Article

Efficient Smoke Detection Based on YOLO v5s

Hang Yin, Mingxuan Chen, Wenting Fan, Yuxuan Jin, Shahbaz Gul Hassan * and Shuangyin Liu *

College of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China
* Correspondence: mhasan387@cau.edu.cn (S.G.H.); hdlsyxlq@126.com (S.L.)

Abstract: Smoke detection based on video surveillance is important for early fire warning. Because the smoke is often small and thin in the early stage of a fire, using the collected smoke images for the identification and early warning of fires is very difficult. Therefore, an improved lightweight network that combines the attention mechanism and the improved upsampling algorithm has been proposed to solve the problem of small and thin smoke in the early fire stage. Firstly, the dataset consists of self-created small and thin smoke pictures and public smoke pictures. Secondly, an attention mechanism module combined with channel and spatial attention, which are attributes of pictures, is proposed to solve the small and thin smoke detection problem. Thirdly, to increase the receptive field of the smoke feature map in the feature fusion network and to solve the problem caused by the different smoke scenes, the original upsampling has been replaced with an improved upsampling algorithm. Finally, extensive comparative experiments on the dataset show that improved detection model has demonstrated an excellent effect.

Keywords: smoke detection; small and thin; lightweight network；attention mechanism; improved upsampling algorithm

MSC: 68T07

1. Introduction

Fire frequently occurs in human daily life, bringing great annoyance. Life and property may suffer from significant losses as a result at the same time. According to human common consensus, smoke appears both before the fire and appears with the fire. Therefore, detecting smoke quickly is one of the most important fire warning directions. However, traditional smoke sensors are only used for detections indoors and have large requirements for the concentration and size of smoke. Fortunately, with the development of technology [1–3], there are many smart monitoring devices, and IoT detection is on the agenda. However, most smoke detection does not focus on the small and thin smoke in the early stage of the fire. Moreover, very small and thin smoke datasets can be found in the early stages of a fire. Given the above problems, detecting small and thin smoke in the early stage of intelligent fire based on real-time monitoring is of great significance.

This paper is described later as follows. Section 2 contains a consideration of related work. In Section 3, methods of the paper are introduced. Section 4 presents the results of the experiments. In Section 5, a conclusion and future work are depicted.

2. Related Work

Existing methods of intelligent smoke detection have been collected and summarized by us. After comparison, the methods can be divided into two categories: smoke detection based on traditional machine learning and smoke detection based on deep learning convolutional neural network feature extraction. Based on traditional machine learning, there

are usually three steps: (1) using foreground extraction or subregional interception of images; (2) followed by the design of digital image processing according to distinctive features; (3) input the numerical features to the machine learning classifier. In [4], the types of feature extraction algorithms commonly used in machine learning are detailed. Most of the existing algorithms are based on color [5–7], texture [8–10], motion features [11,12], and shape [13]. In [6], a color segmentation method was used to classify the smoke moving pixel points successfully. In [7], the authors combined LBP, KPCA, and GPR to propose a new smoke detection channel. Therefore, the smoke was classified according to its texture. Although the smoke constantly moves, this feature was also captured in [9]. Firstly, the area where the smoke was located was known through the color distribution rules, and then the motion energy was used to estimate the saliency map of the pre-segmentation, and the accurate smoke segmentation result is obtained. An innovative method for segmenting the smoke region was proposed to classify smoke pixels based on smoke's color and motion features [11]. In [13], although only pedestrians were detected, it was quite close to the method for detecting smoke. This paper mentioned that the optimal hyperplane is obtained through the distribution in the multi-channel image space. Then, the pedestrian was segmented by shape statistics. This method could also be used in smoke detection. Although traditional detection methods detect smoke more accurately, they can be much slower in terms of efficiency than deep learning methods. Traditional machine learning methods require subjective judgments on the features to be extracted, and they often tend to lack large data to support the diversity of smoke features caused by different environments at different times. Moreover, when detecting small smoke, the accuracy needs to be improved. Therefore, people now prefer to use deep learning convolutional neural networks for smoke detection.

Many approaches have also emerged in smoke detection using deep learning convolutional neural networks since the introduction of AlexNet [14], one of the convolutional neural networks. In [1,15], the VGG16 convolutional neural network [16] was used as the backbone network of model detection and improved accordingly. In order to address the identification of smoke in haze and improve the robustness of the network model, an artificial smoke dataset was also used in [1]. Moreover, the authors use ImageNet to pretrain the weights before training their dataset, thus solving the problem of interference caused by the natural environment, as expected. Furthermore, in [15], the authors also used VGG16 as the backbone network [16] and added spatial and channel attention mechanisms. Finally, feature-level and decision-level fusion models were added to reduce the model parameters. Therefore, it reduced the size of the model and improved the accuracy.

Moreover, [17] also studied smoke detection in haze weather. A dark channel-assisted, hybrid attention, and feature fusion algorithm was proposed. An unbalanced data set was trained first, improving smoke detection accuracy in a haze environment. In addition, to solve the large deformation of smoke shape in the case of large outdoor wind speed, [18] proposed a cascade classification of smoke and a deep convolutional neural network based on AlexNet to improve smoke detection in some extreme environments. In both [19,20], the authors included a BN (batch normalize) [21] layer, which aimed to unify the scattered data and normalize the data in each layer, thus achieving a training model acceleration as well as overfitting mitigation. Dual deep convolutional neural networks, DCNN and SBNN, were used in [19]. The authors added BN layers to both networks to detect smoke accurately. The role of the SBNN network was to extract detailed information about smoke, and the role of the SCNN was to capture the basic information about the smoke. Finally, the ninth max-pooling layer of the SBNN network was removed and connected to the feature fusion to achieve the dual network connection. In [20], the data set was first preprocessed by detecting the dynamic track of smoke, and the suspicious smoke area was obtained. Next, the SqueeezeNet lightweight convolutional network [22] was used for feature extraction. It is worth noting that the authors used a three-network progressively improved SqueezeNet network, a network with BN layers, and a depth-wise separable convolution instead of the traditional convolutional network. In

[23], to reduce the detection difficulty and real-time detection monitoring, the existing convolutional neural was modified and a new convolutional neural network SCCNN was proposed to get good results for real-time smoke detection. In [24], the authors adopted the lightweight object detection network Efficientdet-D2 [25]. The problem of false-negative detection results caused by insufficient consideration of scene information in actual smoke scenes was solved by adding a self-attention mechanism to the network. Furthermore, the problem of false detection caused by class smoke was solved by successfully obtaining multi-level nodes for a multi-feature fusion of smoke.

In general, the deep learning smoke detection methods proposed above all have achieved excellent smoke detection results. However, there are two main problems with smoke detection. (1) In [1,15,18], the authors mainly solve the problem of smoke in the haze environment. Nevertheless, there is less corresponding work for detecting small and thin smoke in the early stage of the fire. This is one of the most effective and rapid ways to prevent fire occurrence. (2) The detection algorithm can be more lightweight because Edge computing based on lightweight algorithms is the trend in deep learning. Using a more lightweight detection model smoke was detected faster without internet interference. Therefore, there are the following challenges: (1) Deep learning relies heavily on an effective dataset and finding an effective small and thin smoke dataset in the early stage of fire from the internet are difficult. (2) Detections for small and thin are more difficult than normal smoke detection because they are located in small areas and carry less information. (3) The accuracy of smoke detection using a lightweight model is usually lower than that of a large network model.

In order to solve the problems raised above, an improved YOLO v5s CNN network based on the spatial and the channel attention mechanisms and replacing the original upsampling content-aware reassembly of features (CARAFE) is proposed by us. Firstly, smoke video was shoot from an empty warehouse through cameras. Frame-by-frame screenshots have been taken to get the initial dataset and use public datasets to enhance the scene robustness of data. Secondly, a channel and spatial attention mechanism is added after the first two C3 convolutions in the feature fusion network [26]. A novel upsampling CARAFE [27] instead of the first one in the feature fusion network upsampling was used. The proposed smoke detection framework is shown in Figure 1. Specifically,

- In order to solve the problem of being less small and thin smoke in the early stage of fire, practical shooting was carried out to collect real-time dataset. Smoke generators, sheets, and cotton ropes were used during the shooting as different small and thin smoke sources.
- A combination of spatial attention mechanism and channel attention mechanism network has been added to the feature fusion network of YOLO v5s to solve the problem of small and thin smoke detection at the beginning of a fire. Instead of a single spatial attention module, a combination of spatial and channel modules has been used. When extracting smoke, this will assign a higher weight to the area where the smoke is located and the channel. Therefore, the model can pay more attention to the smoke itself to reduce the interference of the scene and solve the problem of such small and thin smoke detection.
- In order to further improve the detection effect of smoke, the improved upsampling CARAFE has been taken in the feature fusion network of YOLO v5s instead of the original nearest neighbor interpolation upsampling. Compared with the nearest neighbor interpolation upsampling, the CARAFE algorithm can obtain a larger receptive field in the smoke photo to aggregate information. The contents of the smoke pictures are perceptually processed by generating adaptive kernels in real-time. Moreover, the algorithm has fewer parameters and a faster calculation speed, which is suitable for purpose of detecting smoke.
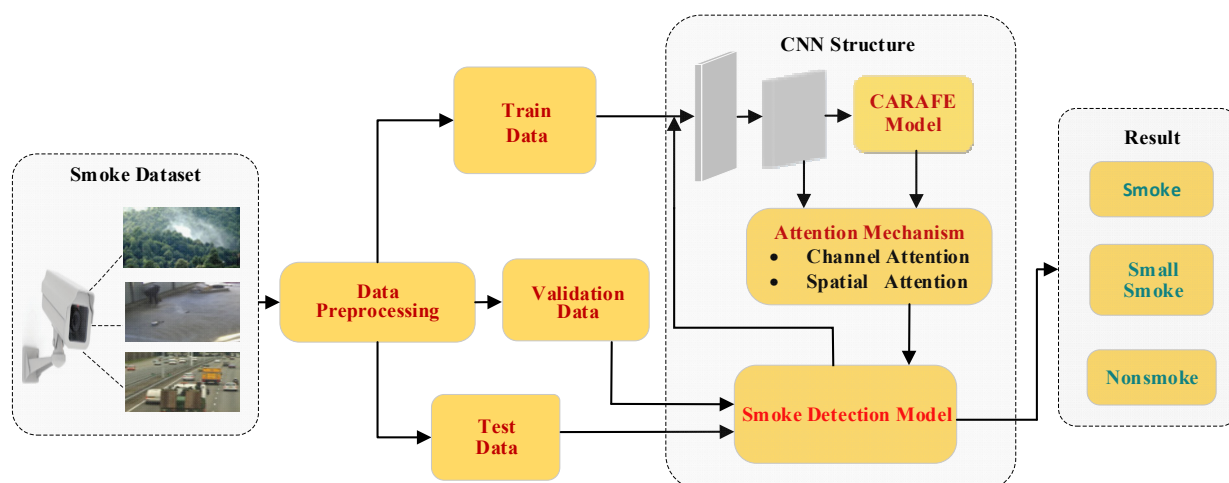
Figure 1. Overall framework.

3. Methods

An efficient smoke detection method that combines an attention mechanism with a novel upsampling algorithm has been proposed to address small and thin smoke detection in the early stage of fire. As seen from Figure 1, firstly, offline videos of common combustibles were taken just as they were starting to burn. Then, the original video data were preprocessed to obtain data in the form of images. The normal smoke dataset in different scenarios and the smoke-free dataset in different scenarios were combined to form a complete smoke dataset, divided into training, validation, and test datasets. Next, based on the YOLO v5s network, the structure of the convolutional neural network was redesigned, and an attention mechanism combined with spatial and channel has been added. The attention mechanism is used after the C3 module in feature fusion. Next, the first upsampling of the feature fusion network has been replaced with a novel upsampling CARAFE. Then, the training set was used to train the new convolutional neural network to obtain the smoke detection model. Finally, the smoke detection model types are normal smoke, small smoke, and non-smoke.

3.1. YOLO v5 Object Detection Network

Since Joseph Redmon proposed the YOLO [28] object detection algorithm in 2016, there has been increasing acceptance of this single-stage object detection algorithm. Compared with Fast RCNN [29], Faster RCNN [30] and other two-stage object detections, the YOLO series may be inferior to them in terms of detection accuracy. However, the significant reduction in the size of network parameters and the significant increase in detection speed will make it more suitable for real-time target detection. So far, the original author has continued to propose YOLO v2 [31], YOLOv3 [32] versions whereas Alexey Bochkovskiy proposed YOLO v4 [33] and YOLO v5 series that have been updated and maintained on GitHub. These YOLO series object detection algorithms continue improving target detection accuracy. It would be a suitable choice for us to use for smoke detection in real-time. Therefore, the more lightweight version s in YOLO v5s was chosen to conduct experiments and improve its performance to obtain higher object detection evaluation indicators.

As can be seen in Figures 2 and 3, the model of YOLO v5s removes the region proposal network and significantly improves the detection speed compared with the two-stage algorithm mentioned above. The model of YOLO V5 consists of three parts, namely Backbone Network, Neck Network, and Detect. Backbone Network is the most important part of the overall network. Because it takes a critical role in feature extraction of smoke pictures, which is an initial part of the network. The role of the Neck Network is to fuse

features from the Backbone Network. The part of Detect can create a bounding box (location box) to detect smoke. YOLO V5 is configured with four performance models of different sizes. The parameters are arranged from low to high as YOLO v5s, YOLO V5m, YOLO V5l, and YOLO V5x. YOLO V5 uses CSPDarknet53 as a feature extraction network [34]. The feature fusion neck network is composed of an integrated spatial pyramid pooling fast (SPPF) network, feature pyramid networks (FPN) [35], and pixel aggregation network (PAN) [36].
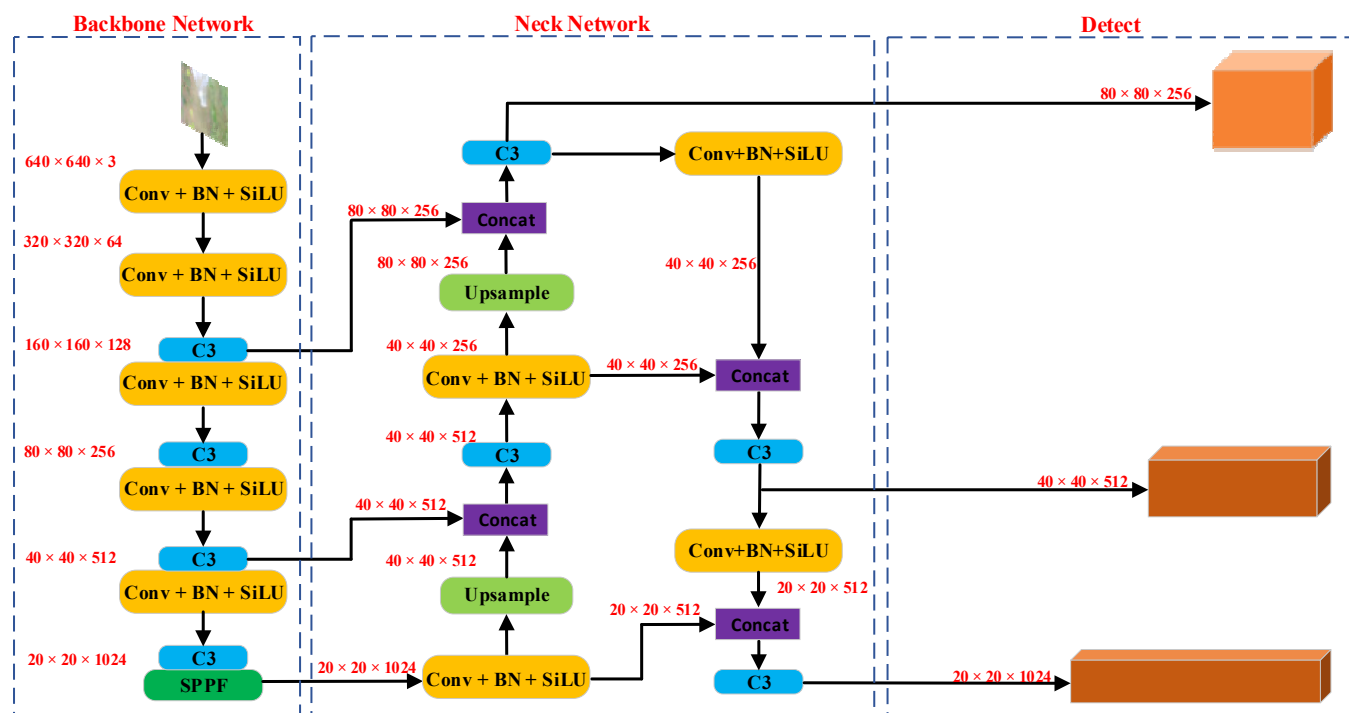
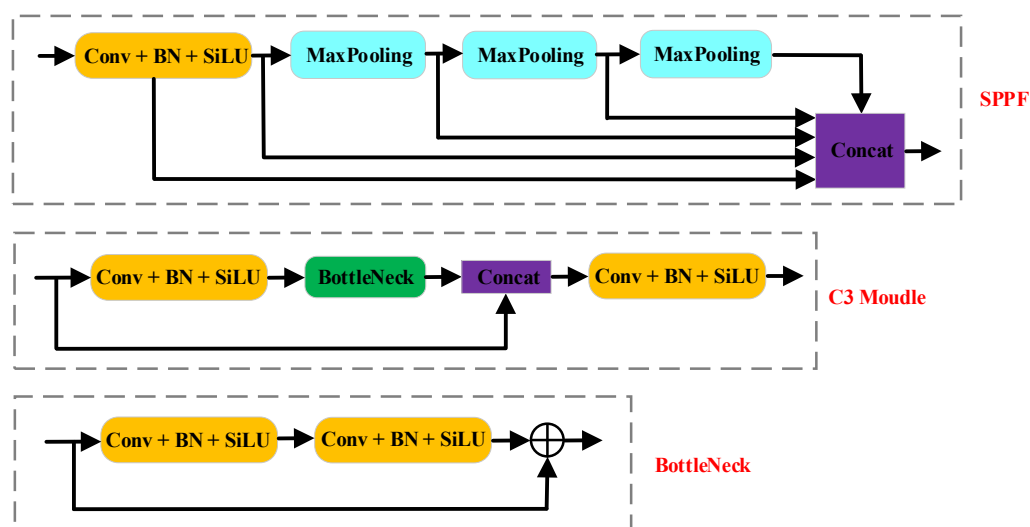**Figure 2.** YOLO v5s algorithm model (1).

Figure 3. YOLO v5s algorithm model (2).

The loss function of YOLO v5s consists of three parts, classification loss, localization loss, and confidence loss. Using binary cross entropy (BCE) loss, classification loss is used to indicate whether the anchor box matches the previously calibrated classification. Localization loss indicates the difference between the prediction and calibration frames using

Complete-IOU (C-IOU). Moreover, confidence loss also represents the confidence error of the network with BCE loss.

3.2. Small and Thin Smoke Detection Using Spatial and Channel Attention Mechanisms

The timely detection of small and thin smoke in the early stages of fire is a key factor affecting the ability to detect fires early and protect lives and property accurately. Since small objects are always located in small areas and carry little information, the features finally extracted by the multi-layer convolutional neural network are few compared with the background, resulting in weak feature expression ability and reduced detection ability for small objects. A combined channel and spatial attention mechanism module has been added to address this issue.

As shown in Figure 4, channel attention emphasizes the channel features of smoke, which give higher weights to object regions in the image. After the channel attention, the color of the photos input channel is more obvious. The weight of model becomes higher. Spatial attention emphasizes the location features representing the smoke and gives them higher weight. After the spatial attention algorithm, the location's color representing the smoke becomes darker. Therefore, in extracting smoke features, this attention mechanism can pay more attention to the location of smoke to reduce background interference and improve the accuracy of small and thin smoke detection.
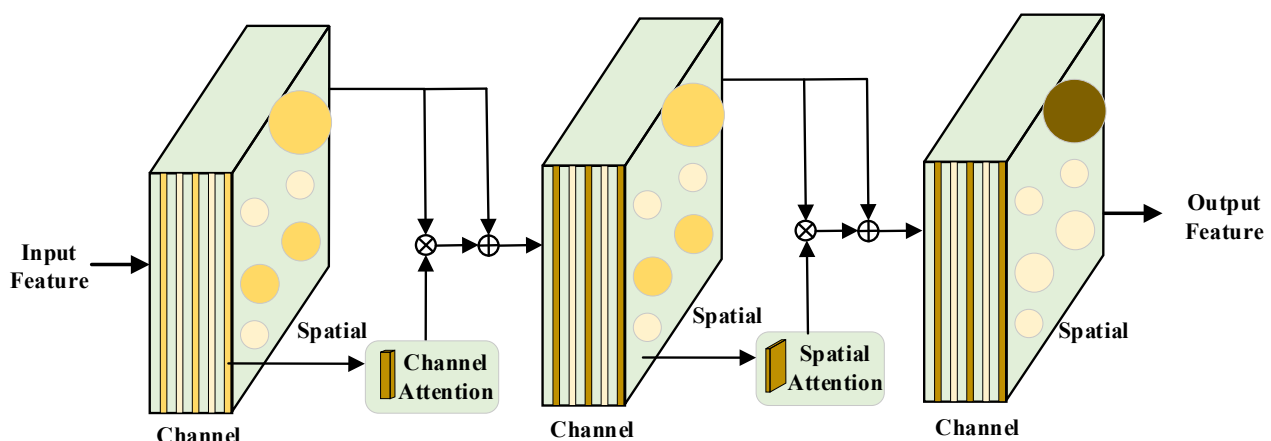

Figure 4. Attention mechanism.

Taking the output of C3 of the basic convolutional neural network YOLO v5s feature fusion network as the input feature X_i, the channel attention layer is composed of convolutional layers with W_1 and B_1 parameters, which represent the weight and bias of the convolutional layer, respectively. In the channel attention layer, average pooling and max pooling operations have been used to aggregate feature information. Then, feature extraction is performed through multiple fully connected layers. Finally, sigmoid activation has been used to generate the weights W_1 for each channel. When the feature X_i passes through the channel attention, the output feature X_{o1} is obtained, as

$$X_{o1} = X_i \ + \ (X_i \ \times \ W_1 \ + B_1) \tag{1}$$

Two pooling operations have been used consecutively in the spatial attention layer to aggregate the channel information from the X_{o1} feature maps. Next, feature extraction is performed through multi-layer convolution. Finally, the sigmoid activation generates the weighted spatial attention W_2. The final output feature X_{o2} obtained from the output feature X_{o1} after channel attention as input and B_2 is biased, as

$$X_{o2} \ = \ X_{o1} \ + \ (X_{o1} \ \times \ W_2 + B_2) \tag{2}$$

3.3. CARAFE Upsampling

In the original YOLO V5s, the feature fusion network upsampling is achieved by nearest-neighbor interpolation. However, nearest-neighbor interpolation only considers sub-pixel neighborhoods. It cannot capture the rich semantic information required for dense prediction tasks. In addition, deconvolution [37] is also one of the upsampling approaches. However, it also has two drawbacks: a deconvolution operator covers the same kernel throughout the image, regardless of the underlying content. This limits its ability to respond to local changes, and it comes with a large number of parameters. However, CARAFE has the attributes of a larger field of view, content-aware handing, lightweight, and fast to compute. Large field of view can receive more information of smoke pictures to complete the detection task. The attribute of content-aware handing uses adaptive kernels instead of the fixed kernel to better process different features of smoke. Lightweight and fast to compute, can detect smoke in real-time without increasing parameters much, which is the expectation of using a lightweight CNN.

The original upsampling has been changed to CARAFE up-sampling. As shown in Figure 5, CARAFE consists of two steps: the kernel prediction module and the content-aware reorganization module. In the kernel prediction module, the feature map of a given smoke image is C × H × W, and a convolution kernel with a 1 × 1 channel compression convolution C2 was performed. Then, to encode convolution, the number of channels were redistributed, where σ is the upsampling factor (assuming as an integer). Then, pixel shuffling is performed to expand the receptive field of upsampling for the smoke feature map. Next, the feature map is normalized to reduce the number of parameters in operation. Then, in the content-aware recombination module, the feature map obtained using the prediction kernel and the feature map obtained by ordinary upsampling are used for the dot product to reorganize the feature with the prediction kernel. Therefore, the formula of the kernel prediction module and the content-aware reorganization module are as follows:
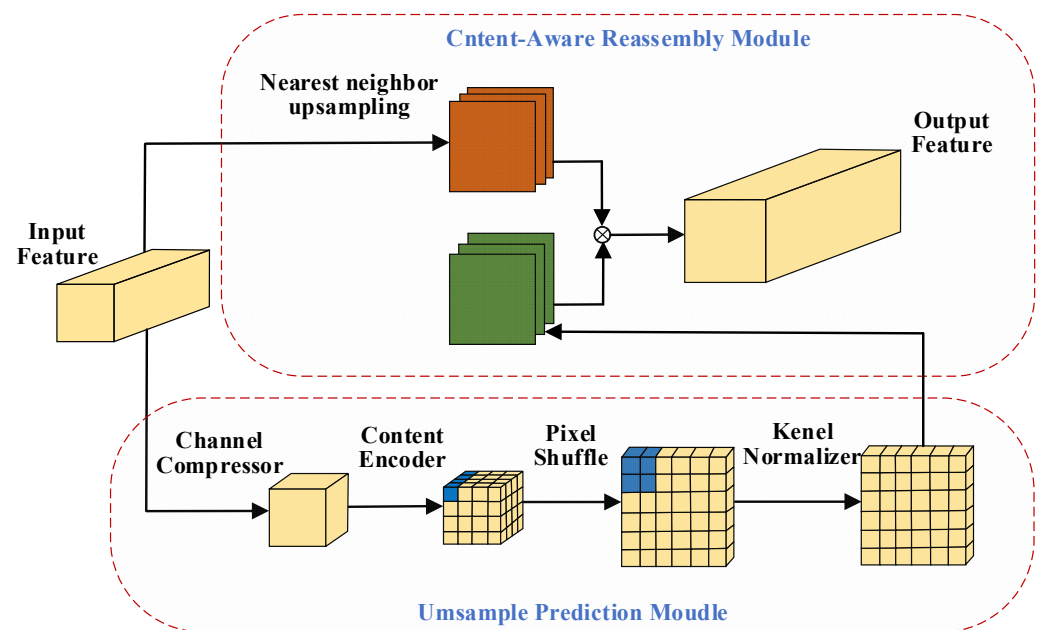

Figure 5. Flow of CARAFE.

$$K_{encoder} = k_{up} - 2 \tag{3}$$

$$W_o = \psi(N(X_1, K_{encoder})) \tag{4}$$

$$Z_o = \phi(N(X_1, k_{up}), W_o) \tag{5}$$

Among them, every target location requires a $k_{up} \times k_{up}$ reassembly kernel, where k_{up} is the reassembly kernel size. ψ represents the kernel prediction module, $K_{encoder}$ is the convolution kernel of the coding convolution, $N(X_l, k)$ is the X sub-region of X centered on this position, and W_o is the output of the prediction module. ϕ is the content-aware reorganization module, and Z_o is the total output of the upsampling model.

3.4. Innovative Datasets

A unique dataset was sought to address the lack of small and thin smoke in the early stages of fire in public datasets. These data include the smoke data of the smoke generator after the smoking sheet and the cotton rope is burned. The normal smoke and non-smoke pictures of the public dataset were combined to create a new dataset.

As shown in Figure 6, the smoke in the first photo of a column of small smoke of Figure 6 is from the smoke generator, the second photo from a smoke sheet, third photo from cotton rope. A total of 120 videos were collected, and the videos are divided into three types of smoke, namely the smoke from the smoke generator, the smoking sheet, and the cotton rope. The small and thin smoke dataset is unique. Because the small and light smoke images within 100×100 pixels from the high-definition video screenshots of 1080×1920 pixels were screened out, the smoke of the smoke generator is relatively uniform. Its smoking principle is to use the manual button to smoke, and the difficulty of smoke detection is relatively simple compared with the other two types of smoke. The smoke emitted by the cotton rope and the smoking sheet after burning will not be so obvious. Its initial smoke and smoke are relatively small, equivalent to the smoke of indoor objects that do note easily cause fires. Moreover, the smoke of cotton rope is particularly small, which is in line with the requirements of small and thin smoke in the early stage of fire.
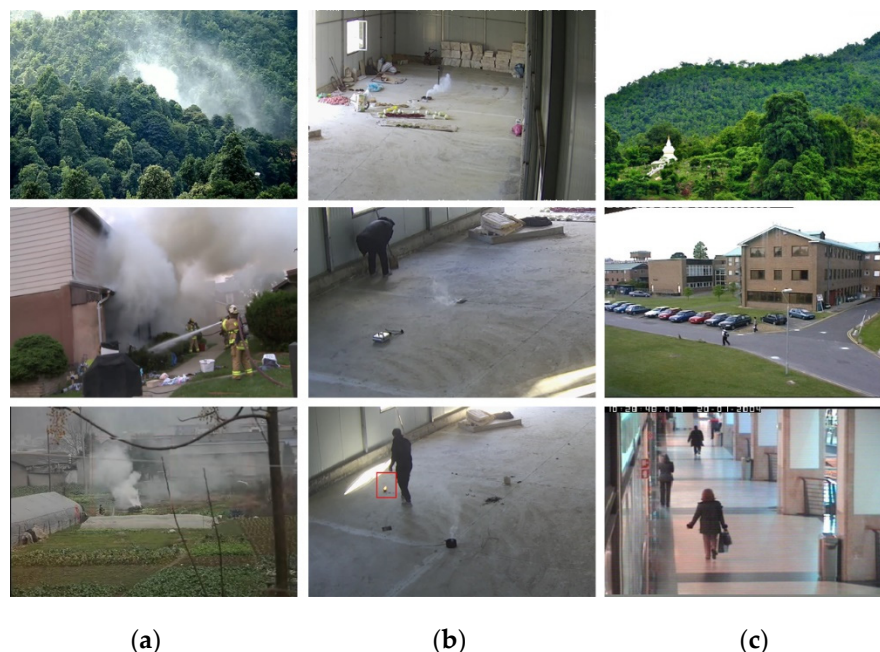

(**a**) (**b**) (**c**)

Figure 6. Dataset. (**a**) The smoke images were downloaded from the internet; (**b**) The small and thin images were created in a warehouse; (**c**) The non-smoke images were downloaded from the internet.

In addition, in Figure 6, photos are added of normal smoke and non-smoke from the internet for training to enhance the robustness of model, which will lead to an imbalance between smoke and non-smoke data. Moreover, as shown in Figure 7, a dataset was added for the small and thin, such as horizontal flip (a), rotating 45 degrees (b), and rotating 315 degrees (c).
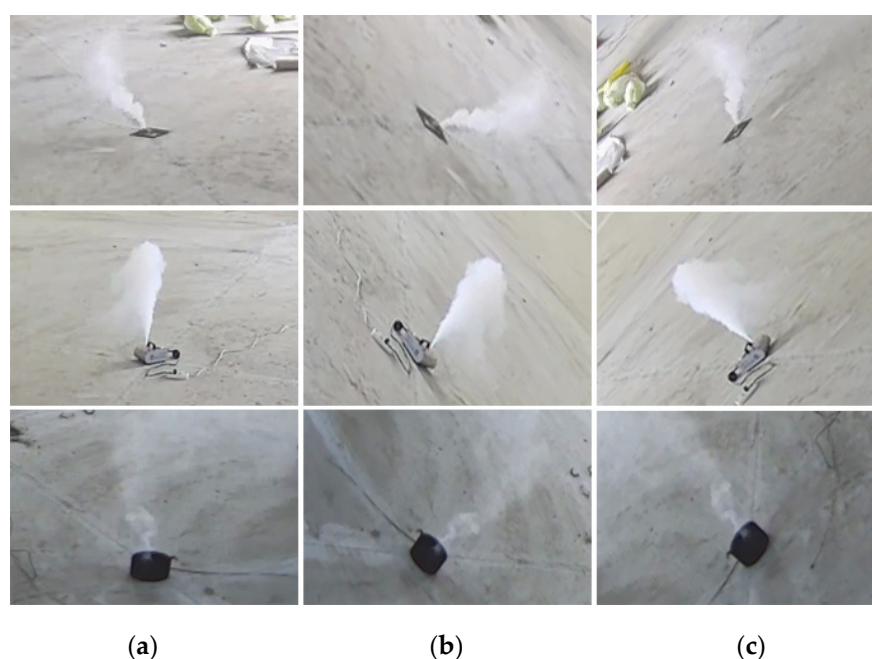
(**a**)   (**b**)   (**c**)

Figure 7. Data enhanced images. (**a**) The images were made horizontal flip; (**b**) The images were rotated 45 degrees; (**c**) The images were rotated 315 degrees.

As shown in Table 1, 10,800 pictures have been used. After data enhancement, there will be 4000 pictures, and then add 3400 pictures to the public data set, so there are 7400 pictures of smoke data. Moreover, 3400 pieces of non-smoke photos are also prepared. So the total dataset contains a total of 10,800 images. The dataset has been divided into a training set, validation set, and test set, accounting for 81%, 9%, and 10%, respectively.

Table 1. Number of dataset categories.

Type	Train	Val	Test	Total
Smoke	2475	306	340	3400
Small Smoke	3640	360	400	4000
Non-smoke	2475	306	340	3400
Total	8590	972	1080	10,800

4. Results

Open-source deep learning framework PyTorch has been used to train a smoke detection model based on the basic convolutional neural YOLO V5s, combining an attention mechanism and an improved upsampling network CARAFE3.1. To evaluate the algorithm's performance, firstly, the algorithm is tested on public smoke datasets and a smoke dataset. Secondly, the algorithm is compared with the existing excellent algorithms based on different evaluation metrics. The model's detection speed and parameters are tested to verify the algorithm's real-time detection performance. In order to verify the effect of CARAFE, a comparative experiment is applied. Ablation experiments with the attention mechanism are also conducted.

4.1. Evaluation Criteria

After the experiment, precision, recall, F1-Score, and $AP_{0.5}$ ($mAP_{0.5}$) are used to evaluate the model detection and compare it with the classic model. Precision is the ratio of the number of samples accurately predicted to be positive to the sum of the number of samples that are predicted to be true. A recall is the ratio of the number of samples accurately predicted to be positive to the sum of the positive samples. F1 score is the harmonic

mean of precision and recall. $AP_{0.5}$ is the average precision when the confidence level is 0.5, and the area enclosed by the PR curve $mAP_{0.5}$ is the average value of AP value under all categories. For example, Formulas (6)–(10), which refer to class *i*, belong to normal smoke, little smoke, and non-smoke. TP_i means that the model predicts the *i-th* sample as the *i-th* sample. FP_i means that the model predicts samples that do not belong to class *i* as class *i*. TN_i means that the model predicts samples that do not belong to class *i* as not belonging to class *i*. FN_i shows that the *i-th* sample predicted by the model does not belong to the *i-th* sample. The code of *r* is the abbreviation for recall and the code of *P* is the abbreviation for precision. The definition of *P(r)* is function with recall as abscissa and Precision as ordinate. The formulas are as follows:

$$\text{Precision} = \frac{1}{3} \times \sum_{i}^{3} \frac{TP_i}{TP_i + FP_i} \tag{6}$$

$$\text{Recall} = \frac{1}{3} \sum_{i}^{3} \frac{TP_i}{TP_i + FN_i} \tag{7}$$

$$\text{F1} - \text{Score} = \frac{2 \times P \times R}{P + R} \tag{8}$$

$$\text{AP} = \int_{0}^{1} P(r) \, dr \tag{9}$$

$$\text{mAP} = \frac{\sum_{i}^{3} \text{AP}}{3} \tag{10}$$

The algorithm was compared with some representative single-stage networks based on convolutional neural networks and excellent object detection networks, namely YOLO V4 [33], SSD [38], Efficient-d2, Retinanet [39], and YOLO V5s, to evaluate the performance of proposed smoke detection method. The most of the traditional smoke detection methods extract features subjectively, which is easily affected by the external environment. Their performance is lower than the use of depth features.

Therefore, the comparison between method and traditional methods is not fair.

The proposed model is compared with SSD, RetinaNet, Efficientdet-d2, YOLO v4, and YOLO v5s original models in the self-created dataset through four evaluation metrics, namely Precision, Recall, F1-Score, $AP_{0.5}(mAP_{0.5})$. To fully and objectively demonstrate proposed method's effectiveness on the smoke object detection task, the following four experiments are conducted: (1) Overall experimental results from the data set were compared. (2) The experimental results of detecting small smoke in the early stage of the fire were compared. (3) The detection experiments performed on the smoke-free pictures are compared. (4) The model detection speed and parameters of the models are compared. (5) The effects of CARAFE module are compared.

4.2. Training Environment and Hyperparameters

The experimental environment is based on the Ubuntu 18.04 operating system and GeForce RTX 3090 GPU. In the training parameter, the dataset is trained for 100 epochs with batch size 16. Moreover, SGD is an optimizer with a learning rate of 0.001.

4.3. Results Compared to Dataset

Table 2 shows the overall evaluation metric of different object detection models in the dataset. The evaluation indicators of different target detection models are above 88% and close to 90%. Different target detection models have nice results for large-scale smoke image detection in this case. Among them, the SSD [38] and RetinaNet [39] object detection models perform weakly in the dataset compared to the other three models. Based on the result, the original YOLO V5s model has certain advantages over other models in dataset, which is why YOLO V5s was chosen for further improvement. Moreover, a channel attention and spatial attention mechanism were used and improved upsampling CARAFE to improve YOLO V5s and get better results on our dataset.

Table 2. Results of the overall dataset comparison.

Method/Crieria	Precision (%)	Recall (%)	F1-Score (%)	mAP$_{0.5}$ (%)
SSD	90.44	86.65	88.50	90.58
Retinanet	90.31	89.24	89.31	89.57
Efficientdet-d2	90.22	88.90	89.50	91.39
YOLO v4	90.80	80.33	89.70	90.30
YOLO v5s	91.23	89.82	90.49	91.51
Proposed	92.72	91.20	91.92	92.61

4.4. Results Compared on the Small and Thin Smoke Dataset

As shown in Table 3, the evaluation indicators were used to obtain by different models to detect objects only on the small smoke dataset in the early fire stage. As shown in Table 3, yolov4 performs poorly overall on unique dataset. Although the original model of YOLO V5s also has a good detection effect on the detection of small smoke in dataset, better results were obtained on the small smoke data set in the early stage of the fire, and the four evaluation indicators all reached more than 83%. Therefore, the proposed model to detect small and thin smoke has achieved an ideal result.

Table 3. Results on the small and thin smoke dataset in the early fire stage.

Method/Crieria	Precision (%)	Recall (%)	F1-Score (%)	AP$_{0.5}$ (%)
SSD	79.56	75.60	77.50	80.82
Retinanet	80.49	78.32	78.40	79.53
Efficientdet-d2	81.78	79.31	80.52	81.13
YOLO v4	78.51	76.00	77.00	77.16
YOLO v5s	83.72	80.21	81.92	83.34
Proposed	87.84	83.71	85.75	85.93

4.5. The Result of the Detection Experiment on the Non-Smoke Dataset

Table 4 shows that different models have been used to detect non-smoke pictures in the test dataset. The table shows that different object detection models, whether the proposed model or other models, detect non-smoke pictures well.

Table 4. Results on the non-smoke dataset.

Method/Crieria	Precision (%)	Recall (%)	F1-Score (%)	AP$_{0.5}$ (%)
SSD	99.27	99.32	99.78	99.32
Retinanet	98.78	99.39	99.28	99.14
Efficientdet-d2	99.64	99.01	99.48	99.39
YOLO v4	99.79	99.79	99.48	99.33
YOLO v5s	99.49	99.35	99.39	99.68
Proposed	99.75	99.53	99.28	99.83

4.6. Detection Speed and Parameter Results

In order to evaluate whether the detection speed of the algorithm reaches real-time detection, the average detection speed of different methods on the test set was tested, and the test results are shown in Table 5. From the table, SSD [38] has the fastest detection speed on test set, reaching 75.38 detections per second. The proposed model is not the fastest among the comparison models due to network changes, and its detection speed is slightly slower than the original model. However, it also detected 69 pictures per second, which is far beyond the frame rate of everyday HD cameras. Furthermore, the parameter of original YOLO v5s is the smallest of them all. However, there is a conclusion that though the improved model's parameter is not the smallest, the parameter of the attention mechanism and CARAFE upsampling algorithm is small.

Table 5. Result of detecting speed of smoke picture and parameter of models.

Method	Detection Speed	Parameter
SSD	75.38 fps	26,151,824
Retinanet	38.97 fps	37,968,692
Efficientdet-d2	11.91 fps	8,086,869
YOLO v4	20.83 fps	64,363,101
YOLO v5s	77.52 fps	7,018,216
Proposed	69.00 fps	7,325,300

4.7. Image Example of a Model Detection Result

Figure 8 shows the effect of the proposed model and different contrasting models in smoke detection. In order to ensure fairness, the detection effects of the same image in each category in the dataset for comparison were selected. From the result, the proposed model has better detection performance.
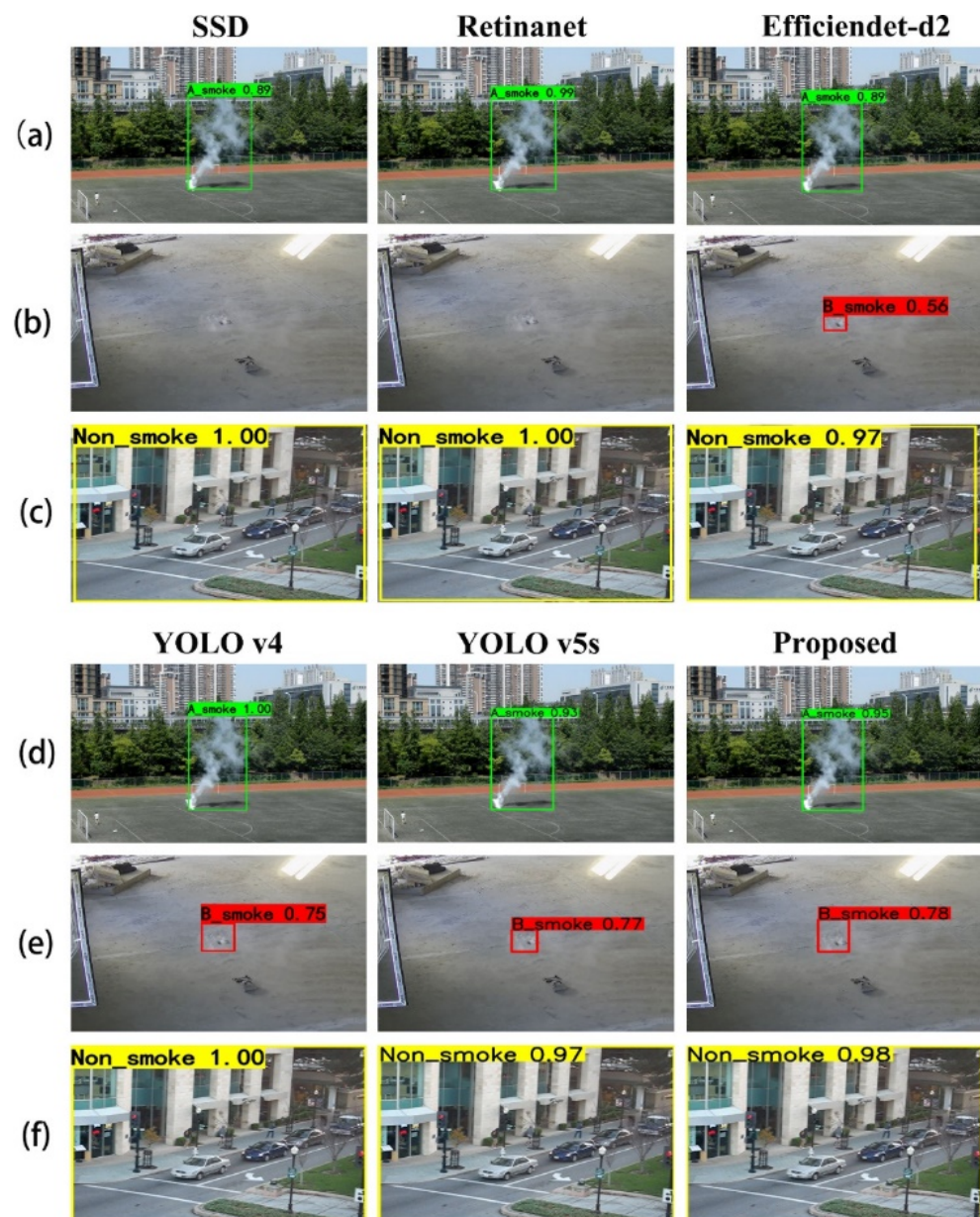
Figure 8. Detection results. (**a**,**d**) The detection result of smoke; (**b**,**e**) the detection result of small and thin smoke; (**c**,**f**) The detection result of non-smoke.

4.8. Comparative Experiment of CARAFE

In this subsection, the improved upsampling CARAFE method is only used to compare it with existing object detection models, such as SSD, Retinanet, Efficientdet-2, YOLO v4, and YOLO 5s, based on created smoke dataset.

The experimental results of the improved upsampling are shown in Table 6. CARAFE, with a detection effect of 92.52% for precision, 90.74% for recall, 91.60 for F1-Score, and 91.83% for $AP_{0.5}$, achieved the best results among all comparison models. Moreover, all the upsampling in the feature fusion network with the CARAFE module was replaced. After replacing the original upsampling in the YOLO v5s feature fusion network with the improved upsampling CARAFE, the detection effect was more than 0.5% lower than the effect of replacing only one upsampling. The improved upsampling CARAFE can increase the receptive field of the smoke feature fusion network and adapt to the content information of specific smoke in real-time. There is a conclusion that if CARAFE modules replace both upsampling, the weights of the front and rear feature fusion networks will be disordered, which is not better for smoke detection.

Table 6. Comparison of carafe experiment.

Method/Crieria	Precision (%)	Recall (%)	F1-Score (%)	$AP_{0.5}$ (%)
SSD	90.44	86.65	88.50	90.58
Retinanet	90.31	89.24	89.31	89.57
Efficientdet-d2	90.22	88.90	89.50	91.39
YOLO v4	90.80	80.33	89.70	90.30
2 × CARAFE	91.20	90.20	90.68	91.40
1 × CARAFE	92.50	90.70	91.60	91.80

4.9. Comparison of Attention Mechanism Ablation Experiments

In this subsection, the proposed method using only the attention mechanism models is compared with existing object detection, such as SSD, RetinaNet, Efficientdet-D2, YOLO v4, and YOLO 5s, based on the self-created smoke dataset. The quantitative results of the Note module are shown in Table 7. From the table, when only using the channel attention module that removes the spatial attention model, the detection effect is not optimal or lower than the index of Recall and $AP_{0.5}$ of the original model. However, when the spatial attention mechanism was added, the model's precision was 93.20%, the recall was 89.1%, F1-Score was 92.10, and $AP_{0.5}$ was 91.83%, achieving the best results among all comparison models. The proposed channel attention emphasizes the feature channel representing the smoke and gives it a higher weight. After channel attention, the color of the channel representing the smoke becomes red, while the color of the background channel becomes smoke area. The model can thus focus on the smoke in the detection task, thereby improving the detection efficiency of small smoke.

Table 7. Comparison of attention mechanism ablation experiment.

Method/Crieria	Precision (%)	Recall (%)	F1-Score (%)	$AP_{0.5}$ (%)
SSD	90.44	86.65	88.50	90.58
Retinanet	90.31	89.24	89.31	89.57
Efficientdet-d2	90.22	88.90	89.50	91.39
YOLO v4	90.80	80.33	89.70	90.30
Channel Attention	92.00	89.17	90.53	91.20
Channel + Spatial Attention	93.20	91.00	92.10	91.83

In Figure 9, heatmaps of different small smoke in self-created dataset are compared. The column in (a) represents the original image, the column in (b) represents the heatmap without attention mechanism added, and the column in (c) represents the heatmap with attention added. Considering the situation, after adding attention, the focus on small smoke will be closer to the source of the smoke. This effect is consistent with the expectation that more attention will lead to the accelerated discovery of the source of the smoke in the early stage of the fire to prevent the fire from spreading.
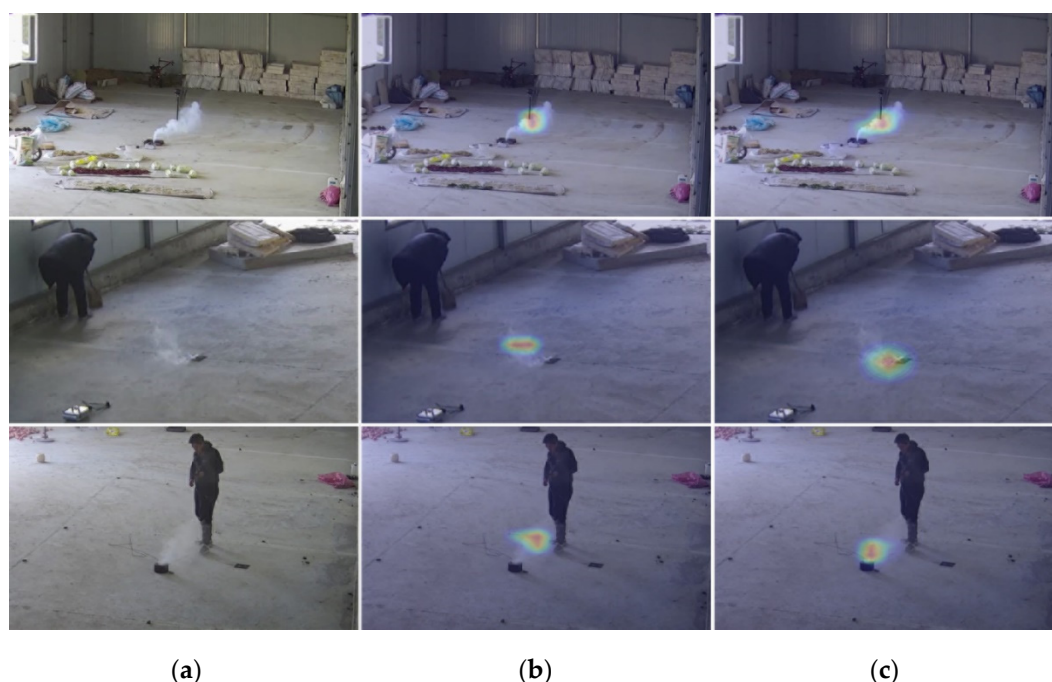
(**a**)　　　　　　　　　(**b**)　　　　　　　　　(**c**)

Figure 9. Heatmaps of the attention mechanism. (**a**) are different original images of small and thin smoke; (**b**) are the detections of (**a**) with no attention; (**c**) are the detections of (**a**) with our attention mechanism. After adding attention, the focus on small smoke will be closer to the source of the smoke.

5. Conclusions and Future Work

This paper proposes a new method with an attention mechanism and an improved upsampling algorithm to solve the small and thin smoke detection problem. Firstly, an innovative smoke dataset was created, consisting of self-created small and thin smoke images and public smoke images. Secondly, an attention mechanism module combining spatial and channel attention is used to solve the problem of small and thin smoke detection. Thirdly, a light-weighted upsampling module is used to improve further the ability to identify small smoke and ensure the model's real-time detection characteristics. Extensive experiments on the results show that the proposed method has higher precision, recall, F1-score and $mAP_{0.5}(AP_{0.5})$ than existing methods under the premise of guaranteeing real-time performance.

In the future, the proposed algorithm will be deployed on embedded systems and development boards, such as Jetson Nano, Beagle Bone, and Raspberry Pi 3B+. In addition, the algorithm will be improved to obtain detailed information about the smoke, such as the burning substances that cause it and the speed of the smoke spreading.

Author Contributions: Conception, H.Y. and M.C.; methodology, H.Y., M.C. and W.F.; software, M.C. and Y.J.; validation, H.Y., M.C., W.F.,Y.J. and S.L.; formal analysis, H.Y., S.G.H. and Y.J.; investigation, H.Y., M.C. and S.L.; resource, H.Y.; data, H.Y. and M.C.; writing—original draft preparation, H.Y., M.C., W.F. and Y.J.; writing—review and editing, H.Y., S.G.H., S.L. and M.C.; projection administration, H.Y. and S.L. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: Not applicable.

References

1.　Khan, S.; Muhammad, K.; Mumtaz, S.; Baik, S.W.; de Albuquerque, V.H.C. Energy-Efficient Deep CNN for Smoke Detection in Foggy IoT Environment. *IEEE Internet Things J.* **2019**, *6*, 9237–9245.
2.　Khan, M.; Rafik, H.; Jamil, A.; Jaime, L.; Haoxiang, W.; Wook, B.S. Secure surveillance framework for IoT systems using probabilistic image encryption. *J. IEEE Trans. Industr. Inform.* **2018**, *14*, 3679–3689.
3.　Mehdi, M.; Ala, A.; Mohsen, G.; Seok, O.J. Semisupervised Deep Reinforcement Learning in Support of IoT and Smart Cit y Service. *IEEE Internet Things J.* **2018**, *5*, 624–635.
4.　Sheng, L.; Yuzheng, J. State-of-art of video based smoke detection algo-rithms. *J. Image Graph.* **2013**, *18*, 1225–1236.
5.　Keun, K.D.; Fang, W.Y. Smoke detection in video. In Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering, Los Angeles, CA, USA, 31 March–2 April 2009; CSIE: Danville, CA, USA, 2009; pp. 759–763.
6.　Ye, S.; Bai, Z.; Chen, H.; Bohush, R.; Ablameyko, S. An effective algorithm to detect both smoke and flame using color and wavelet analysis. *Pattern Re Cogn. Image Anal.* **2017**, *27*, 131–138.
7.　Simone, C.; Paolo, P.; Rita, C. Vision based smoke detection system using image energy and Color Information. *Mach. Vis. Appl.* **2011**, *22*, 705–719.
8.　Feiniu, Y.; Xue, X.; Jinting, S.; Hongdi, L.; Gang, L. Non-Linear Dimensionality Reduction and Gaussian Process Based Classification Method for Smoke Detection. *IEEE Access* **2017**, *5*, 6833–6841.
9.　Feiniu, Y.; Jinting, S.; Xue, X.; Yuming, F.; Zhijun, F.; Tao, M. High-order local ternary patterns with locality preserving projection for smoke detection and image classification. *Inf. Sci.* **2016**, *372*, 225–240.
10.　Maruta, H.; Nakamura, A.; Kurokawa, F. A new approach for smoke detection with texture analysis and support vector machine. In Proceedings of the IEEE International Symposium on Industrial Electronics, Bari, Italy, 4–7 July 2010; IEEE: Washington, DC, USA, 2010.
11.　Jia, Y.; Yuan, J.; Wang, J.; Fang, J.; Zhang, Y.; Zhang, Q. A Saliency-Based Method for Early Smoke Detection in Video Sequences. *Fire Technol.* **2016**, *52*, 1271–1292.
12.　Ko, B.; Kwak, J.; Nam, J. Wildfire smoke detection using temporospatial features and random forest classifiers. *Opt. Eng.* **2012**, *51*, 017208.
13.　Jifeng, S.; Xin, Z.; Wankou, Y.; Hualong, Y.; Guohai, L. Learning discriminative shape statistics distribution features for pedestrian detection. *Neurocomputing* **2016**, *184*, 66–77.
14.　Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classi fication with Deep Convolutional Neural Networks. *Adv. Neural Inf. Processing Syst.* **2012**, *25*, 1097–1105.
15.　Lijun, H.; Xiaoli, G.; Sirou, Z.; Liejun, W.; Fan, L. Efficient attention based deep fusion CNN for smoke detection in fog environment. *Neurocompting* **2012**, *434*, 224–238.
16.　Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognitionm. *arXiv* **2014**, arXiv:1409.1556.
17.　Xiaoli, G.; Hangyu, H.; Zhengwei, W.; Lijun, H.; Le, Y.; Fan, L. Dark-channel based attention and classifier retraining for smoke detection in foggy environments. *Digit. Signal Processing Rev. J.* **2022**, *123*, 103454.
18.　Hang, Y.; Yurong, W. An improved algorithm based on convolutional neural network for smoke detection. In Proceedings of the 2019 IEEE International Conferences on Ubiquitous Computing and Communications and Data Science and Computational Intelligence and Smart Computing, Net-Working and Services, IUCC/DSCI/SmartCNS, Shenyang, China, 21–23 October 2019; pp. 207–211.
19.　Yingshu, P.; Yi, W. Real-time forest smoke detection using hand-designed features and deep learning. *Comput. Electron. Agric.* **2019**, *167*, 105029.
20.　Ke, G.; Zhifang, X.; Junfei, Q.; Weisi, L. Deep Dual-Channel Neural Network for Image-Based Smoke Detection. *IEEE Trans. Multimed.* **2020**, *22*, 311–323.
21.　Sergey, I.; Christian, S. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *PMLR* **2015**, *37*, 448–456.
22.　Forrest, N.I.; Song, H.; Matthew, M.; Khalid, W.A.; William, D.; Kurt, J.K. SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 M B model size. *arXiv* **2016**, arXiv:1602.07360.
23.　Chenghua, L.; Bin, Y.; Hao, D.; Hongling, S.; Xiaoping, J.; Jing, S. Real-time video-based smoke detection with high accuracy and efficiency. *Fire Saf. J.* **2020**, *117*, 103184.
24.　Jiang, M.; Zhao, Y.; Yu, F.; Zhou, C.; Peng, T. A self-attention network for smoke detection. *Fire Saf. J.* **2022**, *129*, 103547.
25.　Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Washington, DC, USA, 2020.
26.　Guo, M.H.; Xu, T.X.; Liu, J.J.; Liu, Z.N.; Jiang, P.T.; Mu, T.J.; Zhang, S.H.; Martin, R.R.; Cheng, M.M.; Hu, S.M. Attention mechanisms in computer vision: A survey. *Comput. Vis. Media* **2021**, *8*, 331–368.

27. Wang, J.; Chen, K.; Xu, R.; Liu, Z.; Loy, C.C.; Lin, D. CARAFE: Content-Aware ReAssembly of Features. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: Washington, DC, USA, 2019; pp. 3007–3016.

28. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In *2016 IEEE Conference on Computer Vision and Pattern Re cognition (CVPR)*; IEEE: Washington, DC, USA, 2016; pp. 779–788.

29. Girshick, R. Fast R-CNN. In *2015 IEEE International Conference on Computer Vision (ICCV)*; IEEE: Washington, DC, USA, 2015; pp. 1440–1448.

30. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149.

31. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Washington, DC, USA, 2017; pp. 6517–6525.

32. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, *e-prints*, arXiv:1804.02767.

33. Bochkovskiy, A.; Wang, C.Y.; Liao, H. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.

34. Wang, C.-Y.; Liao, H.-Y.M.; Wu, Y.-H.; Chen, P.-Y.; Hsieh, J.-W.; Yeh, I.-H. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recogniti on Workshops (CVPRW)*; IEEE: Washington, DC, USA, 2020; pp. 1571–1580.

35. Lin, T.-Y.; Dolla'r, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Washington, DC, USA, 2017; pp. 936–944.

36. Wang, W.; Enze, X.; Xiaoge, S.; Yuhang, Z.; Wenjia, W. Efficient and Accurate Arbitrary-Shaped Text Detection With Pixel Aggregation Network. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: Washington, DC, USA, 2019; pp. 8439–8448.

37. Noh, H.; Hong, S.; Han, B. Learning Deconvolution Network for Semantic Segmentation. In *2015 IEEE International Conference on Computer Vision (ICCV)*; IEEE: Washington, DC, USA, 2015; pp. 1520–1528.

38. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016, Lecture Notes in Computer Science*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; p. 9905.

39. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dolla'r, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327.