG network realizes the possibility of complex training and simple inference. Based on this concept, the network structure is integrated into the backbone network of YOLOv7 to improve the performance of the fire smoke detection model. RepVGG proposes to utilize multi-branch structures like ResNet in the training-time model and converts the multi-branch structure into a VGG-style planar structure in the inference time model using reparameterization techniques [26]. The RepVGG network has two advantages. The reason why the multi-branch structure used in the training time model is better than extracting features is that the idea of residual branching in ResNet has been proved by many network models to have better detection performance. In the inference phase, the model is fused with a more computationally efficient 3×3 convolutional structure through the reparameterization technique, which results in lossless compression of the model, a reduction in the number of model parameters and an increase in the speed of model inference.

Reparameterization mainly consists of two steps: convolution layer and batch normalization layer (BN) fusion, and convolutional branch fusion:

Step 1: fusion of convolutional layer and BN layer, convolutional layer equation:

$$Conv(x) = B + W(x) \tag{1}$$

BN layer equation:

$$BN(X) = \gamma * \frac{(x - mean)}{\sqrt{var}} + \beta \tag{2}$$

incorporate the convolutional layer into the BN layer:

$$BN(Cnov(x)) = \gamma * \frac{W(x) + B - mean}{\sqrt{var}} + \beta \tag{3}$$

simplify the above formula:

$$BN(Cnov(x)) = \frac{W(x) * \gamma}{\sqrt{var}} + [\frac{(B - mean) * \gamma}{\sqrt{var}} + \beta] \tag{4}$$

get:

$$\begin{cases} W_{fised} = \dfrac{W * \gamma}{\sqrt{var}} \\ B_{fised} = \dfrac{(B - mean) * \gamma}{\sqrt{var}} + \beta \end{cases} \tag{5}$$

The fusion result can be shown in the Equation (6):

$$BN(X) = \gamma * \frac{(x - mean)}{\sqrt{var}} + \beta \tag{6}$$

Furthermore, the fusion of convolutional layers and BN layers is not limited to RepVGG. At present, most target detection models use a combination of convolutional layer + BN layer + activation function, which has been performed from YOLOv2 to YOLOv7, so the fusion of the convolutional layer and BN layer can also be implemented in YOLOv7 [27].

Step 2: convolution branch fusion. As shown in Figure 6, the process of convolution branch fusion is given. The 1×1 convolution branch can be considered as filling the 1×1 convolution kernel into the 3×3 convolution kernel. The 3×3 convolution branch will fill the input to keep the size of the output feature map unchanged. The reason why the convolution kernel itself does not need to be changed is that the 3×3 convolution is the final fusion target. As shown in Figure 6, the identity branch that is first converted to a 1×1 convolution is a direct mapping from the output to the input. The branch identity is filled in and is turned into a 3×3 convolution. Finally, three 3×3 convolution kernels are added together to transform the multi-convolution branch structure into a single 3×3 convolution structure.

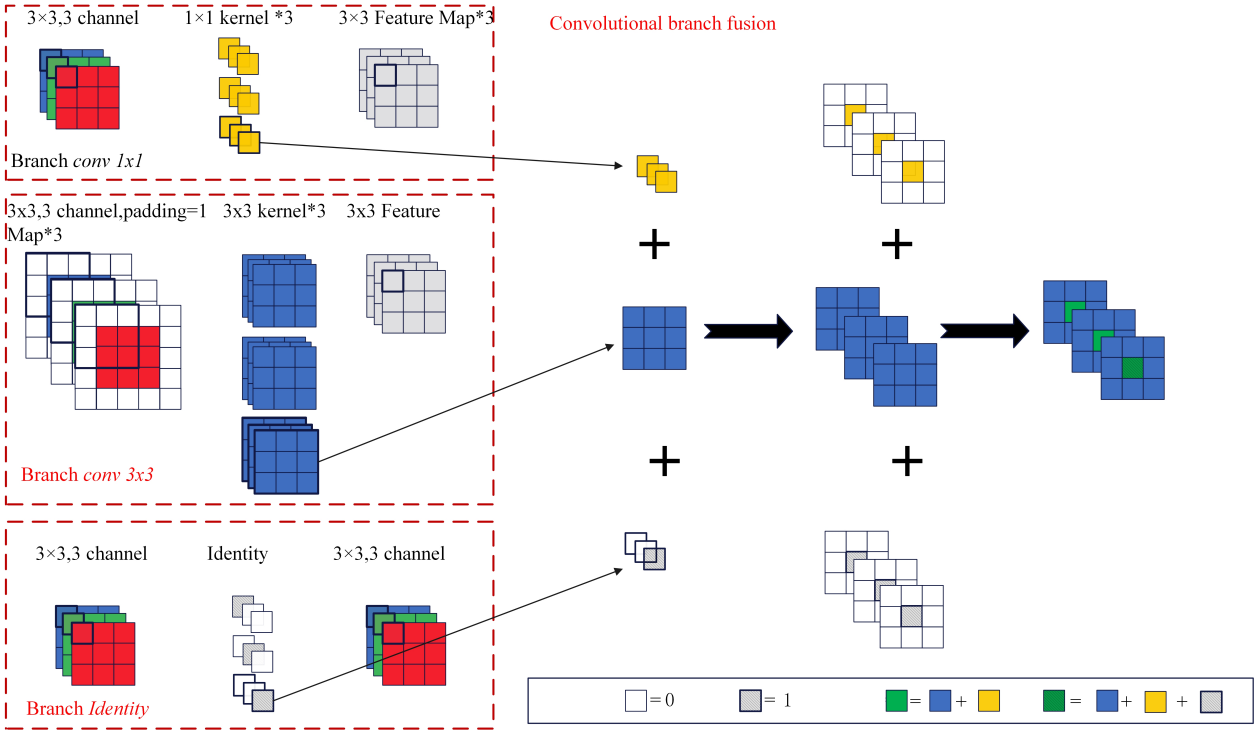


Figure 6. Convolution branch fusion flowchart.

In this study, the default stride is 1, so both the input and output feature maps have the same size. The reason why the stride must be 1 in the identity branch is that the output is a direct mapping of the input in order to keep the input and output feature mappings the same size.

2.3.2. ECA Attention

Fire smoke is not clearly distinguishable from complex backgrounds in low pixel or thin smoke conditions, so this part of the feature information is lost when the original YOLOv7 network performs feature extraction. In addition, fire smoke occurs in a wide range of scenes and in more complex environments, which increases the interference of backgrounds information to the algorithm during detection. In order to solve the above problems, this study introduces the ECA attention module to the backbone network of YOLOv7, which allows the network to focus more acutely on the fire smoke itself, enhances the focus on the core features of the fire smoke, suppresses the interference features and effectively reduces the interference of background information to deal with the problem of smoke detection in complex scenes and low pixels.

ECA module is improved from the SE module, and they are all channel attention mechanisms. As shown in Figure 7, the SE module's method of channel compression of the input to obtain the feature map is detrimental to the dependencies between learning channels, so it is important to avoid the effect of dimensionality reduction on learning channels. The ECA attention mechanism avoids this process by efficiently implementing local cross-channel interactions and extracting dependencies between channels using one-dimensional convolution, which can significantly reduce model complexity and maintain model performance [28].

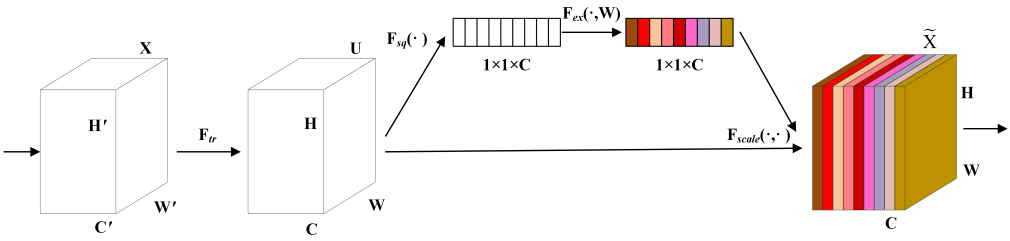


Figure 7. SE attention structure diagram.

The ECA attention mechanism works as shown in Figure 8, where the input feature map is first pooled on average, performed by a one-dimensional convolution operation, in which the weights of each feature value are generated through a Sigmoid activation function, and finally combines with the original feature layer. Among them, the ECA module generates channel weights by performing a one-dimensional convolution with a kernel size k, where k is adaptively determined by the mapping of the channel dimension C. The formula for adaptively determining the size of the convolution kernel is shown in Equation (7):

$$k = \left| \frac{\log_2 C + b}{\gamma} \right|_{odd} \tag{7}$$

where k represents the size of the convolution kernel; C represents the number of channels; γ means that k only takes an odd number; and the symbols γ and b are used to change the ratio of the convolution kernel size to the number of channels C.

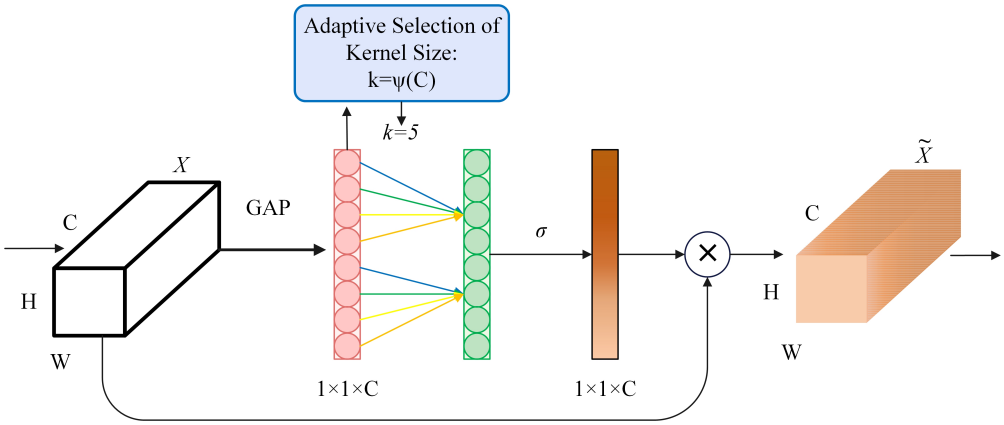


Figure 8. ECA attention structure diagram.

2.4. Loss Function

The YOLOv7 model consists of three components, namely coordinate loss (L_{box}), target confidence loss (L_{obj}) and classification loss (L_{cls}), whose total loss of the model is the weighted sum of the three losses. Among them, the confidence loss function and the classification loss function adopt the binary cross entropy loss function, while the coordinate loss adopts the CIoU loss function [29].

$$LOSS = W_1 \times L_{box} + W_2 \times L_{cls} + W_3 \times L_{obj} \tag{8}$$

In Equation (8), W_1, W_2 and W_3 are weight values of the three loss functions, respectively. The optimized loss function in this study is the coordinate loss function, and the traditional CIoU loss function is replaced by the SIoU regression loss function.

SIoU Loss

Traditional regression losses such as GIoU [30], DIoU [31] and CIoU [32] only consider the distance between the predicted frame and the real frame, the overlapping area and the aspect ratio. The reason why the traditional regression loss functions converge slowly is that the angle between the true frame and the predicted frame is not considered. Therefore, Gevorgyan proposed the SIoU loss function that re-describes the distance through the angle cost. This method is more effective at speeding up the training convergence process and works by causing the prediction frame to first move to the nearest axis (x-axis or y-axis) and then regressing along that axis to achieve global convergence. The SIoU regression loss function consists of four parts: angle loss, distance loss, shape loss and IoU loss [33]. The parameters used in the SIoU loss function are shown in Figure 9.

Angular loss: the decision to use minimization β or α is made by determining whether the angle is greater than 45°, which can be defined by the following formula:

$$\wedge = 1 - 2 * \sin^2(\arcsin(\frac{c_h}{\sigma}) - \frac{\pi}{4}) \tag{9}$$

in:

$$\frac{C_h}{\sigma} = \sin(\alpha). \tag{10}$$

$$C_h = \max(b_{cy}^{gt}, b_{cy}) - \min(b_{cy}^{gt}, b_{cy}) \tag{11}$$

$$\sigma = \sqrt{(b_{cx}^{gt} - b_{cx})^2 + (b_{cy}^{gt} - b_{cy})} \tag{12}$$

In the Equations (9)–(12), C_h is the height difference between the center point of the real frame and the predicted frame, while σ is the distance between the center point of the real frame and the predicted frame. The symbols b_{cx}^{gt} and b_{cy}^{gt} are denoted as the center coordinates of the real frame, while b_{cx} and b_{cy} represent the center coordinates of the predicted frame.

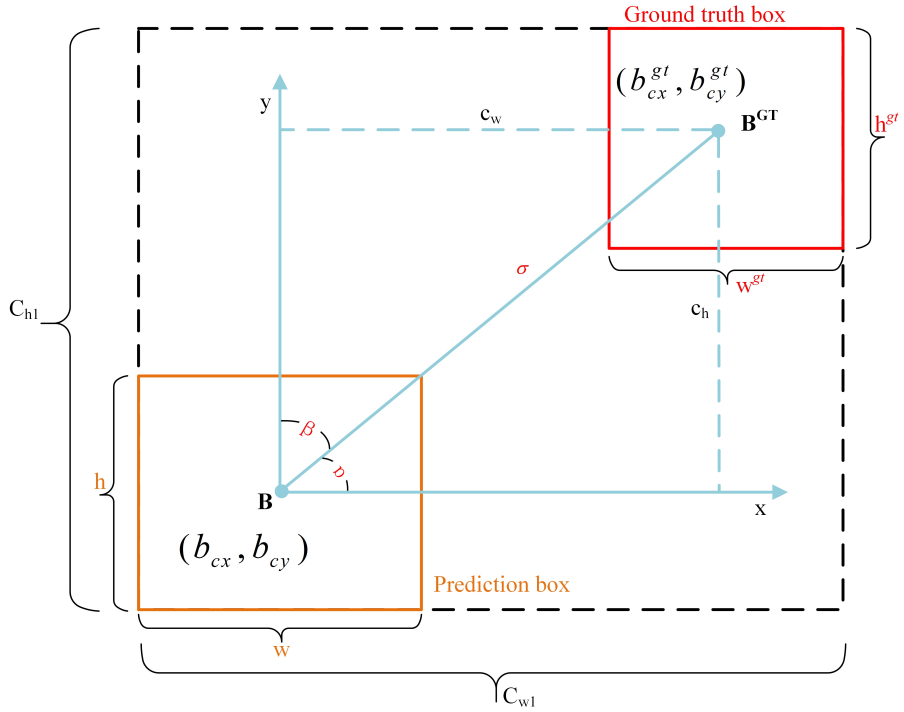


Figure 9. Schematic diagram of calculation parameters.

Distance loss: the distance loss represents the distance between the predicted frame and the center point of the real frame. By combining the angle loss, the distance loss is defined as follows:

$$\Delta = 2 - e^{-\gamma(\frac{b_{cx}^{gt} - b_{cx}}{C_w})^2} - e^{-\gamma(\frac{b_{cy}^{gt} - b_{cy}}{C_w})^2} \tag{13}$$

in:

$$\gamma = 2 - \wedge \tag{14}$$

In the Equations (13) and (14), C_w is the width of the minimum bounding rectangle of the real frame and the predicted frame, and C_h is the height of the minimum bounding rectangle of the real frame and the predicted frame.

Shape loss: the definition of shape loss is given in Equation (15):

$$\Omega = [1 - e^{-\frac{|w - w^{gt}|}{\max(w, w^{gt})}}]^{\theta} + [1 - e^{-\frac{|h - h^{gt}|}{\max(h, h^{gt})}}]^{\theta} \tag{15}$$

This equation shows that (w, h) represents the width and height of the predicted frame; the symbols w^{gt} and h^{gt} represent the width and height of the predicted frame; and θ controls the degree of attention to shape loss.

IoU loss: the following equation shows the definition of IoU loss, which is calculated by the intersection ratio of the true frame to the predicted frame shown in Figure 10.

$$IoU = \frac{A \cap B}{A \cup B} \tag{16}$$

In summary, the SIoU loss function is defined as follows:

$$L_{box} = 1 - IoU + \frac{\Delta + \Omega}{2} \tag{17}$$

As the angular cost increases, the loss function would be more fully expressed, while it reduces the probability of the penalty term being zero, which allows the loss function to

converge more smoothly and improves the regression accuracy so that prediction errors can be reduced.

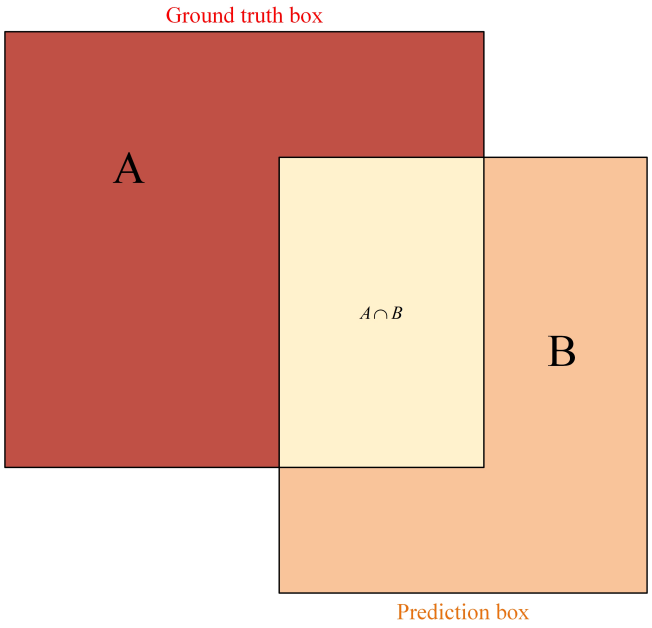


Figure 10. Schematic diagram of IoU calculation.

2.5. Soft-NMS

The non-maximum suppression algorithm (NMS) is the standard configuration of the anchor series target detection algorithms. Nowadays, most One-Stage and Two-Stage algorithms use NMS as the last layer of the network in the inference stage, such as YOLOv5, SSD and Faster-RCNN [34].

The essence of NMS is to search for local maximum values and suppress non-maximum value elements, which are often used to eliminate redundant detection frames in target detection (eliminating repeated detection frames from left to right and only retaining the current maximum confidence detection frame). This method may increase the difficulty so that when the intersection ratio of the detection frame to the maximum confidence frame is greater than a threshold setting, the detection frame will be deleted directly. The reason why using the NMS algorithm reduces the detection accuracy of the occluded object is that the occluded detection target occupies fewer pixels in the image and is more closed to the occluded frame. The YOLOv7 network used in this study also uses NMS. Therefore, when the network detects dense or obscured fire smoke, the non-maximum suppression algorithm directly eliminates predicted frames of the intersection ratio that has the highest confidence score which exceeds a certain threshold of the IoU compared with other predicted frames. This can lead to a consequence where the prediction frames for obscured smoke suppress all the prediction frames for obscured smoke, and then a missed detection occurs.

In order to address the above problems, this study uses the soft non-maximum suppression algorithm (Soft-NMS) algorithm instead of the NMS algorithm. The Soft-NMS algorithm is different from the NMS algorithm, in that it does not directly remove detection frames with low confidence, but selects a prediction frame with the highest confidence as a benchmark, and later re-scores it based on the score recurrence results. When the detected smoke overlaps, the prediction frame will not be deleted so as to improve the model's ability to detect occluded smoke and overlapping smoke. The formula definition of Soft-NMS is calculated as follows:

$$S_i = \begin{cases} S_i & , IoU(M, b_i) < N_t \\ S_i e^{-\frac{IoU(M,b_i)^2}{\sigma}} & , IoU(M, b_i) \geq N_t \end{cases} \quad (18)$$

In the Equation (18), S_i represents the score of the i-th detection frame; M represents the candidate frame with the highest score; b_i represents the score of the i-th detection frame; N_i is the set threshold; and IoU is the intersection ratio of b_i and M.

3. Result and Discussion

3.1. Experimental Environment

The experiments in this paper are conducted on Ubuntu 18.04 with an Intel(R) Core(TM) i7-12700K 3.61 GHz CPU and an NVIDIA RTX A4000 graphics card with 16 GB of video memory. The network framework is built by using Pytorch 1.7.1 with CUDA version 11.0, and the version of Python language environment is 3.8.

3.2. Model Evaluation Indicators

In order to objectively and comprehensively evaluate the performance of the proposed model in this paper, precision, average precision, recall, F1-score and frames per second (FPS) are used as the indicators to measure the prediction results. P, AP, R and the calculated parameters in F1 are all derived from the confusion matrix, as shown in Table 2.

Table 2. Confusion matrix.

Reference \ Prediction	Positive	Negative
Positive	Ture Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	Ture Negative (FP)

When the IoU between the detected frame and the marked frame is greater than the threshold, the model detects it as a positive sample; otherwise, it is considered as a negative sample detected by the model. Based on this, the sample results of the object detection model can be divided into true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). The expressions of P, AP, R and F_1 are derived as follows:

$$P = TP/(TP + FP) \tag{19}$$

$$R = TP/(TP + FN) \tag{20}$$

$$AP = \int_0^1 PdR \tag{21}$$

$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{22}$$

In the above formulas, P is the number of correctly predicted positive samples in the verification data set divided by the number of positive samples predicted by the model. R refers to the number of correctly predicted positive samples in the verification data set divided by the number of actual positive samples. mAP is the average of multiple classes. The target detection method in this study is mainly aimed at the single-class target detection of fire smoke. On this basis, both mAP and AP represent the same meaning. This paper only mentions AP, which is one of the important indicators to measure the performance of object detection. The value of AP represents the integral of the P-R curve. The larger the value of AP, the better the detection effect of the algorithm and the higher the recognition accuracy would be. The score of F_1 represents the average of the P-R curve precision value, which is the harmonic mean of P and R.

Frames Per Second (FPS) is an important indicator to measure the detection speed, whose formula is defined as follows:

$$FPS = \frac{1}{t} \tag{23}$$

In the Equation (23), t represents the time required to process each frame of image.

In addition, this study also uses the number of model parameters (Params) and FLOPs, which measure the model performance. The sum of the parameters of each layer of the neural network structure reflects the size of the model. FLOPs refers to the number of floating-point operations, which reflects the amount of calculation in the model and can be used to measure the complexity of the model.

3.3. Model Training

In the experiments in this chapter, the number of model training is 100; the batch size of model training is set to 16; and the input size is set to 640 × 640. Regularization is performed each time through the BN layer to update the weights of the model. The momentum factor is set to 0.917, and the decay of the weights is set to 0.0005. The initial vector was set to 0.01, and the enhancement factors for hue (H), saturation (S) and brightness (V) were 0.015, 0.7 and 0.4, respectively. During the training process, the Tensorboard visualization tool is used to record data and observe various types of losses, and to save the model weights for each Epoch.

3.4. Method and Effect Analysis

In order to verify the effectiveness of the proposed model in this paper, numerical experiments are conducted. The model parameters, FLOPs of YOLOv7 and RepVGG-YOLOv7 are all calculated. The effects of the loss function, attention mechanism and non-maximum suppression algorithm selected in this study are used for a comprehensive comparison. The benchmark model for each experiment is YOLOv7. The reason why this experiment mainly uses *P*, *R*, *AP* and FPS to evaluate the validity of the model is that this study focuses on improving the accuracy of smoke detection and the speed of detection.

The Params and FLOPs of YOLOv7 and RepVGG-YOLOv7 are calculated and the results are shown in Table 3. Based on the data shown in Table 3, it can be seen that the Params and FLOPs of RepVGG-YOLOv7 are smaller than those of YOLOv 7. On the one hand, the reason why the Params and FLOPs of RepVGG-YOLOv7 are slightly lower is that the convolutional and BN layers of RepVGG-YOLOv7 are fused during the inference process. On the other hand, the reason why the Params and FLOPs of RepVGG-YOLOv7 are reduced is that the reparameterization of the RepVGGBlock and the convolutional branches are fused.

Table 3. Comparison of Params and FLOPs for training and inference.

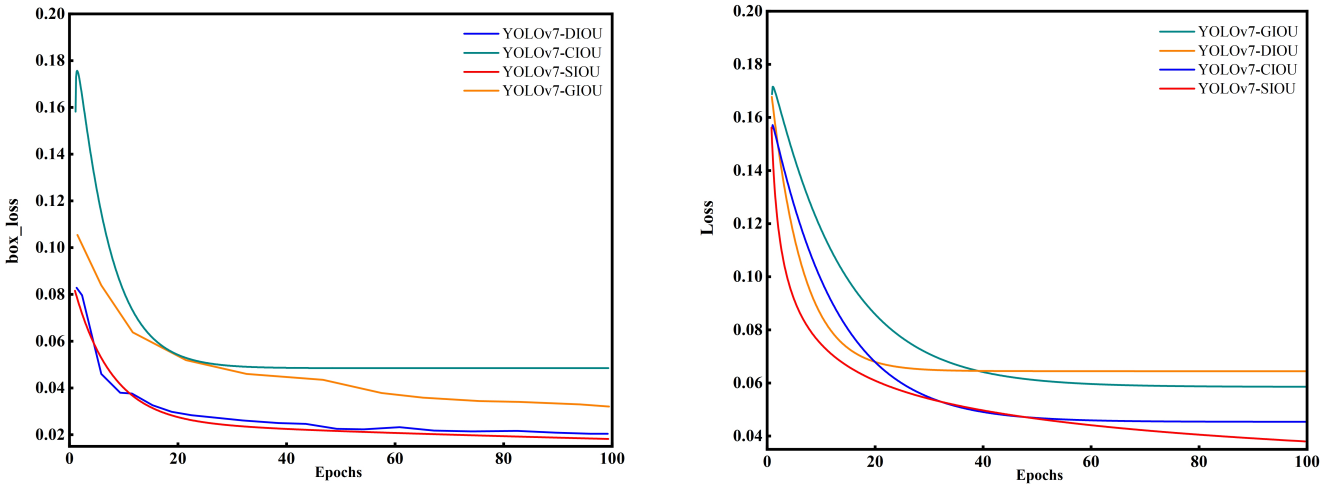
	Model	Col Params (M)	FLOPs (G)
Training	YOLOv7	36.90	103.5
	RepVGG-YOLOv7	35.83	102.7
Inference	YOLOv7	36.71	102.9
	RepVGG-YOLOv7	35.31	102.1

In order to verify the effectiveness of the loss function selection and to compare different loss functions, the original YOLOv7 model uses the CIoU loss function. Therefore, this study was compared with the GIoU, DIoU, CIoU and SIoU performance, and the obtained results are shown in Table 4.

Table 4. Performance of different loss functions.

Loss Function	P	R	AP
GIoU	0.902	0.884	0.901
DioU	0.911	0.891	0.910
CIoU(YOLOv7)	0.934	0.911	0.912
SIoU(Our)	0.939	0.917	0.921

As shown in Table 4, the loss function selected in this study has a certain improvement in P, R and AP compared with other loss functions, with an improvement of 0.05 in P, 0.06 in R and 0.09 in AP, compared with the CIoU loss function used in YOLOv7. Experimental results show that the SIoU loss function selected in this paper has good performance in smoke detection. The training results of different loss functions are shown in Figure 11. After more than 40 iterations, the training curves of different loss functions tend to be stable, and experiments prove that the SIoU loss function selected in this study has a better decay rate and convergence ability.



(**a**) box_loss curve (**b**) Loss curve

Figure 11. Training curves of different loss functions.

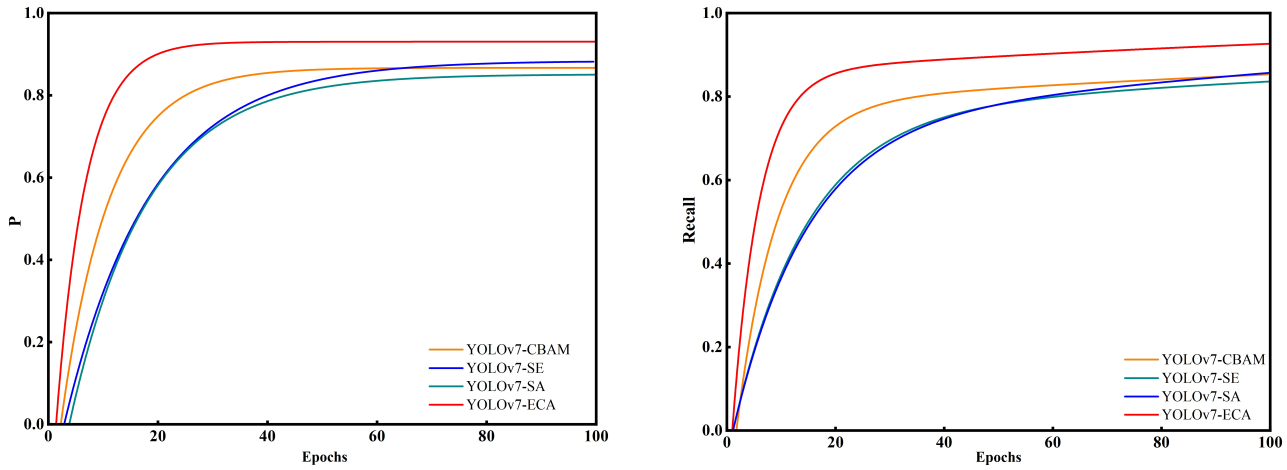
In order to incorporate the attention mechanism with the best boosting effect in the network of this study, SA spatial attention, SE channel attention block, CBAM attention and ECA channel attention are added to the backbone of the model for training, as well as comparison, in this paper. The specific results are shown in Table 5.

Table 5. Performance of different attention mechanisms.

Attention	P	R	FPS
Backbone	0.934	0.922	119.7
+SA	0.914	0.919	114.0
+SE	0.932	0.921	108.3
+CBAM	0.936	0.927	112.5
+ECA	0.938	0.931	110.4

As shown in Table 5, different attention mechanisms have different effects on the detection performance of the model. The addition of some attention mechanisms will improve the performance of the model, and some will reduce the detection ability of the model. The introduction of SA reduces the accuracy of the model by 2%. Although SE attention takes into account the attention to the channel, the experimental results show that the performance of the model is reduced by the inclusion of the SE attention mechanism. Compared with the original network, P is reduced by 0.2% and FPS is reduced by 1.4. CBAM attention uses spatial information by reducing the channel dimension of the input tensor and then calculating spatial attention through convolution. However, convolution can only capture local relationships. Therefore, the reason why the model performance is improved is that the CBAM attention is added to the network. The ECA attention mechanism avoids the impact of dimensionality reduction on the learning channels and uses one-dimensional convolution to efficiently implement local cross-channel interactions

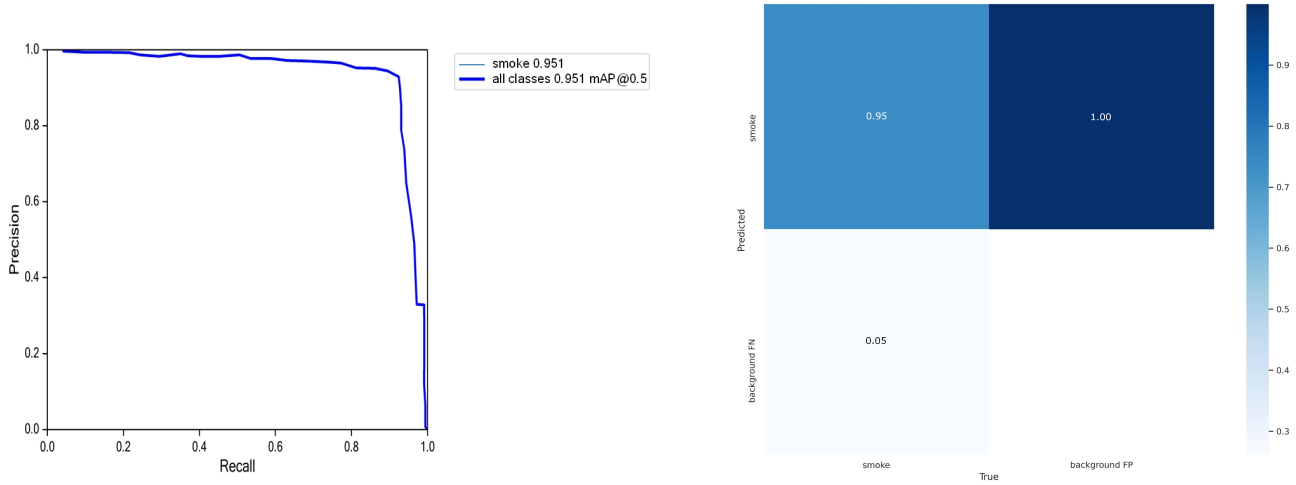
and to extract dependencies between channels, which significantly reduces model complexity while maintaining model performance. The results show that the P increased by 0.4% and the FPS decreased by 9.3, compared with the original network after adding the ECA attention mechanism to the backbone network. This proves that the ECA attention mechanism can effectively improve the network detection ability in smoke detection. The training curves of different attention mechanisms are shown in Figure 12. It can be seen that the training results of the attention mechanism selected in this paper are better than other attention mechanisms.



(**a**) Accuracy curve (**b**) Recall curve

Figure 12. Training curves of different attention mechanisms.

The *P-R* curve of the model is shown in Figure 13a. The horizontal axis of the P-R curve is the recall rate and the vertical axis is the accuracy rate, where the rate of increase in recall rate and change in accuracy rate can be visualized. If the curve in the graph is close to the upper right-hand corner, it shows that the accuracy decreases slowly as the recall rate increases and indicates better overall performance of the model. Figure 13b is the confusion matrix, whose row directions and column directions represent the real labels and predicted categories, respectively. The values in each row show that the accuracy of smoke detection is 95.1%. The confusion matrix is a summary of the prediction results for the classification problem, and it is evident that the smoke detection results are accurate.



(**a**) P-R curve (**b**) Confusion matrix

Figure 13. The model training results of this paper.

3.5. Comparison of Different Models

In order to verify the detection performance of the improved model in this paper, the proposed model is compared with current advanced target detection networks on the same experimental environment as well as the same smoke dataset, which mainly tests the F_1 score, P, R, AP and FPS of the model, and the experimental results are shown in Table 6.

Table 6. Test results of different detection networks.

Method	Backbone	F_1 Score	P	R	AP	FPS
One-stage detector						
SSD	VGG19	0.833	0.835	0.831	0.841	31.2
RetinaNet	ResNeXt101	0.717	0.734	0.701	0.753	-
M2Det	VGG16	0.758	0.774	0.743	0.751	-
YOLOv4	CSPDarkNet53	0.904	0.907	0.901	0.897	12.1
YOLOv4-tiny	CSPDarkNet53_tiny	0.887	0.899	0.877	0.891	23.5
YOLOv5s	CSPDarkNet53	0.917	0.926	0.910	0.915	67.5
YOLOv7	E-ELAN	0.922	0.934	0.911	0.922	119.7
Two-stage detectors						
Faster R-CNN	VGG16	0.847	0.854	0.841	0.861	-
R-FCN	ResNet101	0.699	0.712	0.688	0.692	-
FPN	ResNet101	0.732	0.745	0.721	0.701	-
Our						
RepVGG-YOLOv7	E-ELAN	0.937	0.951	0.924	0.937	104.7

Compared with the two-stage target detection networks, the single-stage target detection networks offer better real-time performance, and are suitable for fire and smoke detection. The current advanced two-stage target detection has higher accuracy, but its detection speed is comparatively lower. From the experimental results in Table 6, it can be seen that the single-stage target detectors SSD and RetinaNet have faster detection speed, but their accuracy is low. The detection accuracy of M2Det is slightly higher than SSD and RetinaNet, but its detection speed is low and the real-time performance is poor. The YOLOv5s in the Table 6 shows some improvement in accuracy and detection speed over YOLOv4 and YOLOv4-tiny, with an increase in AP, and FPS has increased by 55.4 FPS and 44 FPS, respectively. Compared to several other networks, YOLOv7 has outstanding performance in both detection accuracy and detection speed; especially compared with YOLOv5s, YOLOv7 has some improvement in detection accuracy and detection speed. The RepVGG-YOLOv7 proposed in this study is improved from YOLOv7. The results show some improvement in detection accuracy but a slight reduction in its detection speed, which is reduced by 15 FPS, but it still meets the real-time requirements. RepVGG-YOLOv7, which outperforms the other models, in Table 6, for fire smoke detection, achieves a good balance between detection speed and detection accuracy.

3.6. Ablation Experiment

In order to further explore the impact of different improved parts of the algorithm on the performance of the detection algorithm in this paper, experimental validation of each improvement of the YOLOv7 network structure is carried out using YOLOv7 as the basic algorithm. The ablation experiments, for which experimental results are shown in Table 7, are carried out on the self-built dataset from this paper.

It can be seen from the above table that the addition of RepVGGBlock to the original YOLOv7 network improves P by 1% relative to the original YOLOv7 network and reduces the number of model parameters, but its impact on the detection speed is not significant. With the addition of the ECA attention mechanism, detection accuracy is improved by 0.2% compared to the original network, and detection speed is kept in line with the original network. The reason why the model complexity is significantly reduced and the model performance is maintained is that the ECA attention mechanism extracts the dependencies between channels. After the loss function is improved, the SIoU loss function is used. Due

to the increased angular cost, the loss function is more fully expressed while reducing the probability of the penalty term being zero, which allows the loss function to converge more smoothly and improves the regression accuracy to reduce the prediction error. In this way, the prediction error is reduced. It can be seen from the experiment that the accuracy of model detection is increased by 0.5%, but the detection speed of the model will not be reduced. The introduction of Soft-NMS will improve the model's ability to detect occluded smoke and overlapping smoke. Therefore, the experimental results show that the accuracy of the model has increased by 0.3%. At the same time, the introduction of RepVGGBlock, ECA attention, the SIoU loss function and Soft-NMS significantly improves the detection accuracy of the detection algorithm. In summary, the introduction of the modules does not have a major impact on the detection speed of the algorithm, which still achieves 104.7 fps in this paper.

Table 7. Results of ablation experiments.

Model	P	Δ	FPS	Δ	Params (M)	Δ
YOLOv7	0.934	0	119.7	0	36.71	0
+RepMGGBlock	0.935	+0.001	110.6	−9.1	35.31	−1.4
+ECA	0.938	+0.004	114.4	−5.3	36.72	+0.01
+SIoU	0.939	+0.005	119.2	−0.5	36.71	0
+Soft-NMS	0.937	+0.003	106.9	−12.8	36.71	0
+RepMGGBlock+ECA+Soft-NMS	0.946	+0.012	104.9	−14.8	35.41	−1.3
+RepVGGBlock+ECA+SIoU+Soft-NMS	0.951	+0.017	104.7	−15.0	35.41	−1.3

3.7. Comparison of Visualization Results

In order to analyze the validity of the proposed model in this paper more intuitively, the detection results of the proposed algorithm in this paper are compared with the SSD, YOLOv5s and YOLOv7 algorithms. To fully demonstrate the detection capabilities of different algorithms in different scenarios and different targets, the visual detection results mainly include small target smoke detection results and smoke detection results in complex backgrounds.

3.7.1. Small Target Smoke Detection Results

The detection results are shown in Table 8. The small target images selected in this paper include indoor and outdoor scenes. Pictures contain multiple smoke targets and outdoor scenes.

In Table 8a, the SSD algorithm failed to detect all smoke targets and misidentified the reflection in the water as the actual smoke area. The YOLOv5s and YOLOv7 algorithms detect all smoke targets, but their detection areas are larger than the actual area, which contain more background information. However, RepVGG-YOLOv7 enhances the extraction of smoke features, detection of small target smoke and it has a high degree of confidence. In Table 8b, the reason why the SSD algorithm fails to detect the smoke is that the room is well-lit and the smoke is white. YOLOv5s and YOLOv7 detected only part of the smoke area, while RepVGG-YOLOv7 detected most of the smoke area. In Table 8c, all four algorithms can detect the smoke target, but RepVGG-YOLOv7 detects the most accurate area of smoke and its confidence level is the highest.

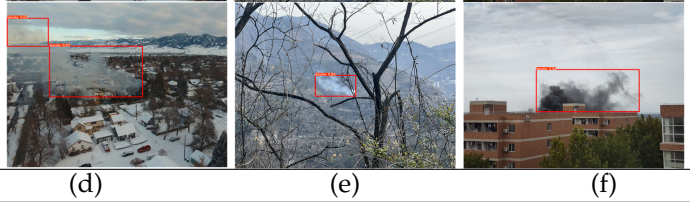
Table 8. Visualization results of small target smoke detection.

Method	Detection Result		
SSD			
YOLOv5s			
YOLOv7			
RepVGG-YOLOv7			
	(a)	(b)	(c)

3.7.2. Smoke Detection Results in Complex Backgrounds

The detection results are shown in Table 9. The complex background smoke selected in this paper includes mainly obscured smoke images, as well as images containing cloud, fog, snow and other types of smoke.

Table 9. Visualization results of smoke detection under complex backgrounds.

Method	Detection Result		
SSD			
YOLOv5s			
YOLOv7			
RepVGG-YOLOv7			
	(d)	(e)	(f)

In Table 9d, there are distractions such as white clouds, snow and black mountains, as well as thin smoke edges and some similarity between the smoke background image and the smoke, so the SSD algorithm has a false detection situation where it misidentifies the black mountains in the figure as the smoke. The YOLOv5s and YOLOv7 algorithms can accurately detect areas of smoke. The reason why the YOLOv5s and YOLOv7 algorithms need to be improved in the detection of smoke edges is that they ignore the thin smoke area in the upper left corner of the figure. The RepVGG-YOLOv7 algorithm uses two detection frames to detect all of the smoke. In Table 9e, there are black trees blocking the smoke target. Although the SSD algorithm can detect the smoke, the detection area includes the black trees in the foreground. The YOLOv5s algorithm has a higher detection accuracy than the SSD detection algorithm, but it still contains more non-smoke areas. RepVGG-YOLOv7 significantly improves the effect of detection, whose effect is more accurate to detect smoke areas. Table 9f mainly shows the black smoke target, which contains several black smoke areas, mainly windows, and branches. Although the SSD algorithm detects the smoke area, it also mistakenly detects the black window area in the lower right corner of the figure as smoke. The YOLOv5s algorithm also mistakenly detects the tree branch to the left of the smoke area as smoke. Although both the YOLOv7 and RepVGG-YOLOv7 algorithms accurately detect the smoke, RepVGG-YOLOv7 has a better confidence level.

4. Conclusions

In this paper, we propose a novel detector named RepVGG-YOLOv7 for fire smoke detection, which focuses on the improvement of the YOLOv7 backbone network. Firstly, the introduction of RepVGG makes it possible for the network to achieve simple inference with complex training, and the application of ECA attention as well as the SIoU loss function makes it possible for the model to converge faster when training and focus more acutely on the target to be detected. The experiments prove that the proposed model in this paper performs best compared with other state-of-the-art networks. It demonstrates the detection effectiveness of the model without increasing its complexity. However, there are still some deficiencies in this work. On the one hand, although the fire smoke dataset in this paper is self-built and expanded, there are still some scenarios that are not included in the dataset, such as fire smoke at night, smoke from special occasions, etc. In the future, the dataset will continue to be improved and expanded, and will also be constructed into a dataset recognized by a wide range of scholars. On the other hand, the proposed fire smoke detection algorithm requires intensive computations and makes it difficult to reduce parameters. Thus, it is hoped to continue to reduce the number of parameters in the smoke detection model to make it lighter so that it can be deployed in practice under realistic conditions and support smoke detection services.

Author Contributions: Formal analysis, X.C., Y.X. and Y.F.; Investigation, X.C. and Y.X.; Methodology, X.C. and Y.X.; Project administration, X.C. and Y.X.; Resources, X.C., Y.X., Y.F. and Y.Z.; Writing—original draft, X.C. and Y.X.; Writing—review and editing, X.C. and Y.X.; Conceptualization, Y.X.; Software, Y.X.; Visualization, Y.X.; Data curation, Y.X.; Funding acquisition, Q.H.; Supervision, Q.H., Y.F. and Y.Z.; Validation, Q.H. and Y.F. All authors have read and agreed to the published version of the manuscript.

Funding: The work is supported by the National Natural Science Foundation of China under Grant No. 61673253, the Chinese Postdoctoral Science Foundation under Grant No. 2020M683562, the Specialized Research Fund for Xi'an University Talent Service Enterprise Project under Grant No. 23GXFW0027, the Natural Science Foundation of Shaanxi Province of China under Grant No. 2022JM-331, and the Key Research and Development Program of Shaanxi Provincial Science and Technology under Grant No. 2023-YBGY-142.

Institutional Review Board Statement: Written informed consent has been obtained from the patient(s) to publish this paper.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Hu, Y.; Zhan, J.; Zhou, G.; Chen, A.; Cai, W.; Guo, K.; Hu, Y.; Li, L. Fast forest fire smoke detection using MVMNet. *Knowl.-Based Syst.* **2022**, *241*, 108219. [CrossRef]
2. Mozaffari, M.; Li, Y.; Ko, Y. Real-time detection and forecast of flashovers by the visual room fire features using deep convolutional neural networks. *J. Build. Eng.* **2023**, *64*, 105674. [CrossRef]
3. Khan, F.; Xu, Z.; Sun, J.; Khan, F.M.; Ahmed, A.; Zhao, Y. Recent advances in sensors for fire detection. *Sensors* **2022**, *22*, 3310. [CrossRef] [PubMed]
4. Jiao, Z.; Zhang, Y.; Xin, J.; Mu, L.; Yi, Y.; Liu, H.; Liu, D. A deep learning based forest fire detection approach using UAV and YOLOv3. In Proceedings of the 2019 1st International Conference on Industrial Artificial Intelligence, Shenyang, China, 23–27 July 2019; pp. 1–5.
5. Huo, Y.; Zhang, Q.; Jia, Y.; Liu, D.; Guan, J.; Lin, G.; Zhang, Y. A deep separable convolutional neural network for multiscale image-based smoke detection. *Fire Technol.* **2022**, *241*, 1–24. [CrossRef]
6. Xue, X.; Feiniu, Y.; Lin, Z. From tradition to depth: Visual smoke recognition, detection and segmentation. *J. Image Graph.* **2019**, 1627–1647.
7. Zhou, B.L.; Song, Y.L.; Yu, M.H. Fire smoke detection algorithm based on image disposal. *Fire Sci. Technol.* **2016**, *35*, 390–393.
8. Gomez-Rodriguez, F.; Arrue, B.C.; Ollero, A. Smoke monitoring and measurement using image processing: Application to forest fires. *Autom. Target Recognit. XIII* **2003**, *5094*, 404–411.
9. Filonenko, A.; Hernández, D.; Jo, K. Real-time smoke detection for surveillance. In Proceedings of the 2015 IEEE 13th International Conference on Industrial Informatics, London, UK, 22–24 July 2015; pp. 568–571.
10. Liu, Z.; Yang, Y.; Ji, X. Flame detection algorithm based on a saliency detection technique and the uniform local binary pattern in the YCbCr color space. *Signal Image Video Process.* **2016**, *10*, 277–284. [CrossRef]
11. Frizzi, S.; Kaabi, R.; Bouchouicha, M.; Ginoux, J.; Moreau, E.; Fnaiech, F. Convolutional neural network for video fire and smoke detection. In Proceedings of the 2016 42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; pp. 877–882.
12. LeCun, Y.; Bottou, L.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
13. Jiang, P.; Ergu, D.; Liu, F.; Cai, Y.; Ma, B. A Review of Yolo algorithm developments. *Procedia Comput. Sci.* **2022**, *199*, 1066–1073. [CrossRef]
14. Jia, D.; Zhou, J.; Zhang, C. Detection of cervical cells based on improved SSD network. *Multimed. Tools Appl.* **2022**, *81*, 13371–13387. [CrossRef]
15. Saponara, S.; Elhanashi, A.; Gagliardi, A. Real-time video fire/smoke detection based on CNN in antifire surveillance systems. *J. Real-Time Image Process.* **2021**, *18*, 889–900. [CrossRef]
16. Chen, X.; Xue, Y.; Zhu, Y.; Ma, R. A novel smoke detection algorithm based on improved mixed Gaussian and YOLOv5 for textile workshop environments. *IET Image Process.* **2023**, *17*, 1991–2004. [CrossRef]
17. Wang, Z.; Wu, L.; Li, T.; Shi, P. A smoke detection model based on improved YOLOv5. *Mathematics* **2022**, *10*, 1190. [CrossRef]
18. Al-Smadi, Y.; Alauthman, M.; Al-Qerem, A.; Aldweesh, A.; Quaddoura, R.; Aburub, F.; Mansour, K.; Alhmiedat, T. Early Wildfire Smoke Detection Using Different YOLO Models. *Machines* **2023**, *11*, 246. [CrossRef]
19. Yuan, F.; Zhang, L.; Wan, B.; Xia, X.; Shi, J. Convolutional neural networks based on multi-scale additive merging layers for visual smoke recognition. *Mach. Vis. Appl.* **2019**, *30*, 345–358. [CrossRef]
20. Zhang, Q.; Lin, G.; Zhang, Y.; Xu, G.; Wang, J. Wildland forest fire smoke detection based on faster R-CNN using synthetic smoke images. *Procedia Eng.* **2018**, *211*, 441–446. [CrossRef]
21. Gagliardi, A.; de Gioia, F.; Saponara, S. A real-time video smoke detection algorithm based on Kalman filter and CNN. *J. Real-Time Image Process.* **2021**, *241*, 1–11. [CrossRef]
22. Lin, G.; Zhang, Y.; Xu, G.; Zhang, Q. Smoke detection on video sequences using 3D convolutional neural networks. *Fire Technol.* **2019**, *55*, 1827–1847. [CrossRef]
23. Durve, M.; Orsini, S.; Tiribocchi, A.; Montessori, A.; Tucny, J.; Lauricella, M.; Camposeo, A.; Pisignano, D.; Succi, S. Benchmarking YOLOv5 and YOLOv7 models with DeepSORT for droplet tracking applications. *Eur. Phys. J. E* **2023**, *46*, 32. [CrossRef]
24. Hussain, M.; Al-Aqrabi, H.; Munawar, M.; Hill, R.; Alsboui, T. Domain feature mapping with YOLOv7 for automated edge-based pallet racking inspections. *Sensors* **2022**, *22*, 6927. [CrossRef] [PubMed]
25. Zhao, C.; Shu, X.; Yan, X.; Zuo, X.; Zhu, F. RDD-YOLO: A modified YOLO for detection of steel surface defects. *Measurement* **2023**, *214*, 112776. [CrossRef]
26. Song, K.; Zhang, Y.; Lu, B.; Chi, W.; Sun, L. UAV Forest Fire Detection based on RepVGG-YOLOv5. In Proceedings of the 2022 IEEE International Conference on Robotics and Biomimetics, Xishuangbanna, China, 5–9 December 2022; pp. 1277–1282.
27. Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; Sun, J. Repvgg: Making vgg-style convnets great again. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 13733–13742.
28. Weng, J.; Han, T.; Shi, K.; Li, G. Impact analysis of ECA policies on ship trajectories and emissions. *Mar. Pollut. Bull.* **2022**, *179*, 113687. [CrossRef]

29. Reddy, E.; Rajaram, V. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the 2022 6th International Conference on Electronics, Communication and Aerospace Technology, Coimbatore, India, 1–3 December 2022; pp. 1255–1260.
30. Reddy, E.; Rajaram, V. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 658–666.
31. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. 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.
32. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **2021**, *52*, 8574–8586. [CrossRef] [PubMed]
33. Gevorgyan, Z. SIoU loss: More powerful learning for bounding box regression. *arXiv* **2022**, arXiv:2205.12740.
34. Wang, L.; Mu, X.; Ma, C.; Zhang, J. Hausdorff iou and context maximum selection nms: Improving object detection in remote sensing images with a novel metric and postprocessing module. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [CrossRef]

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