applied sciences

Communication

Light-YOLOv5: A Lightweight Algorithm for Improved YOLOv5 in Complex Fire Scenarios

Hao Xu, Bo Li and Fei Zhong *

School of Mechanical Engineering, Hubei University of Technology, Wuhan 430068, China
* Correspondence: hg_zfxs@sina.com

Abstract: Fire-detection technology is of great importance for successful fire-prevention measures. Image-based fire detection is one effective method. At present, object-detection algorithms are deficient in performing detection speed and accuracy tasks when they are applied in complex fire scenarios. In this study, a lightweight fire-detection algorithm, Light-YOLOv5 (You Only Look Once version five), is presented. First, a separable vision transformer (SepViT) block is used to replace several Cross Stage Partial Bottleneck with 3 convolutions (C3) modules in the final layer of a backbone network to enhance both the contact of the backbone network to global information and the extraction of flame and smoke features; second, a light bidirectional feature pyramid network (Light-BiFPN) is designed to lighten the model while improving the feature extraction and balancing speed and accuracy features during a fire-detection procedure; third, a global attention mechanism (GAM) is fused into the network to cause the model to focus more on the global dimensional features and further improve the detection accuracy of the model; and finally, the Mish activation function and SIoU loss are utilized to simultaneously increase the convergence speed and enhance the accuracy. The experimental results show that compared to the original algorithm, the mean average accuracy (mAP) of Light-YOLOv5 increases by 3.3%, the number of parameters decreases by 27.1%, and the floating point operations (FLOPs) decrease by 19.1%. The detection speed reaches 91.1 FPS, which can detect targets in complex fire scenarios in real time.

Keywords: fire detection; Light-YOLOv5; global attention mechanism; lightweight

check for updates

Citation: Xu, H.; Li, B.; Zhong, F. Light-YOLOv5: A Lightweight Algorithm for Improved YOLOv5 in Complex Fire Scenarios. *Appl. Sci.* **2022**, *12*, 12312. https://doi.org/10.3390/app122312312

Academic Editors: Byung-Gyu Kim and Yu-Dong Zhang

Received: 8 October 2022
Accepted: 29 November 2022
Published: 1 December 2022

1. Introduction

Fires can have a significant impact on public safety, and every year, they cause a high number of deaths, injuries, and property damage. The timely detection of fires can dramatically reduce casualties and losses.

Traditional fire-detection methods mainly use smoke and temperature sensors with a limited detection range, scenarios, and extended response times. With the developments in the areas of artificial intelligence and machine learning, fire detection based on deep learning is extensively used. However, fire-detection scenarios are often too complex and changeable. In this case, the generalization and robustness of traditional fire-detection algorithms are insufficient, and their deployment in low-computing-power platforms is challenging. In this paper, we propose a lightweight Light-YOLOv5s algorithm for complex fire-scene-detection scenarios based on YOLOv5 to address the shortcomings of existing fire-detection methods. The contributions of this paper are as follows:

1. Replacement of the last layers of the backbone network with SepViT Block and strengthening of the network's connection to global feature information.
2. We propose a Light-BiFPN structure to reduce the computational cost and parameters while enhancing the fusion of multi-scale features and enriching the semantic features.
3. We incorporate the global attention mechanism into YOLOv5 to enhance the overall feature-extraction capability of the network.
4. We verify the validity of the Mish activation and SIoU loss functions.

The remainder of the paper is organized as follows: Section 2 focuses on the studies related to fire-detection practices; Section 3 describes the model's framework and the implementation details; Section 4 verifies the algorithm's effectiveness through experiments; and Section 5 concludes with a summary.

2. Related Work

Due to the irregular shape of smoke, uneven spatial distribution, and short existence time, it is not easy to accurately detect smoke. Traditional methods usually detect smoke by some its features, such as color, texture, and shape. Favorskaya et al. [1] used dynamic texture features to detect smoke using two- and three-dimensional LBP histograms, which can exclude wind interference in static scenes. Dimitropoulos et al. [2] used the HSV model and adaptive median algorithm for pre-processing and a high-order linear dynamic system for the dynamic texture analysis of smoke, significantly improving the overall detection accuracy. Wang et al. [3] proposed a flame-detection method that combines the dynamic and static features of a flame in a video and reduces the influence of the environment by combining flame-color and local features.

In recent years, with the rapid development of machine learning techniques, the latest target-detection algorithms in deep learning have been applied to fire-detection practices. Wang et al. [4] used an improved YOLOv4 network for real-time smoke and fire detection, dramatically reducing the number of parameters to improve the overall detection speed and successfully deploying this to UAVs, but with a lower accuracy than the original algorithm. Zhang et al. [5] proposed a T-YOLOX fire-detection algorithm using the VIT technique to improve the accuracy of detecting smoke, fire, and people, but did not discuss the number of parameters or the computational effort involved. Zhao et al. [6] proposed an improved Fire-YOLO algorithm for forest fires to enhance the detection of small targets in fires and reduce the model size. Nonetheless, they did not discuss the number of parameters or computational effort. Li et al. [7] improved the algorithm of YOLOv3-tiny to improve the fire-detection accuracy by multi-scale fusion and k-means clustering. However, the detection speed was not ideal, and the application scenario was singular. Yue et al. [8] reduced the false-detection rate by increasing the resolution of the feature map and expanding the perceptual field, but the detection speed was not satisfactory. However, some of these articles mentioned above are unsatisfactory in terms of speed and accuracy, and some are too homogeneous in terms of the environment to balance these three issues. Therefore, this paper proposes a Light-YOLOv5s method for the purpose of fire detection to achieve a balance of speed and accuracy in complex environments.

3. Methods

3.1. Baseline

YOLOv5 has n, s, and m versions. We selected YOLOv5n, which has both speed and accuracy functions, as the baseline for improvement following an experimental comparison, and we labeled the improved model Light-YOLOv5, whose structure is presented in Figure 1.

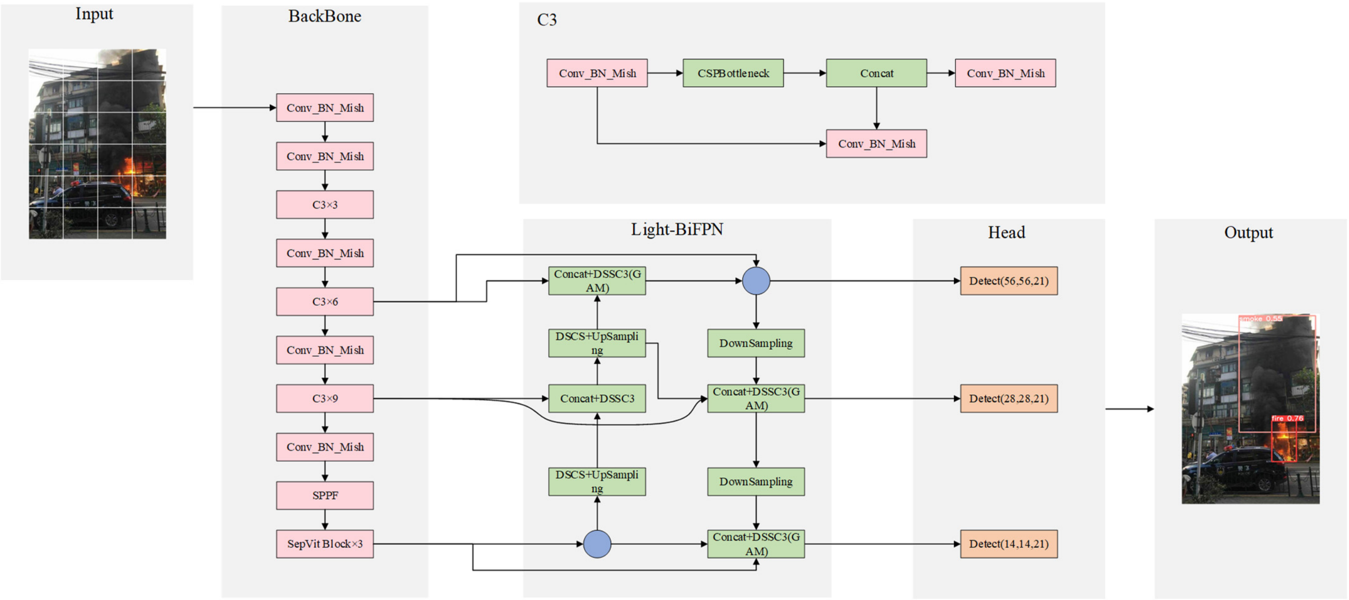

Figure 1. The architecture of the Light-YOLOv5 model.

3.2. Separable Vision Transformer

In recent years, vision transformer [9,10] has achieved great success in various computer vision tasks, boasting a performance that exceeds that of CNNs in essential domains. However, these performances usually come at the cost of an increased computational complexity and the number of parameters.

A separable vision transformer [11] solves this challenge by maintaining its accuracy while balancing the computational costs. In this paper, we replaced the final layer of the backbone network with the SepViT (Separable Vision Transformer) Block, which enhances the feature-extraction capability of the model and optimizes the relationship of the global information of the network. In SepViT Block, the depthwise and pointwise self-attention values reduce the computation and enable local information communication and global information interaction in windows. First, the input feature map was divided into windows, and each window was considered to be an input channel of the feature map, and each window contained different types of information. Then, depthwise self-attention (DWA) was performed on each window token and its pixel tokens. The DWA focused on fusing the spatial information of the channels, similar to a depthwise convolution in MobileNet [12–14]. The operation of DWA is as follows:

$$\text{DWA}(f) = \text{Attention}(f \cdot W_Q, f \cdot W_K, f \cdot W_V) \tag{1}$$

where f is the feature tokens, composed of window tokens and pixel tokens. W_Q, W_K, and W_V represent three linear layers for query, key, and value computations in a routine self-attention task. Attention means a standard self-attention operation. Pointwise self-attention (PWA) is similar to the pointwise convolution operation in MobileNet, except that a pointwise convolution is used to fuse information from different channels while PWA establishes connections between windows. After completing the DWA operation, PWA builds relationships among windows and generates the attention map by LayerNormalization (LN) and a Gelu activation function. The operation of PWA is as follows:

$$\text{PWA}(f, wt) = \text{Attention}(\text{Gelu}(\text{LN}(\ wt)) \cdot\ W_Q,\ \text{Gelu}(\text{LN}(wt)) \cdot\ W_K, f) \tag{2}$$

where wt represents the window token. Then, SepViT Block can be expressed as

$$\overset{\sim n}{f} \ = \text{Concat}(f^{n-1}, wt) \tag{3}$$

$$\ddot{f}^{n} = \text{DWA}(\text{LN}(\tilde{f}^{n})) \tag{4}$$

$$\dot{f}^{n}, \dot{wt} = \text{Slice}(\ddot{f}^{n}) \tag{5}$$

$$\hat{f}^{n} = \text{PWA}(\dot{f}^{n}, \dot{wt}) + f^{n-1} \tag{6}$$

$$f^{n} = \text{MLP}(\text{LN}(\hat{f}^{n})) + \hat{f}^{n} \tag{7}$$

where f^n represents the SepViT Block. $\dot{f}^{n}$ and $\dot{wt}$ are the feature maps and learned window tokens. Concat denotes the concatenation operation. Slice indicates the slice operation. Figure 2 presents the structure of the SepViT Block.

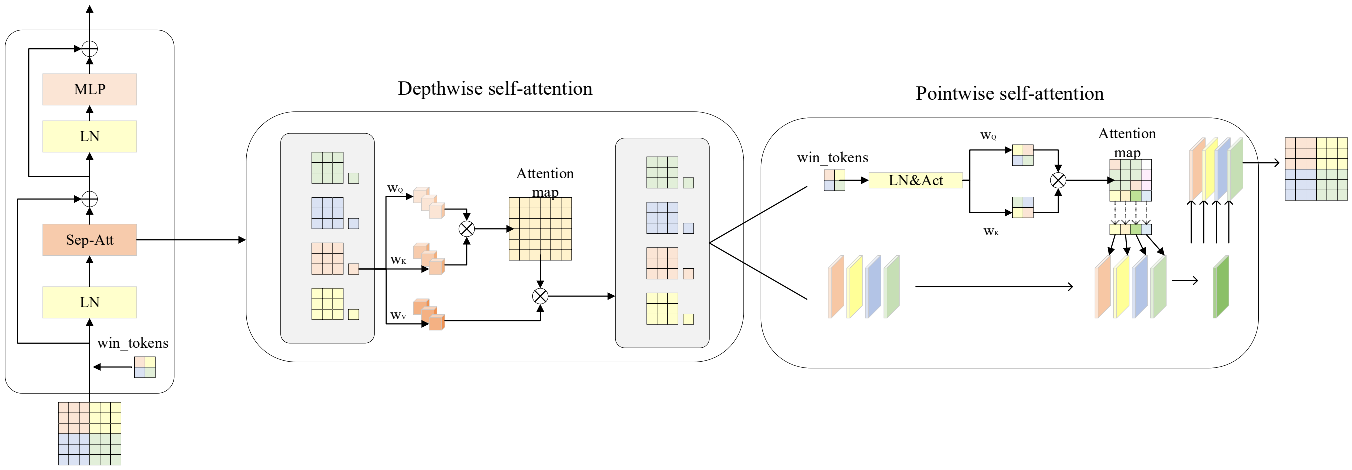

Figure 2. The overall structure of the SepViT Block. MLP represents Multi-Layer Perceptron. Sep-Att represents Separable attention.

3.3. Light-BiFPN Neck

This section was inspired by the following articles: MobileNet, ShuffleNet [15,16], EfficientDet [17], GhosetNet [18], and PP-LCNet [19]. We designed a lightweight neck network that we termed Light-BiFPN (Light Bidirectional Feature Pyramid Network).

In the practice of fire detection, where speed and accuracy are equally important factors, we observed that there was a minor difference in the accuracy between depth-wise separable convolution (DSC) and ghost convolution. DSC can reduce the number of parameters and calculations to a greater extent. However, DSC also has the disadvantage that the channel information of the input image is separated during the computing process. To solve this problem, we improved the DSC block in [19] by channel shuffling the DSC output features. We termed the improved the DSSConv module, whose structure is presented in Figure 3b, where the depth-separable convolution consists of depth and point convolutions and the input of a $H \times W \times C$ feature map P; the depth-wise convolution with one filter per input channel can be described as

$$G_{k,l,m} = \sum_{i,j} K_{i,j,m} \cdot P_{k+i-1,l+j-1,m} \tag{8}$$

where K is the depth-wise convolutional kernel with size $H_k \times W_k \times C$, where the m_{th} filter K is applied to the m_{th} channel P to produce the m_{th} channel of the filtered output feature map G. Then, the new features are generated by a 1×1 point convolution. The calculation process is presented in Figure 3a.

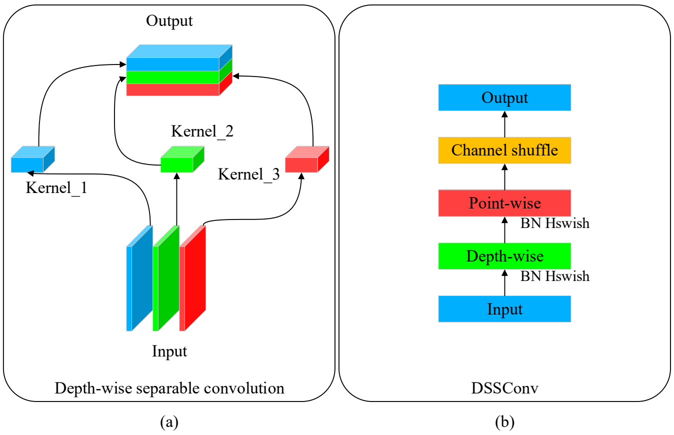

Figure 3. (**a**) The calculation process of the DSC. (**b**) The structure of the DSSConv module.

We designed DSSbottleneck and DSSC3 based on the bottleneck and C3 modules of YOLOv5, whose structures are presented in Figure 4a,b.

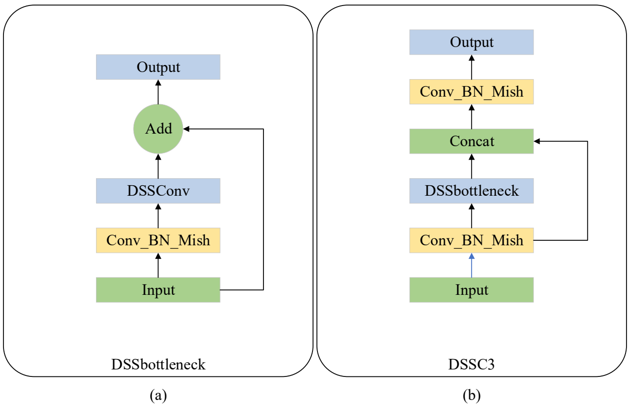

Figure 4. (**a**) The structure of the DSSbottleneck module. (**b**) The structure of the DSSC3 module.

YOLOv5 uses PANET [20] on the neck to perform feature extraction and fusion. It uses bottom-up and top-down bidirectional fusion methods and achieves good results. However, the environment of fire detection is usually too complex, and more features need to be fused to achieve better results. BiFPN is a weighted bidirectional feature pyramid network that connects input and output nodes of the same layer across other layers to achieve higher-level fusion and shorten the information transfer path between higher and lower layers. Since weighting causes an inevitable computational increase, we removed the weighted feature fusion to create a more lightweight neck network.

3.4. Global Attention Mechanism

The complex environment of fire detection is prone to false and missed detections. GAM (Global Attention Mechanism) [21] strengthens the connection between space and channels, reduces the loss of information concerning flames and smoke in a fire, and amplifies the features of the global dimension. Given an input feature map $F_1 \in R^{H\times W\times C}$, the output F_3 is defined as

$$F_3 = M_S(M_c(F_1)\otimes F_1)\otimes(M_c(F_1)\otimes F_1) \tag{9}$$

where M_c is the channel map and M_S is the spatial map; $\otimes$ represents the element-wise multiplication. We added GAM to the bottleneck module, whose structure is presented in Figure 5.

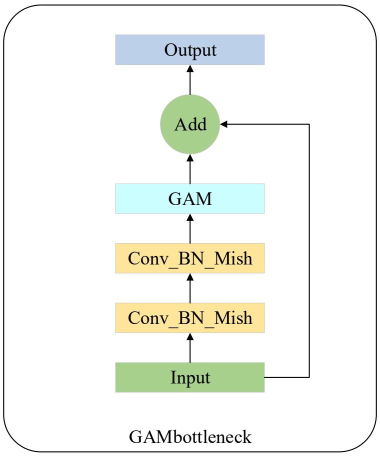

Figure 5. The structure of the GAM bottleneck.

3.5. IoU Loss and Activation

The loss function of Light-YOLOv5 consists of two main components: classification loss and bounding box regression loss. The most classical form of bounding box regression loss is IoU [22] loss, and the most commonly used version in the YOLO series is CIoU [23]. As the research progresses, there are increasingly more variants of IoU, such as DIoU [24], GIoU [25], EIoU [26], and the latest, SIoU [27]. They are defined as follows:

$$Loss_{IoU} = 1 - IoU, IoU = \left|\frac{A \cap B}{A \cup B}\right| \tag{10}$$

$$Loss_{GIoU} = 1 - IoU + \frac{C - (A \cup B)}{C} \tag{11}$$

$$Loss_{DIoU} = 1 - IoU + \frac{\rho^2_{(b,b^{gt})}}{d^2} \tag{12}$$

$$\begin{aligned} &Loss_{CIoU} = 1 - IoU + \tfrac{\rho^2_{(b,b^{gt})}}{d^2} + \alpha v, \\ &\alpha = \tfrac{v}{(1-IoU)+v}, \\ &v = \tfrac{4}{\pi^2}(arctan\tfrac{w^{gt}}{h^{gt}} - arctan\tfrac{w}{h})^2 \end{aligned} \tag{13}$$

$$Loss_{EIoU} = 1 - IoU + \frac{\rho^2_{(b,b^{gt})}}{d^2} + \frac{\rho^2_{(w,w^{gt})}}{C_w^2} + \frac{\rho^2_{(h,h^{gt})}}{C_h^2} \tag{14}$$

$$\begin{aligned} &Loss_{SIou} = 1 - IoU + \tfrac{\Delta+\Omega}{2}, \\ &\Delta = \sum_{t=x,y} (1 - \mathrm{e}^{-\gamma\rho_t}), \\ &\rho_x = \tfrac{b^{gt}_{c_x} - b_{c_x}}{c_w}, \rho_y = \tfrac{b^{gt}_{c_y} - b_{c_y}}{c_h}, \gamma = 2 - \Lambda, \\ &\Lambda = 1 - 2 * \sin^2(\arcsin(x) - \tfrac{\pi}{4}), \\ &x = \tfrac{c_h}{\sigma} = \sin(\alpha), \\ &\sigma = \sqrt{\left(b^{gt}_{c_x} - b_{c_x}\right)^2 + \left(b^{gt}_{c_y} - b_{c_y}\right)^2}, \\ &c_h = \max(b^{gt}_{c_y}, b_{c_y}) - \min(b^{gt}_{c_y}, b_{c_y}), \\ &\Omega = \sum_{t=w,h} (1 - \mathrm{e}^{-\omega_t})^{\theta}, \\ &\omega_w = \tfrac{\left|w - w^{gt}\right|}{\max(w,w^{gt})}, \omega_h = \tfrac{\left|h - h^{gt}\right|}{\max(h,h^{gt})} \end{aligned} \tag{15}$$

where the parameters A and B represent the areas of the ground-truth and prediction bounding boxes, which are rectangular, to describe the spatial location of objects; C denotes the minimum enclosing box of the ground-truth and prediction bounding boxes; b, b^{gt}

represent the centroids of the prediction and ground-truth bounding boxes, respectively; ρ represents the Euclidean distance between the two centroids; d is the diagonal distance of the smallest enclosing region that can contain both the prediction and ground-truth bounding boxes; α is the weight function; and v is used to measure the similarity of the aspect ratios.

The CIoU used by YOLOv5 relies on the aggregation of bounding box regression metrics and does not consider the direction of the mismatch between the desired ground box and predicted "experimental" box. This leads to its inferiority to SIoU in terms of training speed and prediction accuracy.

In lightweight networks, HSwish, Mish, and LeakyReLu are quicker than ReLu regarding training speed. They can be defined as

$$Hswish(x) = x \cdot \frac{\mathrm{ReLu6}(x+3)}{6} \tag{16}$$

$$Mish(x) = x \cdot \tanh(\log(1+\mathrm{e}^{x})) \tag{17}$$

$$\mathrm{LeakyReLu}(x) = \max(ax, x) \tag{18}$$

We observed in our experiments that the Mish activation function was more accurate than the others, and detailed comparison experiments are presented in Section 4.

4. Experiment

4.1. Datasets

Since there is a lack of authoritative datasets for the purpose of fire detection, the dataset used in this paper was derived from public datasets [28,29] and Web images and contains 21,136 images. The dataset we collected contained various scenarios, such as forest, indoor, urban, and traffic fires. Figure 6 presents a part of the dataset.

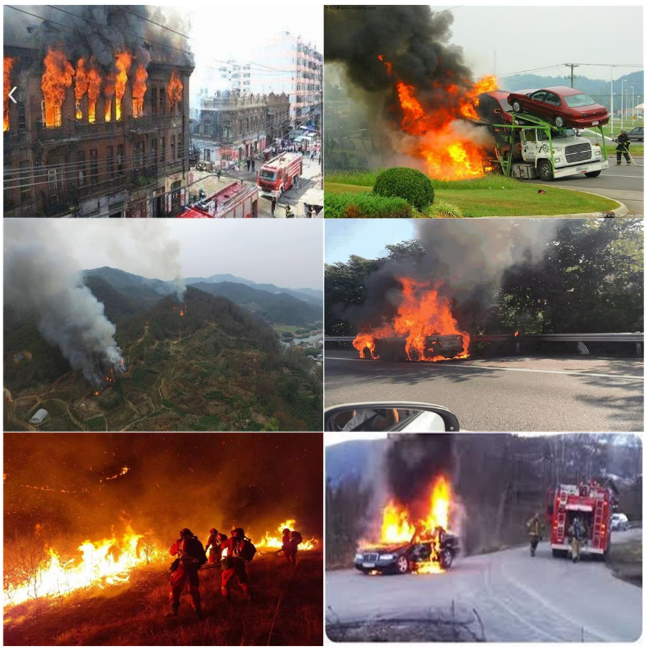

Figure 6. Example images of the dataset.

4.2. Training Environment and Details

We used the Ubuntu 18.04 operating system, NVIDIA GeForce RTX3060 GPU, CUDA11.1, Python3.8.8, and PyTorch1.8.0. We randomly divided the dataset into training, validation, and test data by 8:1:1, and used an SGD optimizer for training purposes; the batch size was set to 16, the initial learning rate was 0.01, with 100 training epochs, and the size of the input image was 448 × 448.

4.3. Model Evaluation

In this paper, precision (P), recall (R), average precision (AP), mean average precision (mAP), F1 score, parameters, FLOPs, and frames per second (FPS) were used as the evaluation metrics for the model's performance, where AP is the area under the PR curve and mAP denotes the average AP for each category. The specific formulas are as follows:

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}. \tag{19}$$

$$\mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{20}$$

$$\mathrm{mAP} = \frac{1}{n}\sum_{i-1}^{n}\int_{0}^{1} P(R)dR \tag{21}$$

$$\mathrm{F1} = 2 \times \frac{P \times R}{P + R} \tag{22}$$

where TP (true positive) indicates that the sample is correctly divided into positive samples; FP (false positive) means that the sample is incorrectly divided into positive samples; FN (false negative) means that the sample is incorrectly divided into negative samples; and *n* denotes the number of categories.

FLOPs refer to the number of computations, which measures the complexity of the model. FPS refers to the number of frames per second transmitted.

4.4. Result Analysis and Ablation Experiments

To verify the model's validity further, we performed a series of ablation experiments in this section. As shown in Table 1, we compared the effects of different versions of YOLOv5. YOLOv5n has good accuracy with fewer parameters and computations than YOLOv5s and YOLOv5m; therefore, we selected it as the baseline.

Table 1. Performance comparison of different YOLOv5models.

Model	Params (M)	FLOPs (G)	mAP@0.5 (%)	FPS
YOLOv5n	1.77	4.2	67.6	111.1
YOLOv5s	7.02	15.9	69.3	100.0
YOLOv5m	20.87	48.0	70.4	87.78

Furthermore, we conducted comparison experiments on LeakyReLu, Mish, and HSwish activation functions and CIoU, as well as SIoU with YOLOv5n as the baseline. As shown in Table 2, LeakyReLu and HSwish were faster than Mish but worse in accuracy, and SIoU was better than CIoU; therefore, we selected the combination of the Mish activation function and SIoU loss.

Table 2. The comparison results of different activation functions and IoU loss under the same model.

Model	Activation/IoU Loss	Params (M)	mAP@0.5 (%)	FPS
YOLOv5n	LeakyReLu/CIoU	1.77	67.8	107.3
	HSwish/CIoU	1.77	67.3	109.2
	Mish/CIoU	1.77	68.0	95.3
	LeakyReLu/SIoU	1.77	68.3	107.5
	HSwish/SIoU	1.77	67.6	109.4
	Mish/SIoU	1.77	68.7	95.6

To validate the effectiveness of Light-BiFPN further, we used the latest method of replacing the backbone with a lightweight network to conduct comparative experiments. We observed that ShuffleNetv2 had the fastest detection speed, but this was hardly satisfying in terms of its accuracy. In contrast, Light-BiFPN had the second-fastest detection speed after ShuffleNetv2 and a much higher accuracy than the other lightweight networks. The results are presented in Table 3.

Table 3. The comparative experiments of Light-BiFPN and different state-of-the-art lightweight models.

Model	Params (M)	FLOPs (G)	mAP@0.5 (%)	FPS
MobileNetv3-YOLOv5n	1.93	3.5	62.8	98.2
ShuffleNetv2-YOLOv5n	0.71	1.0	61.8	126.3
GhostNet-YOLOv5n	1.39	3.3	64.8	116.4
PPLCNet-YOLOv5n	0.95	2.0	63.5	112.5
Light-BiFPN-YOLOv5n	1.25	3.3	68.6	125.6

Finally, we compared all of the improved methods for the ablation experiments, and the results are presented in Table 4. Compared with the original algorithm, Light-YOLOv5 presented a 3.3% improvement in mAP, a 2% increase in the F1 score, a 27.1% reduction in the parameters, and a 19.1% reduction in FLOPs. Although the detection speed was slower than that of the original algorithm, it also satisfied real-time detection needs.

Table 4. Results of ablation experiments with different modified methods.

Model	Params (M)	FLOPs (G)	F1 (%)	mAP@0.5 (%)	FPS
Baseline (YOLOv5n)	1.77	4.2	66.0	67.6	111.1
Baseline + Light-BiFPN	1.25	3.3	66.5	68.6	128.6
Baseline + Light-BiFPN + SepViT	1.26	3.3	67.0	69.8	120.4
Baseline + Light-BiFPN + SepViT + GAM	1.29	3.4	67.0	70.3	106.5
Baseline + Light-BiFPN + SepViT, Mish, SIoU	1.29	3.4	68.0	70.9	91.1

We also compared these results with the most advanced detectors, at this stage, to further verify the effectiveness of the methods, and the comparison results are presented in Table 5. It can be observed that although Light-YOLOv5 was inferior to YOLOv7-tiny and YOLOv3-tiny in terms of its detection speed, it was much better than these detectors in other parameters, where the mAP and F1 values were 6.8% and 5% higher, respectively, compared with the latest YOLOV7-tiny model, further proving the effectiveness of the method in this paper. The detection effect graph is presented in Figure 7b.

Table 5. Comparison of the results of the most advanced detectors at this stage.

Model	Params (M)	FLOPs (G)	F1 (%)	mAP@0.5 (%)	FPS
YOLOv3-tiny	8.67	12.9	63.0	64.8	201.5
YOLOX-s	8.93	26.8	64.0	65.4	64.6
YOLOv7-tiny	6.01	13.1	63.0	64.1	285.3
Light-YOLOv5	1.29	3.4	68.0	70.9	91.1

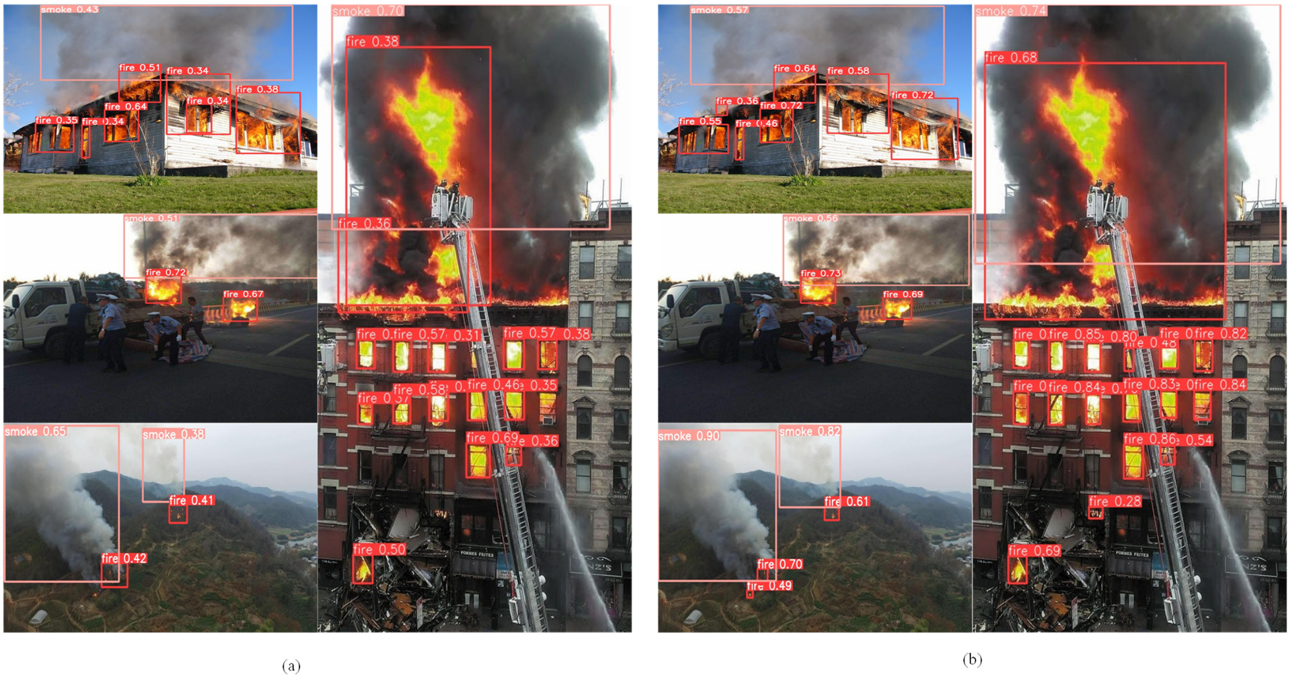

Figure 7. Comparison of Light-YOLOv5 and the original algorithm detection results. (**a**) YOLOv5n and (**b**) Light-YOLOv5.

5. Discussion

The performance of Light-YOLOv5 when applied to complex scenarios was discussed in Section 4.4. As illustrated in Table 5, by comparing the more popular lightweight algorithms at this stage, it can be observed that the mAP of Light-YOLOv5 was 6.1% higher than that of YOLOv3-tiny, 5.5% higher than that of YOLOX-s, and 6.8% higher than that of YOLOv7-tiny; in terms of the F1 scores, Light-YOLOv5 was 5% higher than YOLOv3-tiny and 4% higher than YOLOv7-tiny. Light-YOLOv5's parameters were only 1.29M, and FLOPs were only 3.4G, which means that the algorithm is easier to deploy in low-cost devices. As shown in Figure 7a,b, Light-YOLOv5 had a higher confidence level compared to the original algorithm in all cases, and it was better at detecting small targets, as presented in Figure 7a, where some of the undetected small-target flames were detected in Figure 7b.

However, we observed that the algorithm still presented limitations during the tests, such as a low accuracy in detecting small targets and semi-obscured flames or smoke, which can lead to the failure to detect the device in the first instance when it is more than a certain distance away from the target. These issues may be caused by the fire dataset's age and the lack of clarity of some images; on the other hand, it may be that the model loses too much information in the downsampling process. However, these problems are not insurmountable. For the semi-obscuration problem, we can add more smoke and fire-specific data enhancement during the data pre-processing; for the detection of small targets, we can add a detection head for small targets at the network input, or redesign an anchor for small targets. For example, in Figure 7b, it can be observed that even Light-YOLOv5 had some small targets that were not detected.

6. Conclusions

In order to improve the accuracy and speed of image-based fire-detection algorithms currently used in complex scenarios, a Light-YOLOv5 algorithm was proposed in this study. The new algorithm was constructed based on YOLOv5n. In addition, the final layer of the backbone network was replaced with SepViT Block to strengthen the connection between the backbone network and the global information; a self-designed Light-BiFPN to strengthen the network feature extraction and lighten the network; the fusion GAM module to reduce the loss of information; and finally the Mish activation function to improve the accuracy and SIoU to enhance the convergence speed in training.

The experimental results showed that with the same dataset, Light-YOLOv5 had a 3.3% higher mAP than the baseline model, a 2% higher F1 score, a 27.1% reduction in the number of parameters, and a 19.1% reduction in the computation; even when comparing the most advanced detectors, Light-YOLOv5 had a 6.8% higher mAP than the latest YOLOv7-tiny model, and the detection speed reached 91.1 FPS. Overall, the contribution of this paper is that Light-YOLOv5 dramatically reduces the number of parameters and computations and improves the detection accuracy, which is conducive to real-time fire-detection on mobile devices.

However, there is still room for improvement concerning the accuracy of our algorithm. When the distance to the target is too far, the detection may not be as effective as expected. In future work, we will further investigate how to improve our algorithm's accuracy in fire- and small-target-detection practices.

Author Contributions: Conceptualization, F.Z., B.L. and H.X.; Methodology, H.X.; Software, H.X.; Writing—original draft, H.X.; Writing—review & editing, B.L. and F.Z.; Funding acquisition, B.L. and F.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was sponsored by the National Natural Science Foundation of China (Grant No.: 52005168), the Initial Scientific Research Foundation of Hubei University of Technology (BSQD2020005), and the Open Foundation of Hubei Key Lab of Modern Manufacture Quality Engineering (KFJJ-2020009).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Favorskaya, M.; Pyataeva, A.; Popov, A. Verification of smoke detection in video sequences based on spatio-temporal local binary patterns. *Procedia Comput. Sci.* **2015**, *60*, 671–680. [CrossRef]
2. Dimitropoulos, K.; Barmpoutis, P.; Grammalidis, N. Higher order linear dynamical systems for smoke detection in video surveillance applications. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *27*, 1143–1154. [CrossRef]
3. Wang, X.; Li, Y.; Li, Z. Research on flame detection algorithm based on multi-feature fusion. In Proceedings of the2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020; Volume 1, pp. 184–189.
4. Wang, Y.; Hua, C.; Ding, W.; Wu, R. Real-time detection of flame and smoke using an improved YOLOv4 network. *Signal Image Video Processing* **2022**, *16*, 1109–1116. [CrossRef]
5. Zhang, J.; Ke, S. Improved YOLOX Fire Scenario Detection Method. In Proceedings of the Wireless Communications and Mobile Computing, Dubrovnik, Croatia, 30 May–3 June 2022.
6. Zhao, L.; Zhi, L.; Zhao, C.; Zheng, W. Fire-YOLO: A Small Target Object Detection Method for Fire Inspection. *Sustainability* **2022**, *14*, 4930. [CrossRef]
7. Li, J.; Guo, S.; Kong, L.; Tan, S.; Yuan, Y. An improved YOLOv3-tiny method for fire detection in the construction industry. In Proceedings of the E3S Web of Conferences, Changsha, China, 23–25 April 2021; Volume 253, p. 03069.
8. 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]
9. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the E3S Web of Conferences Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
10. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929, 2020.
11. Li, W.; Wang, X.; Xia, X.; Wu, J.; Xiao, X.; Zheng, M.; Wen, S. Sepvit: Separable vision transformer. *arXiv* **2022**, arXiv:2203.15380.
12. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.

13. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4510–4520.
14. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
15. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6848–6856.
16. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
17. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
18. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1580–1589.
19. Cui, C.; Gao, T.; Wei, S.; Du, Y.; Guo, R.; Dong, S.; Lu, B.; Zhou, Y.; Lv, X.; Liu, Q.; et al. PP-LCNet: A Lightweight CPU Convolutional Neural Network. *arXiv* **2021**, arXiv:2109.15099.
20. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
21. Liu, Y.; Shao, Z.; Hoffmann, N. Global Attention Mechanism: Retain Information to Enhance Channel-Spatial Interactions. *arXiv* **2021**, arXiv:2112.05561.
22. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 20–24 October 2016; pp. 516–520.
23. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **2021**, *52*, 8574–8586. [CrossRef]
24. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
25. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 658–666.
26. Zhang, Y.F.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and Efficient IOU Loss for Accurate Bounding Box Regression. *arXiv* **2021**, arXiv:2101.08158. [CrossRef]
27. Gevorgyan, Z. SIoU Loss: More Powerful Learning for Bounding Box Regression. *arXiv* **2022**, arXiv:2205.12740.
28. Park, J.; Ko, B.; Nam, J.Y.; Kwak, S. Wildfire smoke detection using spatiotemporal bag-of-features of smoke. In Proceedings of the 2013 IEEE Workshop on Applications of Computer Vision (WACV), Clearwater Beach, FL, USA, 15–17 January 2013; pp. 200–205.
29. Hüttner, V.; Steffens, C.R.; da Costa Botelho, S.S. First response fire combat: Deep leaning based visible fire detection. In Proceedings of the 2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR), Curitiba, Brazil, 8–10 November 2017; pp. 1–6.