sustainability

MDPI

Article

Fire-YOLO: A Small Target Object Detection Method for Fire Inspection

Lei Zhao [1], Luqian Zhi [1], Cai Zhao [2] and Wen Zheng [1,3,*]

[1] Institute of Public-Safety and Big Data, College of Data Science, Taiyuan University of Technology, Taiyuan 030024, China; zhaolei_tyut@foxmail.com (L.Z.); zhiluqian4398@link.tyut.edu.cn (L.Z.)
[2] Center of Information Management and Development, Taiyuan University of Technology, Taiyuan 030024, China; zhaocai@tyut.edu.cn
[3] Center for Big Data Research in Health, Changzhi Medical College, Changzhi 046000, China
* Correspondence: zhengwen@tyut.edu.cn

Abstract: For the detection of small targets, fire-like and smoke-like targets in forest fire images, as well as fire detection under different natural lights, an improved Fire-YOLO deep learning algorithm is proposed. The Fire-YOLO detection model expands the feature extraction network from three dimensions, which enhances feature propagation of fire small targets identification, improves network performance, and reduces model parameters. Furthermore, through the promotion of the feature pyramid, the top-performing prediction box is obtained. Fire-YOLO attains excellent results compared to state-of-the-art object detection networks, notably in the detection of small targets of fire and smoke. Overall, the Fire-YOLO detection model can effectively deal with the inspection of small fire targets, as well as fire-like and smoke-like objects. When the input image size is 416 × 416 resolution, the average detection time is 0.04 s per frame, which can provide real-time forest fire detection. Moreover, the algorithm proposed in this paper can also be applied to small target detection under other complicated situations.

Keywords: fire inspection; small target; Fire-YOLO; real-time detection

Citation: Zhao, L.; Zhi, L.; Zhao, C.; Zheng, W. Fire-YOLO: A Small Target Object Detection Method for Fire Inspection. *Sustainability* **2022**, *14*, 4930. https://doi.org/10.3390/su14094930

Academic Editor: Andreas Kanavos

Received: 27 March 2022
Accepted: 18 April 2022
Published: 20 April 2022

1. Introduction

Fires can seriously threaten people's lives and cause major economic losses. According to incomplete statistics, there were 1153 forest fires in China in 2020, 7 of which were major forest fires. It caused an economic loss of CNY 162.19 million and caused damage to the lives and property of the people. Therefore, fire detection is essential to protect forest resources and protect people's lives and property.

In recent years, as fire imaging detection has become a research hotspot, imaging detection has the advantages of early detection, high accuracy, flexible system installation, and effective detection of large spaces and complex building structure fires. Based on deep learning and the You Only Look Once (YOLO) algorithm, an Unmanned Aerial Vehicle (UAV) [1] is used to detect the fire situation, and more accurate results have been achieved than classical methods in the actual test. Compared with UAV, camera detection can achieve 7 × 24 h all-day detection, which can better prevent forest fires. By analyzing the image information obtained from the monitor, the fire can be found in the early stage, so as to avoid the occurrence of fire danger [2]. The classic image-based fire monitoring system monitors the entire forest through satellites, and uses the images collected by the satellites to detect fire events [3]. Based on the YOLO-V5 algorithm, the approach in Ref. [4] detects small objects in remote sensing images and uses the multi-scale anchoring mechanism of a Faster Region Convolutional Neural Network (R-CNN) to improve the detection ability of the YOLO-V5 algorithm for small objects in images. Nevertheless, the images collected by satellites are low-resolution images, and different weather conditions will also affect the

detection results, which often causes problems such as missed detection, false alarms, and delayed detection.

Many researchers have provided and improved different algorithms for fire detection. A color-based fire detection method [5] with a speed of 20 frames per second was proposed. This scheme uses the Support Vector Machine (SVM) classifier to detect fires at a small distance with good accuracy. This method does not perform well when the fire distance is far or the scale of the fire is small [6]. Another correlation method consisting of motion and color features is proposed for fire detection of surveillance video, but this combined method increases the computational cost. Summarizing the color-based methods, it can be noted that these methods are sensitive to brightness and shadows. Hence, the number of false alarms generated by these methods is high.

With the development of machine learning, deep learning techniques have been widely used in detection[7]. The authors of [2] propose an early fire detection framework using fine-tuned Convolutional Neural Networks (CNN), but the model has a high computational cost. The authors of Ref. [8] use a deep fusion CNN for smoke detection, which combines attention mechanism and feature-level and decision-level fusion modules. However, there are still some small targets missed. The proposed method [9] uses a Faster Region-Based Convolutional Neural Network (R-CNN) to detect suspicious fire regions (SRoF) and non-fire regions based on their spatial features. This can successfully improve fire detection accuracy by reducing false detections, yet the detection speed is relatively slow. A forest smoke detection algorithm based on YOLO-V3 and YOLO-V4 is proposed in [10]. Compared with YOLO-V4, the model of YOLO-V3 is smaller and easier to deploy. On this basis, we choose the YOLO-V3 model as the overall algorithm and make improvements to it. An improvement to the YOLO-V3 algorithm is proposed [11]. Hollow convolution and DenseNet are added to the network to improve the detection effect of small-scale flames in the early stage of fire. However, there are problems with inaccurate flame positioning and poor performance in the presence of shielding. The I-YOLOv3-tiny model [12] is used to improve the detection accuracy through network structure optimization, multi-scale fusion, and K-means clustering, but the detection speed needs to be improved. By increasing the resolution of the feature map [13], it reduces the error in fire detection, yet due to the increase in the amount of calculation, the corresponding processing time increases. The method [14] of combining the classification model and the target detection model for fire detection reduces the computational cost and improves the detection accuracy. Nonetheless, it is not suitable for detection scenarios with small targets in the early stage of fire. By replacing the two-step down-sampling convolutional network in the original network architecture with an image bi-segmentation and bi-linear up-sampling network [15], the features of small objects are enlarged and the detection accuracy of small target objects are improved. Although this increases the number of parameters, the computational cost is also increased. For fire detection with real-time requirements, further improvements are still needed. These problems have brought huge challenges to the detection of small targets in fire scenes.

2. Method

2.1. YOLO-V3

YOLO-V3 [16] is an object detection model evolved from YOLO [17] and YOLO-V2 [18] networks. Compared with Faster R-CNN [19], YOLO-V3 is a single-stage detection algorithm, which means that the YOLO network does not require a Regional Proposal Network (RPN), but directly detects the target in the image. This not only takes into account the detection speed and detection accuracy, but also has reduced the size of the parameters of the model. The positioning of YOLO-V1 is not accurate enough, and the recall rate is lower than that of methods based on region proposal. In this regard, YOLO-V2 adds batch normalization, author mechanism, and multi-scale training to the network, so that YOLO-V2 can adapt to the input of different scales while improving detection speed and accuracy. Although the accuracy of YOLO-V2 has been improved a lot, the accuracy still

cannot meet the requirements in the subsequent industrial applications. YOLO-V3 made some improvements on the basis of YOLO-V2, changing softmax loss of YOLO-V2 into logistic loss. Meanwhile, YOLO-V3 uses logistic regression independently for each category, replacing the feature extraction network from DarkENT-19 to DarkENT-53. YOLO-V3 is nearly as accurate as other target detection algorithms but is at least twice as fast.

The feature extraction network of YOLO-V3 is Darknet-53. Darknet-53 extracts the features of fire images input into the YOLO-V3 network through constant use of convolution, standardization, and pooling operations, and continuously carries out feature extraction on fire images by convolution. This method is widely used in various other network models, and ResNet [20] also increases the accuracy of the network by increasing the depth of the network. Although it is possible to zoom in two or three dimensions at the same time, due to the certain relationship between depth and width, complicated manual adjustment is required, that is to say, only the depth and width can be adjusted to achieve better accuracy. At present, the YOLO-V3 model has not been widely used in fire detection.

2.2. Fire-YOLO

Fire-YOLO is a one-stage detection model. The following shows the steps of Fire-YOLO for fire detection. First, the network divides the input fire image into $S \times S$ grids, and detects in each detection grid unit: whether there is flame or smoke if the center of the target to be detected falls on one of the S^2 grids, the grid is responsible for detecting the target to be detected. After that, each grid predicts 3 bounding boxes and gives the confidence scores of these bounding boxes. The definition of confidence calculation is as follows:

$$\text{Confidence} = P_\gamma \times \text{IoU}_{\text{truth}}^{\text{pred}}, P_\gamma(\text{object}) \in [0, 1]$$

When the target falls in the grid $P_\gamma = 1$, otherwise $P_\gamma = 0$. The IoU represents the coincidence between the predicted bounding box and the true bounding box. The confidence level reflects whether there are objects in the grid and the accuracy of predicting the bounding box when the objects are included. When multiple bounding boxes detect the same target at the same time, the YOLO network will use the Non-Maximum Suppression (NMS) method to select the best bounding box.

The use of convolutional neural networks to detect fires by classifying fires and smoke in videos has good accuracy [21]. The work in Ref. [22] improves the YOLO-V3 model and adds multiple feature scales with a smaller resolution, which is helpful for the identification of small flame regions. Inspired by [22–24], the Fire-YOLO model proposed in this paper is considered from the three dimensions of depth, width, and resolution, and realizes a more balanced network architecture. The steps of Fire-YOLO fire detection are shown in Figure 1.

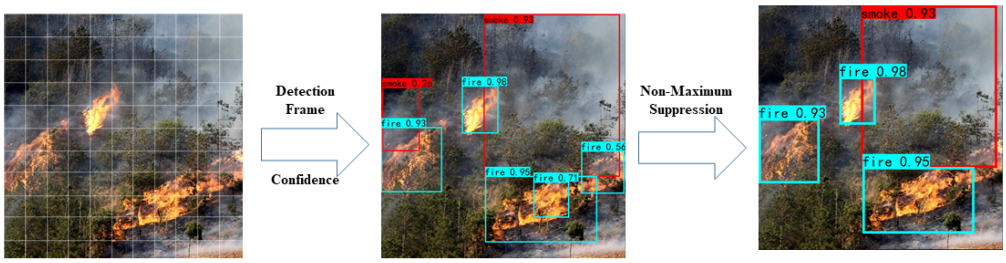

Figure 1. The input image is divided into $S \times S$ grids, and each grid predicts three bounding boxes and confidence scores. Afterward, the optimal bounding box is selected using a non-maximum suppression method.

The proposal of EfficientNet [25] offers the possibility to design a standardized convolutional neural network scaling method. By balancing the three dimensions of network depth, width, and resolution, a more balanced network architecture can be achieved in the three dimensions of depth, width, and resolution without complex manual adjustment.

Since a large number of detection objects in the fire dataset are small flames and smoke, this simple and efficient composite scaling method can further improve the accuracy of fire detection compared with other single-dimensional scaling methods, and can fully save computing resources. Ultimately, the improved EfficientNet used in this paper is faster than the convolutional neural network with the same accuracy, with fewer parameters and smaller models, which has obvious advantages.

The network structure of the proposed Fire-YOLO fire detection model is shown in Figure 2. Different from the residual block used in the feature extraction network Darknet-53 in YOLO-V3, Fire-YOLO's feature extraction network uses a mobile inverted bottleneck convolution (MBConv) [26], which consists of depthwise convolution and Squeeze-and-Excitation Networks (SENet) [27]. In the structure of the MBConv block, above all, a 1×1 convolution kernel is used to increase the dimension of the image, the next to go through depthwise convolution and SENet in turn, and at last use a 1×1 convolution kernel to reduce the dimension of the image output. After the input image is subjected to the feature extraction network and up-sampling, the corresponding feature map will be obtained, which is the vector feature matrix corresponding to the image. In order to obtain higher-level feature information of a small target, the feature map will be processed by a convolutional set, which is composed of five convolutional layers with convolution kernels of 1×1 and 3×3 alternately. Then the pooling layer is used to flatten the high-dimensional vectors into one-dimensional vectors for the activation function to process. After inputting these vectors into the activation function, the corresponding classification results are obtained, and the most likely result is selected as the feature extraction network output. Aiming at the performance defect of the Darknet-53 feature extraction network in balancing the three dimensions of depth, width, and resolution, we use the improved EfficientNet feature extraction network on the fire dataset proposed in this paper which makes up for the performance shortcomings of Darknet-53 in small target detection and improves the feature extraction ability for small target detection. For the input small target image, the learning ability of small target features is strengthened, and the performance of the small target feature extraction network is improved.

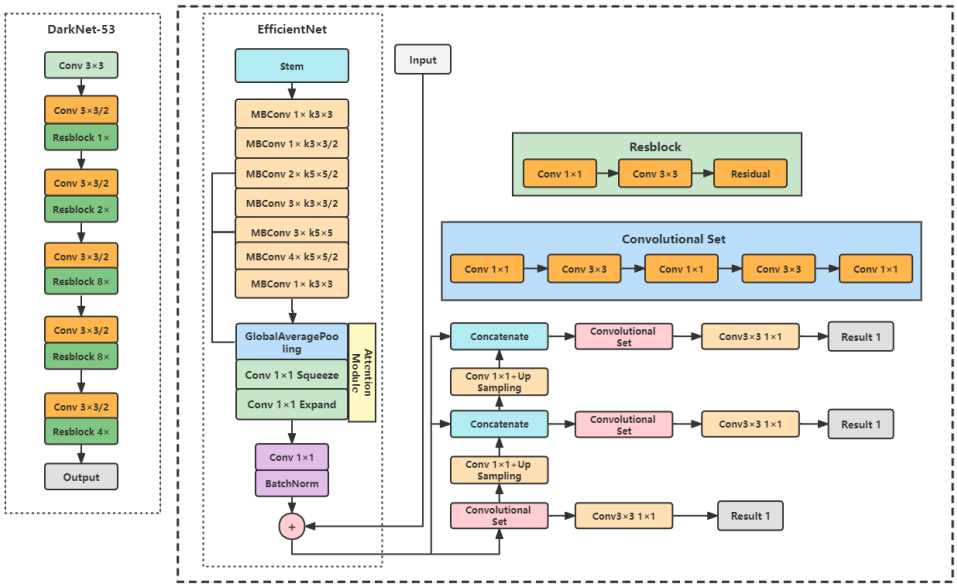

Figure 2. The picture shows the network structure of the Fire-YOLO fire detection model proposed in this paper. The Darknet-53 feature extraction framework is replaced with the EfficientNet network to extract features from the input fire pictures.

To better deal with small target images, the Fire-YOLO model initially scales the input image to 416×416 pixels and then uses EfficientNet to extract features from the image. After multiple layers of depthwise separable convolution, global average pooling, feature

compression, and feature expansion, the feature map learned by depthwise separable convolution is upsampled. Prediction boxes of different scales will be obtained by feature pyramid processing. The Fire-YOLO model proposed in this paper predicts three different scale bounding boxes, which are 13 × 13, 26 × 26, 52 × 52.

Inspired by the work in [28–31], the depthwise separable convolution used in small target detection combines two levels of channel-by-channel convolution and step-by-step point convolution to extract small target features of different granularities.The structure of depth separable convolution is shown in Figure 3.

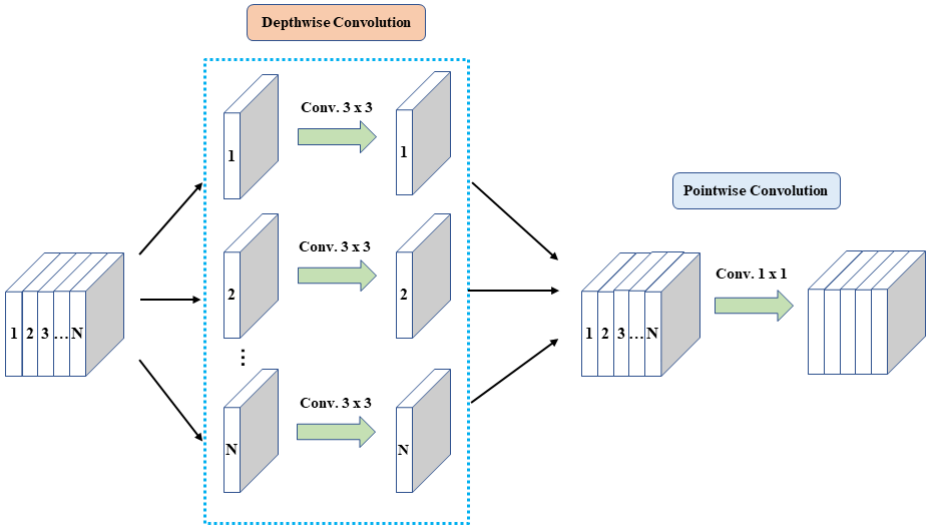

Figure 3. The figure shows the structure of deep separable convolution, which reflects the specific process of channel-by-channel convolution and step-by-step point convolution.

In classic ordinary convolution, each channel will be convolutional calculated by the convolution kernel once, while the channel-by-channel convolution is only convolved on one channel, so the number of channels of the original feature map will not be changed. For instance, the pixel size of the original feature map is $D_f \times D_f \times M$, if channel-by-channel convolution is used, the number of convolutions needs to be the same as the number of channels, which is to use M times $D_k \times D_k \times 1$ convolution kernels are used to carry out convolution calculation for a channel respectively.

Channel-by-channel convolution only allows the convolution kernel to calculate one channel alone, but in the convolutional network, there will be a lack of information transfer between channels, resulting in no exchange of information between channels. Therefore, by adding a point-wise convolution operation, using a 1×1 convolution kernel, the information between channels is further fused smoothly and tightly.

Through such a 1×1 convolution kernel, the interaction of information between information can be strengthened while the amount of calculation is reduced as much as possible, as much feature information as possible can be extracted, and the ability of the model to extract features can be enhanced. For instance, the original feature map is $D_f \times D_f \times M$. Assuming that there are a total of N 1×1 convolution kernels, then the computational cost of one convolution kernel is M, and the computational cost of processing the original feature map by one convolution kernel is $M \times D_g \times D_g$, that is we use a total of N convolution kernels. So the total cost is $N \times M \times D_g \times D_g$.

By comparing the Darknet-53 feature extraction network used in YOLO-V3, the RPN target proposal network used in Faster R-CNN, and the depthwise separable convolution used in Fire-YOLO, it is found that the depthwise separable convolution has an excellent performance in terms of computational complexity. This convolutional structure speeds up the training of the model and improves the detection accuracy of the model. In the practical application of the fire detection model, the processing speed of the network can be

accelerated, if the amount of calculation is small, the purpose of real-time image processing can be achieved, and then the real-time detection of fire danger can be realized. At the same time, the hardware requirements of the model are also reduced to facilitate deployment.

SENet mainly consists of two stages. The first is the Squeeze stage. The dimension of the original feature map is $H \times W \times C$, where H is the height, W is the width and C is the number of channels. X represents a specific block of feature maps. The original feature map is compressed to $1 \times 1 \times C$, that is, $H \times W$ is compressed into one dimension. This one-dimensional parameter covers the previous $H \times W$ global field of image, while the perception area for small targets is wider. The second is the Excitation stage. Afterward obtaining the $1 \times 1 \times C$ vector of Squeeze, a fully connected layer is used and the primacy of each channel is predicted. Subsequently, it is applied to the channel corresponding to the initial feature map so that the feature information of the small target will be given a higher priority. The structure of SENet is shown in Figure 4.

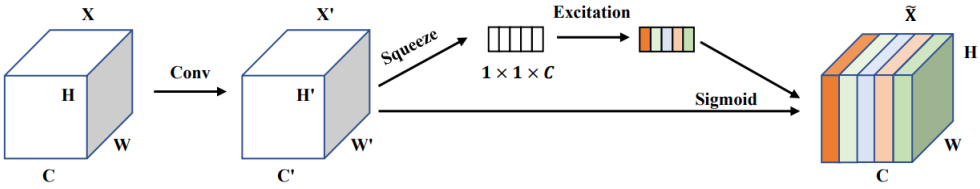

Figure 4. The figure shows the network structure of SENet, which is the graphical representation of the Squeeze stage and Exception stage mentioned above.

In the deep learning network model, the activation function is a continuous and derivable nonlinear function that can fit a nonlinear relationship. The form of the activation function and its derivative is relatively simple, which can speed up the learning speed of the network. A properly used activation function is of great significance to both the training of the model and the accuracy of the model's prediction of the target. The derivative of the activation function should not be too large or too small, too large will cause the gradient to explore, while too small will cause the gradient to disappear, and it is best to stabilize around 1. The model proposed in this paper uses the Swish activation function, the expression is as follows:

$$f(x) = x \times sigmoid(\beta x)$$

Among them, β is a constant or trainable parameter. Swish has the characteristics of no upper bound, lower bound, smooth, and non-monotonic. Swish is better than ReLU on the deep model. As the saturation of the Sigmoid function easily leads to the disappearance of the gradient, drawing on the effect of ReLU, when it is very large, it will approach at this time, but when $x \to \infty$, the general trend of the function is similar to ReLU but more complicated than ReLU. The Swish function can be regarded as a smooth function between the linear function and the ReLU function.

2.3. Performance Indicators

This paper uses the trained Fire-YOLO model to conduct a series of experiments on the test images to verify the performance of the algorithm. The relevant indicators for evaluating the effectiveness of the neural network model are as follows: precision, recall, F1, and AP values.

For the binary classification problem, according to the combination of the true category and the predicted category, samples can be divided into four types: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). The confusion matrix of the classification results is shown in Table 1:

Table 1. The content of the table is the confusion matrix of the real and predicted categories for dichotomous problems.

Labeled Name	Predicted	Confusion Matrix
Positive	Positive	TP
Positive	Negative	FN
Negative	Positive	FP
Negative	Negative	TP

Precision and recall are defined as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

The precision–recall curve (P–R curve) is obtained by taking the precision ratio as the vertical axis and the recall rate as the horizontal axis. The F1 score is also used to evaluate the performance of the model. The definition of F1 score is as follows:

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

This study also used another evaluation index of object detection-average accuracy (AP). It is defined as follows:

$$\text{AP} = \int_1^0 \text{Precision}(\text{Recall}) d(\text{Recall})$$

3. Experimental Analysis

This section introduces the experimental environment, dataset, evaluation indicators of model effect, and analysis of experimental results of the training network.

Through a series of comparative experiments on different models, the superiority of the new model proposed in this paper is analyzed. A fire dataset and small target dataset were used in the experiment process. The accuracy of the target detection network is verified and evaluated through the fire dataset, and it has a good detection effect in complex environments, such as the detection of different light conditions and fire-like smoke targets; the detection is verified on the small target fire dataset. The results confirmed that Fire-YOLO is easier to detect for smaller targets. The Fire-YOLO detection model received 416×416 pixel image as input, since the GPU performance limitations, the batch size is set to 8, each model train 100 epochs, the initial learning rate is 10^{-3}, and it will be divided by 10 after 50 epochs.

3.1. Dataset Acquisition

The dataset used in the experiment was constructed by collecting fire pictures from the fire protection public welfare platform. The fire dataset and the small target dataset are divided into the training set, validation set, and test set, respectively so that the different models can be trained under the same experimental settings. The following is a detailed description of the two datasets:

3.1.1. Fire Dataset

The image data used in this article are fire and smoke images collected on public websites. The original images of 19,819 include images of flames and smoke in different weather and light lines.

After numbering the above dataset images, we use the LabelImg tool to manually label, including drawing bounding boxes and classification categories. Taking into account

the correspondence between labels and data, to ensure that the dataset is evenly distributed, the dataset is randomly divided into the training set, validation set, and test set according to the ratio of 70%, 20%, and 10%. In order to ensure the same experimental environment, the final dataset is stored in the PASCAL VOC dataset format. Positive samples with unclear pixel regions are not labeled in order to prevent overfitting in the neural network, The completed dataset is shown in Table 2.

3.1.2. Small Target Dataset

We made a self-made dataset of 370 images. The contents of the dataset are all flames and smoke containing small targets. By embedding a fire image with a size of 250 × 250 pixels into an image with a size of 1850 × 1850 pixels, the area of the detected target in the image is very small. Finally, we use the LabelImg tool to manually label small targets.

Table 2. The table shows the number of pictures and labels on the training set, test set, and validation set, respectively. The proportions of the three are 70%, 20%, and 10%.

Dataset	Training Set	Validation Set	Test Set	Total Number
Number of images	13,873	3964	1982	19,819
Number of annotated samples	28,031	8009	4004	40,044

3.2. Algorithm Comparison Analysis

In order to verify the performance of the model proposed in this paper, images of flame and smoke are used as the training set. The proposed model is compared with YOLO-V3 and Faster R-CNN detection methods.

The P–R curves of the three models in the testing process are shown in Figure 5. The accuracy, recall, F1 score, and mAP values are shown in Table 3.

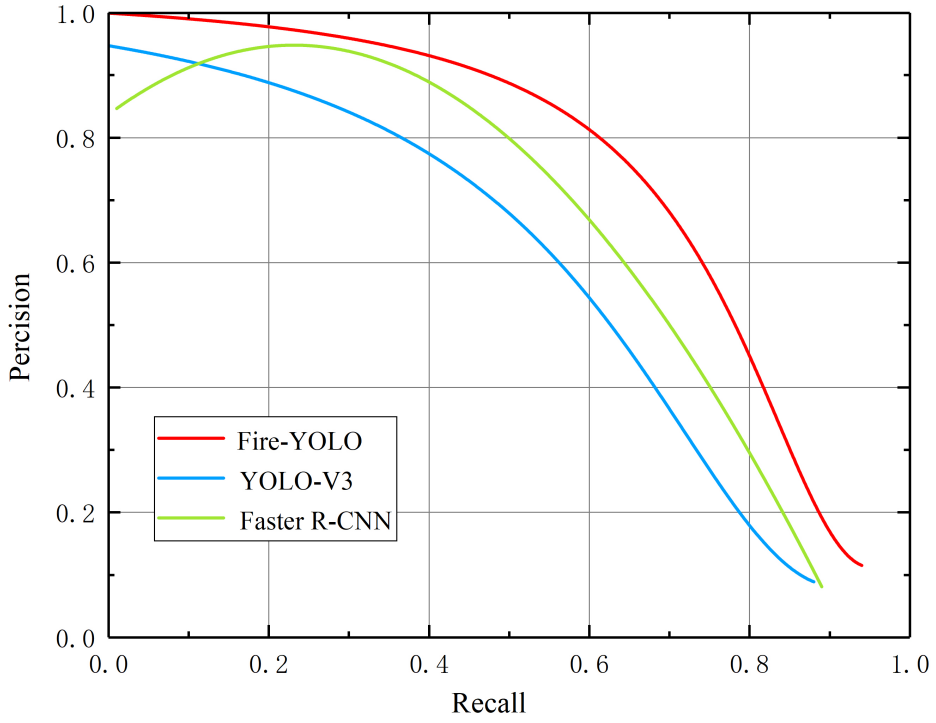

Figure 5. The picture shows the P–R curve of the Fire-YOLO model proposed in this paper and YOLO-V3 and Faster R-CNN during the test.

The proposed Fire-YOLO is superior to YOLO-V3 and Faster R-CNN in terms of detection performance based on the above results. The accuracy of the Fire-YOLO model

is 0.915, the value of F1 is 0.73, and the value of mAP is 0.802, which is higher than the other two models and reflects the superiority of this model. At the same time, Fire-YOLO reduces computing costs, saves resources, and is more conducive to socially sustainable development.

Table 3. The table shows the comparison between the model proposed in this article and YOLOV3 and Faster R-CNN on the training set in terms of accuracy, recall, F1 score, mAP, and model size.

	Faster R-CNN	YOLO-V3	Fire-YOLO
Precision	58.17%	88.92%	91.50%
Recall	81.19%	55.65%	59.62%
F1	51.50%	68.50%	73.00%
mAP	67.08%	73.69%	80.23%
Model Size	108 MB	234 MB	62 MB

3.3. Detection Performance of Small Targets

In the fire detection process, since the camera is too far away from the fire place, the actual fire location occupies a small area in the captured image, which will cause the network model to detect flames and smoke very poorly. By comparing the accuracy, recall, and mAP of three different target detection models on the small target fire dataset, it can be obtained that the proposed Fire-YOLO model has a better detection efficiency for very small target objects than Faster R-CNN and unimproved YOLO-V3 network. The Fire-YOLO trained on the fire small target dataset makes adaptive adjustments in the three dimensions of depth, width, and resolution for the small target images to be detected, which strengthens the interaction between information. Thereby, the ability of Fire-YOLO to extract small target features is strengthened, and the detection accuracy of small target objects is improved. Table 4 shows the specific results of the evaluation indicators of the three model methods on the small target fire dataset.

Table 4. The table shows the comparison between the model proposed in this article and YOLOV3 and Faster R-CNN on the small target fire dataset in terms of accuracy, recall, and mAP.

	Faster R-CNN	YOLO-V3	Fire-YOLO
Precision	29.83%	53.71%	75.48%
Recall	15.70%	29.50%	27.29%
mAP	10.36%	28.10%	39.50%

The detection efficiency of the three different models on the small target fire dataset are quite different. The accuracy and recall rate of the Fire-YOLO model we proposed are more significant compared with other models. The accuracy rate can reach 75.48%, which can realize the detection of small targets. Timely detection in the early stage of a forest fire can greatly reduce the damage to the ecological environment, reduce the economic loss caused by fire, and promote the sustainable development of the ecological environment.

The trained Fire-YOLO network model has a good efficiency in detecting fire targets. We use the Fire-YOLO model to detect very small fire targets and display the detection results graphically. The smaller fire targets are all images from the validation dataset in the small target fire dataset. Fire-YOLO was verified in more than 30 verification illustrations, and the final image detection results are shown in Figure 6. Fire-YOLO can detect all the fire and smoke in the picture, while YOLO-V3 and Faster R-CNN can only detect part of the target in the image in the detection results.

Figure 6. (**a**,**b**) Faster R-CNN, (**c**,**d**) YOLO-V3, (**e**,**f**) Fire-YOLO. The picture shows the results of the Fire-YOLO model proposed in this article and YOLO-V3 and Faster R-CNN to detect very small fire targets.

3.4. Detection Performance of Fire-Like and Smoke-Like Targets

Due to the particularity of detection targets such as flames and smoke, they often encounter the effects of lights similar to flames and the effects of white clouds similar to smoke in actual fire detection scenarios. The existence of these smoky fire targets will adversely affect the accuracy of model detection. In previous research work, most models focus on model optimization, such as better detection accuracy, lighter weight models, and faster detection speed [32–34]. Anyway, the smoke-like fire situation that exists in reality is not considered, and the Fire-YOLO model proposed in this paper detects the smoke-like fire situation.

By comparing the inspection performance of the models on abundant fire-like and smoke-like images, it can be found that Fire-YOLO has better detection efficiency on fire-like and smoke-like targets. Models other than Fire-YOLO misjudged the lights and clouds in the image as fire and smoke, respectively. Obviously, Fire-YOLO is more sensitive to the features of image texture, which is due to the combination of 1×1 convolution kernel and SE (Squeeze-and-Excitation) module in the feature extraction network. Eventually, the robustness of the proposed model in detecting confusing fire objects is improved. The fire-like and smoke-like detection results of the three models are shown in Figure 7. In the above cases, the Fire-YOLO model greatly reduces the occurrence of false detection, reduces labor consumption, and saves social resources.

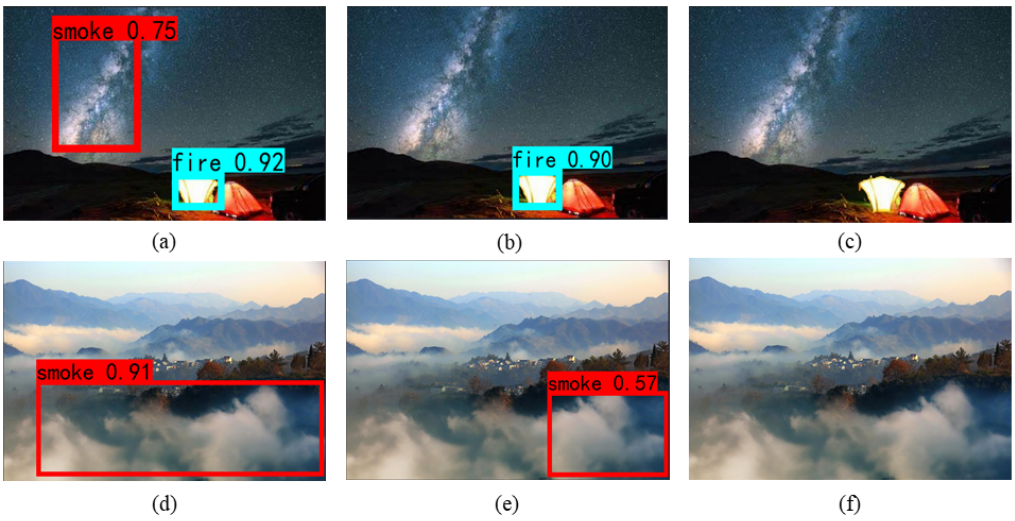

Figure 7. (**a**,**d**) Faster R-CNN, (**b**,**e**) YOLO- V3, (**c**,**f**) Fire-YOLO. The picture shows the detection effect of comparing the Fire-YOLO model with YOLOV3 and Faster R-CNN in the presence of fire-like and smoke-like targets.

3.5. The Detection Performance of the Model under Different Natural Lights

In this section, the performance of Fire-YOLO in the real environment is tested by comparing multiple fire images under different natural lights conditions. In the actual fire detection scene, there will be insufficient light or very strong light. In this kind of scene, it will have a certain impact on the fire detection. The large-scale feature map is used to improve the model to identify small target objects [35], yet there is a misjudgment in low light conditions. The detection results are shown in Figure 8. The detection results have shown that the model has impressive performance under different lighting conditions and is robust to light changes, by comparing the detection performance of Faster R-CNN, YOLO-V3, and Fire-YOLO models. Such advantages of the Fire-YOLO model can reduce the harm of fire to forests, reduce the impact of the greenhouse effect on human beings, and promote sustainable development.

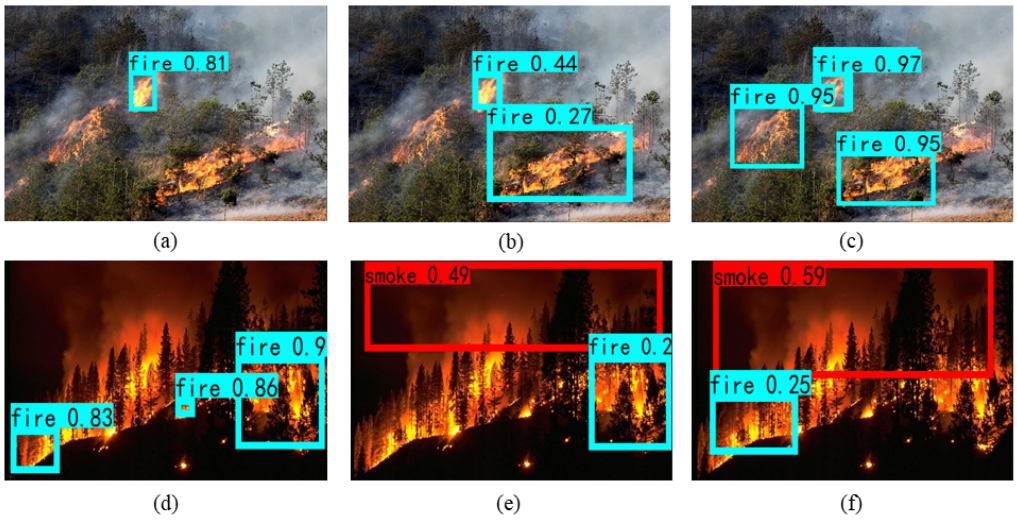

Figure 8. (**a**,**d**) Faster R-CNN, (**b**,**e**) YOLO-V3, (**c**,**f**) Fire-YOLO. The picture shows the fire image detection results of the Fire-YOLO model proposed in this article and YOLO-V3 and Faster R-CNN under different light conditions.

4. Discussion

In Sections 3.3–3.5, we discuss the performance of Fire-YOLO in different application scenarios. The Fire-YOLO model proposed in this paper has achieved gratifying results in small target, fire-like, smoke-like detection, and fire detection under different lightness. It can not only provide real-time detection but also has good robustness in practical applications. Nevertheless, we found that the detection algorithm still has the problem of low detection accuracy and challenging detection of semi-occluded targets in our tests. This phenomenon may be due to the variability of fire and the complexity of fire spread in the detection of flames in the actual environment, which creates a dilemma in fire inspection as shown in Figure 9. It is worth mentioning that this is also an urgent problem [36] to be solved by the current target detection model. What is encouraging is that these difficulties are not insurmountable. In fact, the corresponding deep learning training techniques can be flexibly selected in the detection algorithm, for instance, translation-invariant transformation [37], random geometric transformation [38], and random color dithering [39] on the training image in the training of the model, which shows promising results. In future work, the training process of the Fire-YOLO model will be optimized, and the focus will be on the preprocessing of images. It is hoped that the generalization ability of Fire-YOLO can be further improved through transfer learning.

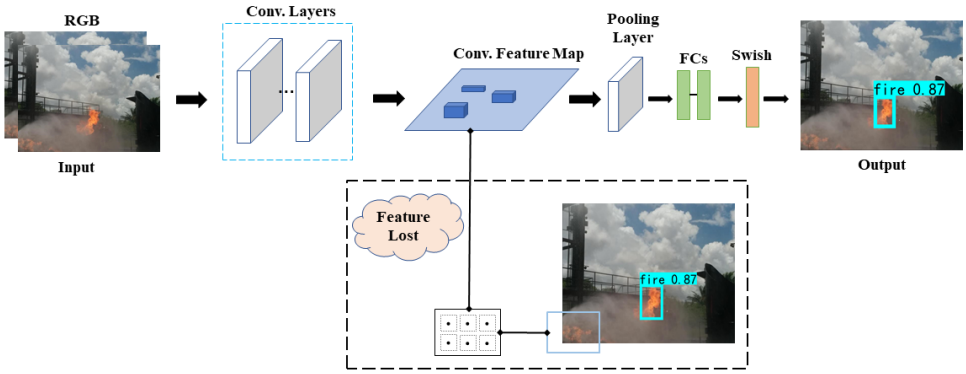

Figure 9. When the Fire-YOLO model detects the concealed flame target, it is easy to lose features in the process of model training, resulting in inaccurate detection results.

5. Conclusions

In the research of this paper, the YOLO-V3 detection model is improved by combining the EfficientNet method for fire detection. The newly proposed model can be used to detect flames and smoke. The Fire-YOLO model proposed in this paper uses EfficientNet to extract the features of the input image, which promotes the feature learning of the model, improves the network performance, and optimizes the detection process of the YOLO-V3 model for extremely small targets. The experimental results show that the Fire-YOLO model proposed in this paper has better performance than the YOLO-V3 model and is better than Faster R-CNN. The Fire-YOLO model can also detect fire targets in real-time. The fast and accurate detection of forest fires can reduce the economic loss caused by forest fires, improve the protection of forests and their ecological environment, and promote the sustainable development of resources.

Future work will focus on applying existing models to detect the size of the fire in the video and other practical tasks. In addition, the detection and model of small targets will be optimized to further improve the accuracy of the detection model.

Author Contributions: L.Z. (Lei Zhao), L.Z. (Luqian Zhi), C.Z., and W.Z. designed the project. L.Z. (Lei Zhao) performed the experiment and analyzed the data. L.Z. (Lei Zhao), L.Z. (Luqian Zhi), C.Z., and W.Z. wrote the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by National Natural Science Foundation of China, Grant No. 11702289, Key core technology and generic technology research and development project of Shanxi Province, No. 2020XXX013, and the National Key Research and Development Project.

Institutional Review Board Statement: Institute of Public-Safety and Big Data, College of Data Science, Taiyuan University of Technology, Taiyuan 030024, China. Center of Information Management and Development, Taiyuan University of Technology, Taiyuan 030024, China. Center for Healthy Big Data, Changzhi Medical College, Changzhi 046000, China.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in the experiment was constructed by collecting fire pictures from the fire protection public welfare platform. The original 19,819 images include images of flames and smoke in different weather and light lines. We made a self-made dataset of 370 images. The contents of the dataset are all flames and smoke containing small targets.

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

References

1. Goyal, S.; Shagill, M.D.; Kaur, A.; Vohra, H.; Singh, A. A YOLO based Technique for Early Forest Fire Detection. *Int. J. Innov. Technol. Explor. Eng. (IJITEE) Vol.* **2020**, *9*, 1357–1362. [CrossRef]
2. Muhammad, K.; Ahmad, J.; Baik, S.W. Early fire detection using convolutional neural networks during surveillance for effective disaster management. *Neurocomputing* **2018**, *288*, 30–42. [CrossRef]
3. Premal, C.E.; Vinsley, S.S. Image processing based forest fire detection using YCbCr colour model. In Proceedings of the 2014 International Conference on Circuit, Power and Computing Technologies, Nagercoil, India, 20–21 March 2014; pp. 1229–1237.
4. Wu, W.; Liu, H.; Li, L.; Long, Y.; Wang, X.; Wang, Z.; Li, J.; Chang, Y. Application of local fully Convolutional Neural Network combined with YOLO v5 algorithm in small target detection of remote sensing image. *PLoS ONE* **2021**, *16*, e0259283. [CrossRef] [PubMed]
5. Habiboğlu, Y.H.; Günay, O.; Çetin, A.E. Covariance matrix-based fire and flame detection method in video. *Mach. Vis. Appl.* **2012**, *23*, 1103–1113. [CrossRef]
6. Lascio, R.D.; Greco, A.; Saggese, A.; Vento, M. Improving fire detection reliability by a combination of videoanalytics. In *International Conference Image Analysis and Recognition*; Springer: Cham, Switzerland, 2014; pp. 477–484.
7. Zhao, C.; Feng, Y.; Liu, R.; Zheng, W. Application of Lightweight Convolution Neural Network in Cancer Diagnosis. In Proceedings of the 2020 Conference on Artificial Intelligence and Healthcare, Taiyuan, China, 23–25 October 2020; pp. 249–253.
8. He, L.; Gong, X.; Zhang, S.; Wang, L.; Li, F. Efficient attention based deep fusion CNN for smoke detection in fog environment—ScienceDirect. *Neurocomputing* **2021**, *434*, 224–238. [CrossRef]
9. Gagliardi, A.; Villella, M.; Picciolini, L.; Saponara, S. Analysis and Design of a Yolo like DNN for Smoke/Fire Detection for Low-cost Embedded Systems. In *International Conference on Applications in Electronics Pervading Industry, Environment and Society*; Springer: Cham, Switzerland, 2020; pp. 12–22.
10. Jindal, P.; Gupta, H.; Pachauri, N.; Sharma, V.; Verma, O.P. Real-Time Wildfire Detection via Image-Based Deep Learning Algorithm. In *Soft Computing: Theories and Applications. Advances in Intelligent Systems and Computing*; Sharma, T.K., Ahn, C.W., Verma, O.P., Panigrahi, B.K., Eds.; Springer: Singapore, 2021; Volume 1381. [CrossRef]
11. Zhang, W.; Wei, J. Improved YOLO v3 Fire Detection Algorithm for Embedded DenseNet Structure and Hollow Convolutional module. *J. Tianjin Univ. (Nat. Sci. Eng. Technol. Ed.)* **2020**, *53*, 976–983.
12. Li, J.; Guo, S.; Kong, L.; Tan, S.; Yuan, Y. An improved YOLOv3-tiny method for fire detection in the construction industry. In *E3S Web of Conferences*; EDP Sciences: Les Ulis, France, 2021; p. 253.
13. Yue, C.; Ye, J. Research on Improved YOLOv3 Fire Detection Based on Enlarged Feature Map Resolution and Cluster Analysis. *J. Phys. Conf. Ser.* **2021**, *1757*, 012094. [CrossRef]
14. Qin, Y.Y.; Cao, J.T.; Ji, X.F. Fire Detection Method Based on Depthwise Separable Convolution and YOLOv3. *Int. J. Autom. Comput.* **2021**, *18*, 300–310. [CrossRef]
15. Cheng, X.; Qiu, G.; Jiang, Y.; Zhu, Z. An improved small object detection method based on Yolo V3. *Pattern Anal. Appl.* **2021**, *24*, 1347–1355. [CrossRef]
16. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
17. 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.
18. 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. 6517–6525.
19. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the International Conference on Neural Information Processing Systems 28, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.

20. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
21. Robert Singh, A.; Athisayamani, S.; Sankara Narayanan, S.; Dhanasekaran, S. Fire Detection by Parallel Classification of Fire and Smoke Using Convolutional Neural Network. In *Computational Vision and Bio-Inspired Computing*; Springer: Singapore, 2021; pp. 95–105.
22. Wang, Z.; Zhang, H.; Hou, M.; Shu, X.; Wu, J.; Zhang, X. A Study on Forest Flame Recognition of UAV Based on YOLO-V3 Improved Algorithm. In *Recent Advances in Sustainable Energy and Intelligent Systems (LSMS 2021, ICSEE 2021)*; Communications in Computer and Information Science; Li, K., Coombs, T., He, J., Tian, Y., Niu, Q., Yang, Z., Eds.; Springer: Singapore, 2021; Volume 1468._47. [CrossRef]
23. Hou, F.; Zhang, Y.; Fu, X.; Jiao, L.; Zheng, W. The Prediction of Multistep Traffic Flow Based on AST-GCN-LSTM. *J. Adv. Transp.* **2021**, *2021*, 9513170. [CrossRef]
24. Zhang, Y.; Ren, J.; Wang, R.; Fang, F.; Zheng, W. Multi-Step Sequence Flood Forecasting Based on MSBP Model. *Water* **2021**, *13*, 2095. [CrossRef]
25. Tan, M.; Le, Q.V. EfficientNet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019; p. 2.
26. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation. *arXiv* **2018**, arXiv:1801.04381v2.
27. Jie, H.; Li, S.; Gang, S.; Wu, E. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
28. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Wey, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
29. Zhao, C.; Zheng, W. Fast Traffic Sign Recognition Algorithm Based on Multi-scale Convolutional Neural Network. In Proceedings of the 2020 Eighth International Conference on Advanced Cloud and Big Data (CBD), Taiyuan, China, 5–6 December 2020; pp. 125–130.
30. Wang, R.; Fang, F.; Cui, J.; Zheng, W. Learning self-driven collective dynamics with graph networks. *Sci. Rep.* **2022**, *12*, 500. [CrossRef] [PubMed]
31. Zheng, W.; Zhang, S.; Xu, N. Jamming of packings of frictionless particles with and without shear. *Chin. Phys. B* **2018**, *27*, 066102. [CrossRef]
32. Zhang, X.; Qian, K.; Jing, K.; Yang, J.; Yu, H. Fire Detection based on Convolutional Neural Networks with Channel Attention. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 3080–3085. [CrossRef]
33. Saponara, S.; Elhanashi, A.; Gagliardi, A. Real-time video fire/smoke detection based on CNN in antifire surveillance systems. *J. Real-Time Image Proc.* **2021**, *18*, 889–900. [CrossRef]
34. Li, W.; Yu, Z. A Lightweight Convolutional Neural Network Flame Detection Algorithm. In Proceedings of the 2021 IEEE 11th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 18–20 June 2021; pp. 83–86.
35. Abdusalomov, A.; Baratov, N.; Kutlimuratov, A.; Whangbo, T.K. An Improvement of the Fire Detection and Classification Method Using YOLOv3 for Surveillance Systems. *Sensors* **2021**, *21*, 6519. [CrossRef] [PubMed]
36. Muhammad, K.; Ahmad, J.; Mehmood, I.; Rho, S.; Baik, S.W. Convolutional Neural Networks Based Fire Detection in Surveillance Videos. *IEEE Access* **2018**, *6*, 18174–18183. [CrossRef]
37. Luo, D.; Wang, D.; Guo, H.; Zhao, X.; Gong, M.; Ye, L. Detection method of tubular target leakage based on deep learning. In Proceedings of the Seventh Symposium on Novel Photoelectronic Detection Technology and Application, Kunming, China, 5–7 November 2020; Volume 11763, p. 1176384.
38. Mumuni, A.; Mumuni, F. CNN architectures for geometric transformation-invariant feature representation in computer vision: A review. *SN Comput. Sci.* **2021**, *2*, 1–23. [CrossRef]
39. Kayhan, O.S.; Gemert, J.C. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 14274–14285.