Open Journal of Applied Sciences, 2022, 12, 2004-2018
https://www.scirp.org/journal/ojapps
ISSN Online: 2165-3925
ISSN Print: 2165-3917

Improved High Speed Flame Detection Method Based on YOLOv7

Hongwen Du, Wenzhong Zhu*, Ke Peng, Weifu Li

College of Computer Science and Engineering, Sichuan University of Science and Engineering, Yibin, China
Email: *zwz@suse.edu.cn

How to cite this paper: Du, H.W., Zhu, W.Z., Peng, K. and Li, W.F. (2022) Improved High Speed Flame Detection Method Based on YOLOv7. *Open Journal of Applied Sciences*, **12**, 2004-2018.
https://doi.org/10.4236/ojapps.2022.1212140

Received: November 25, 2022
Accepted: December 11, 2022
Published: December 14, 2022

Abstract

In order to solve the problems of the traditional flame detection method, such as low detection accuracy, slow detection speed and lack of real-time detection ability. An improved high speed flame detection method based on YOLOv7 is proposed. Based on YOLOv7 and combined with ConvNeXtBlock, CN-B network module was constructed, and YOLOv7-CN-B flame detection method was proposed. Compared with the YOLOv7 method, this flame detection method is lighter and has stronger flame feature extraction ability. 2059 open flame data sets labeled with single flame categories were used to avoid the enhancement effect brought by high-quality data sets, so that the comparative experimental effect completely depended on the performance of the flame detection method itself. The results show that the accuracy of YOLOv7-CN-B method is improved by 5% and mAP is improved by 2.1% compared with YOLOv7 method. The detection speed reached 149.25 FPS, and the single detection speed reached 11.9 ms. The experimental results show that the YOLOv7-CN-B method has better performance than the mainstream algorithm.

Keywords

Light Weight, Detection of Flame, YOLOv7-CN-B, YOLOv7, ConvNeXt

1. Introduction

Fires have become more frequent in recent years. Fire prevention and detection is an important research project which is beneficial to national economy, people's life safety and natural environment. Traditional flame monitoring mainly uses smoke sensors and temperature sensors [1]. Traditional flame monitoring is limited to a fixed and closed small space and relies on monitoring the smoke concentration and temperature threshold in the closed space to detect.

The ability to detect single flame is limited. At the same time, due to the limitation of space, the conditions of outdoor and spatial-temporal flame detection cannot be met.

In view of this, Lasaponara [2] *et al.* proposed an improved adaptive flame detection algorithm based on AVHRR (Advanced Very High Resolution Radiometer); Celik [3] *et al.* proposed a real-time flame detection algorithm that combines target foreground information with color pixel statistics; Zhou [4] *et al.* proposed a flame detection based on flame contour determine whether a target is a flame target based on three features: contour area, edge, and roundness of the detected target. Because the flame target features are affected by color, contour changes and complex scenes, the traditional flame target detection is prone to false detection and the problem of missing detection for small-sized targets.

Compared with the traditional flame detection methods, the exposed detection conditions are limited, the detection method is single, and the detection performance is worse. In the past decade, deep learning-based flame detection methods have developed rapidly. In the literature [5], a small-scale flame detection method based on YOLOv3 algorithm was proposed to achieve the detection of different scales of flames using an improved K-means clustering algorithm. In the literature [6], a Fire-YOLO algorithm is introduced, which adds depth-separable convolution to YOLOv4, reduces the computational and parametric quantities of the model, and improves the perceptual field of the feature layer by using cavity convolution, and achieves a detection speed of 42 frames/sec. A new adaptive selection algorithm for flame image features is introduced in the literature [7], which introduces genetic optimization to the attribute approximation of rough sets and increases the diversity of the population by dynamically pruning and supplementing new individuals, effectively improving the generalization ability of the flame recognition algorithm. The current advanced target detection methods are mainly single-stage and two-stage algorithms. For example, single-stage detection algorithms: RetinaNet [8], EfficientDet [9], YOLO [10], etc. Two-stage detection algorithms: Fast R-CNN [11], Faster R-CNN [12], MASK-RCNN [13], etc. The single-stage detection algorithm, compared to the two-stage detection algorithm, has the property of fast detection speed, which can better meet the real-time flame image detection. Based on this, a single-stage detection algorithm is preferred.

The YOLOV7-CN-B detection method is proposed on the basis of the most advanced target detection method YOLOv7. By combining ConvNext Block to build the CN-B network module, replace the first and last ELAN module of Backbone in YOLOv7, Replace the Bags module (Trainablebag-of-freebies in YOLOv7), the ELAN variant of Head, with the CN-B network module. The YOLOv7 network model is not only lightweight, but also enables the YOLOv7-CN-B network to obtain larger sensitivity field, enhance the ability of flame feature extraction, improve the network performance, obtain higher accuracy and mAP, achieve smaller Parameters, Gradients, Layers and less computation.

In order to verify the superiority of the improved method, an open flame data set labeled with a single flame class was used for verification experiments. The dataset consists of 2059 flame images. Using this data set eliminates the enhancement effect brought by high quality data set, making the comparison experiment effect completely dependent on the performance of the flame detection method itself. The experimental results show that the YOLOV7-CN-B method has higher accuracy and faster detection speed than the original YOLOv7 method. It solves the problem that the traditional flame detection sensor is limited to the fixed and closed small space, and the detection ability of single flame is limited. Meet the outdoor and spatiotemporal flame detection conditions. This paper provides a new detection network model for flame detection in the world.

2. Basis of Theory

2.1. YOLOv7 Model

In July 2022, YOLOv7 was born. YOLOv7 is the latest work of YOLO series. Based on previous work, this network further improves the detection speed and accuracy [14]. The YOLOv7 method has much better performance and has achieved great success in the 5 FPS to 160 FPS range, exceeding the speed and accuracy of currently known target detectors.

In the subject of real-time target detection, two mainstream efficient frame optimization design schemes are developed for CPU (central processing unit) and GPU (graphics processor), respectively. Chien-Yao Wang [15] *et al.* proposed to eliminate frame optimization while focusing on training process optimization, from 1) a more robust loss function; 2) a more efficient label assignment method; and 3) a more efficient training method. The proposed “trainable bag-of-freebies” increases the training cost but does not increase the inference cost while improving the accuracy of detection. As shown in **Figure 1**: the proposed “ELAN” and “E-ELAN” methods for real-time target detectors are based on how re-parameterized module replaces original module, and how dynamic label assignment strategy deals with assignment to different output layered two new problems that can effectively utilize parameters and computations. The proposed method can effectively reduce the amount of parameters and real-time target detector computation, with faster inference speed and higher detection accuracy.

YOLOv7 uses Model scaling for concatenation-based models. When a cascade-based model performs depth scaling, the output width of the computational block also increases. This phenomenon will lead to an increase in the input width of the subsequent transport layers. Therefore, it is proposed that when performing model scaling for cascade-based models, only the depth in the computational block needs to be scaled and the remaining part of the transport layer is performed using the corresponding width scaling. Also, the parametric convolution is re-analyzed by using the gradient flow propagation path in combination with different networks. Chien-Yao Wang [15] *et al.* after analyzing the

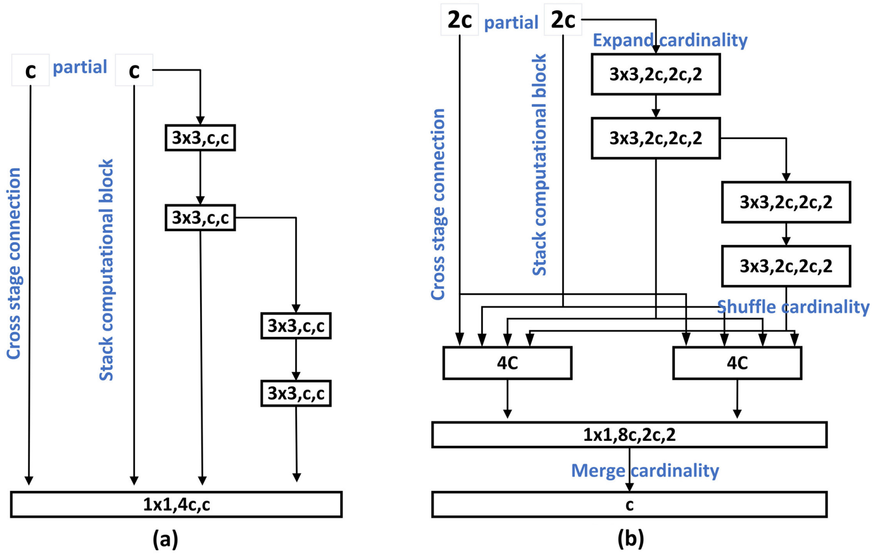


Figure 1. (a) ELAN and (b) E-ELAN.

combination and corresponding performance of RepConv with different architectures, used the RepConv without identity connection (RepConvN) to design the architecture of the planned re-parametric convolution, avoiding that the RepConv that the identity connection breaks the cascade of residuals and DenseNet in ResNet, thus providing more gradient diversity for different feature mappings. YOLOv7 proposes a new label assignment method which guides the auxiliary and bootstrap heads by bootstrap head prediction. In other words, the bootstrap head prediction is used as a guide to generate hierarchical labels from coarse to fine, which are used for auxiliary head and bootstrap head learning, respectively.

The Yolov7 network structure is composed of three parts as shown in Figure 2: Input, Backbone and Head. Backbone is used for feature extraction and Head is used for prediction. The input images are preprocessed, aligned into 640 × 640 (1280 × 1280) RGB images, and input to the Backbone network. The Head layer continues to process the Backbone network output image, completes the pyramid pooling process by SPPCSPC, completes the up sampling process by UPSample, and completes the feature map extraction of three layers with different size by ELAN variant ELAN-H. Feature map feature extraction combined with two CBS modules to complete the feature fusion from the Backbone network feature extraction. After REP and CBM, predict the three types of tasks (classification, background classification and border) of image detection, and output the final results.

2.2. ConvNeXt

Ashish Vaswani [16] *et al.* proposed Transformer in Attention Is All You Need. “Transformer” does not require recurrence and convolutions entirely, and is superior in model quality based solely on attention mechanics. At the same time,

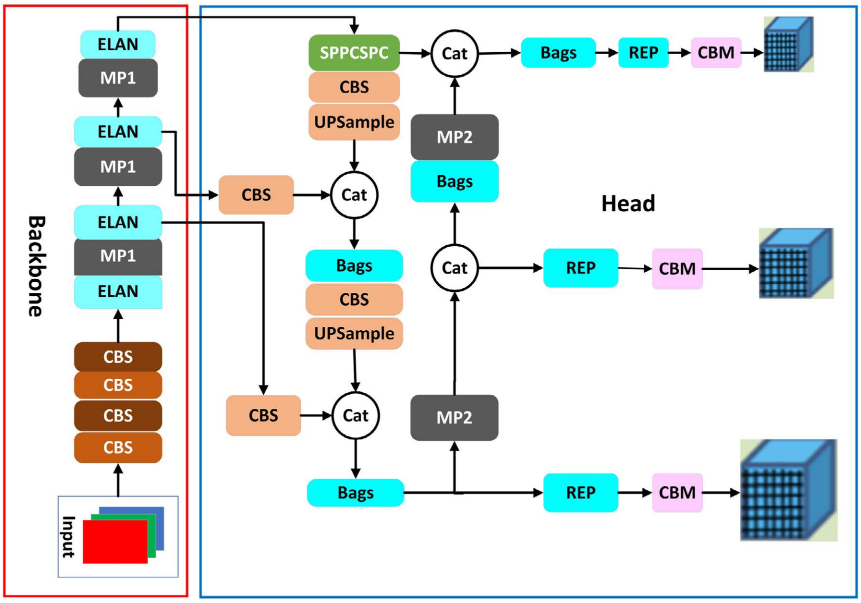


Figure 2. YOLOv7 network structure.

due to its parallelism, the training time is greatly reduced. The Swin Transformer backbone network also sets a new record for object detection and semantic segmentation. The academic community is convinced that Transformer architecture will become the new mainstream of visual modeling. Just when everyone was losing faith in CNN, A ConvNet for the 2020s came along. Zhuang Liu [17] *et al.* reviewed the ConvNet design space and designed and explored the standard ResNet in accordance with the Transformer architecture. The outcome of this exploration is a family of pure ConvNet models dubbed Con-vNeXt [17]. The accuracy of ConvNeXts, which is composed of standard ConvNet modules, is better than that of Swin Transformers in COCO detection and ADE20K segmentation, and keeps the simplicity and efficiency of ConvNets.

Zhuang Liu [17] *et al.* redesigned the ResNet Block based on Swin Tansformer Block and formed the ConvNeXt Block. The improvement process is shown in Figure 3: (The following accuracy data are all from the original literature [17]).

1) Example Change the stacking times of ResNet50 from (3, 4, 6, 3) to (3, 3, 9, 3). Stem is converted into a convolution layer with a convolution kernel size of 4 and a step size of 4.

2) Depth-separable convolution was used to construct the grouping convolution module in ResNeXt bottleneck block to achieve the goal of reducing FLOPs. At the same time, the width of the network was increased from 64 to 96 to compensate for the loss of capacity.

3) After using the Inverted Bottleneck module, the accuracy of bottleneck improves by 0.1% on small models and 0.7% on large ones.

4) Since the MSA module is placed before the MLP module in the Swin Tansformer Block, the depthwise conv is moved up here to follow suit. Move up the depthwise conv module in the ResNet Block from 1 × 1 conv → depthwise conv → 1 × 1 conv to depthwise conv → 1 × 1 conv → 1 × 1 conv.

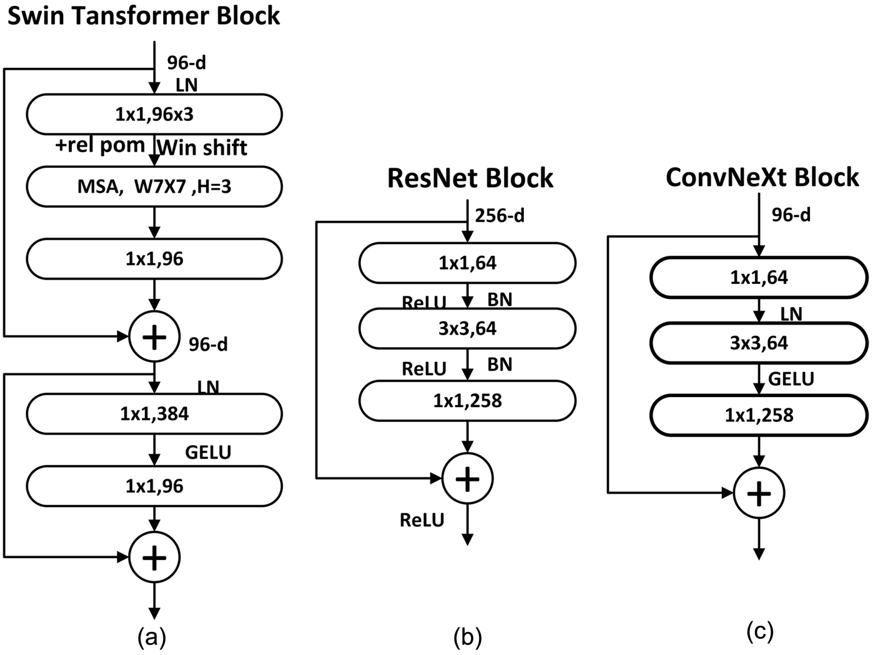


Figure 3. ConvNeXt Block structure.

5) Referring to Swin Tansformer Block, the convolution kernel size of depthwise conv was changed from 3 × 3 to 7 × 7. Accuracy increased by 0.7 percent.

6) As shown in Figure 3, b and a changed the ReLU activation function into GELU activation function, and reduced the use of activation function in ConvNeXt Block. After the reduction, the discovery accuracy increased from 80.6% to 81.3%.

7) Retain the Normalization layer after the depthwise conv. At this time, the accuracy rate has reached 81.4%, which is already higher than Swin-T network.

8) Replacing all BN with LN resulted in a small 0.1% improvement in accuracy. In the meantime, a separate down sampling layer is used in the ConvNext network, and it is formed with a Laryer Normalization plus a convolution layer with a convolution core size of 2 steps and a spacing of 2. The experimental results show that the accuracy rate is improved to 82.0%.

3. YOLOv7-CN-B

Zhuang Liu [17] *et al.* proposed the ConvNeXt Block and combined with three CBS modules to design the CN-B module. As shown in Figure 4, Figure 4(a): The CBS module is composed of Batch Normalization of Conv, BN (BN) and SilU activation functions. The process is shown in Formula (1) through Formula (6).

Output after Conv:

$$S_i = \{K \mid K_N = I_N, K_C = I_C\} \tag{1}$$

Si is the data subset whose mean and standard deviation need to be calculated. *i* is the serial number of the data and represents the location of the data. The

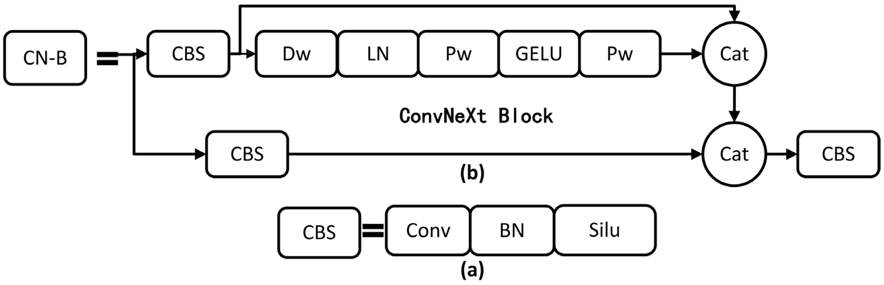


Figure 4. CN-B module.

data dimension after Conv is (*N*, *C*, *H*, *W*), where *N* represents batch size, *C* represents the number of channels of data, *H* and *W* represent the height and width of feature map respectively.

Batch normalization (BN):

Average difference:

$$\mu_i = \frac{1}{m}\sum_{k\in S_i}^{m} x_k \tag{2}$$

Standard deviation:

$$\sigma_i = \sqrt{\frac{1}{m}\sum_{K\in S_i}^{m}\left(X_k - \mu_i\right)^2 + \varepsilon} \tag{3}$$

Normalization:

$$\widehat{x_i} = \frac{1}{\sigma_i}\left(\sum_{i=1}^{n} X_i - \mu_i\right) \tag{4}$$

Restore the output:

$$y_i = \gamma\widehat{x_i} + \beta = \mathrm{BN}_{\gamma,\beta}\left(x_i\right) \tag{5}$$

γ, β as a learning parameter, ϵ of 10^{-5} class of constants.

SilU activation function handles:

$$f\left(y_i\right) = y_i \times \frac{1}{1+\mathrm{e}^{-y_i}} \tag{6}$$

As shown in Figure 4, Figure 4(b): After the image input of the CN-B module, the CBS module conducts longitudinal normalization processing to overcome the problem of gradient dispersion caused by the deepening of the neural network, which is difficult to train, and avoid network linearization at the same time. This is then passed to ConvNext Block and used as lateral Normalization, using LN (Layer Normalization)—the following Formula (7), avoiding the problems with the distribution of mini-batch data in BN and reducing memory usage. The GELU activation function—Formula (8) is used to make the network converge better; Reduce the calculation amount and prevent overfitting by subsampling; The large convolution kernel design increases the receptive field and enables the CN-B module to learn more global information. Finally, a CBS module is linked through Cat and output to the next CBS module.

LN (Layer Normalization):

$$y_i = \gamma \widehat{x_i} + \beta = \mathrm{LN}_{\gamma,\beta}\left(x_i\right) \tag{7}$$

γ, β as a learning parameter, ϵ of 10^{-5} class of constants. Unlike BN, which normalizes each channel of a batch of data, LN normalizes only the specified dimensions of a single data.

GELU activation function:

$$f\left(x\right) = x\int_{-\infty}^{x} \frac{\mathrm{e}^{\frac{(X-\mu)^2}{-2\sigma^2}}}{\sqrt{2\pi\sigma}}\mathrm{d}X \tag{8}$$

YOLOv7-CN-B Network Structure

As shown in Figure 5: The YOLOv7-CN-B network model is based on YOLOv7 and combined with the CN-B network module constructed in this paper to replace the ELAN module of the first and last image processing of the original YOLOv7, so that the backbone extraction network can obtain a larger receptive field and enhance the ability of flame feature extraction. Replace the Bags module (Trainable bag-of-freebies in YOLOv7) with the CN-B network module. Make the YOLOv7-CN-B model lighter than the original network model, increase the number of channels to accelerate the convergence rate, reduce the parameters, and reduce the calculation speed. Improve the network performance of the Head. Finally, YOLOv7-CN-B method achieved good results.

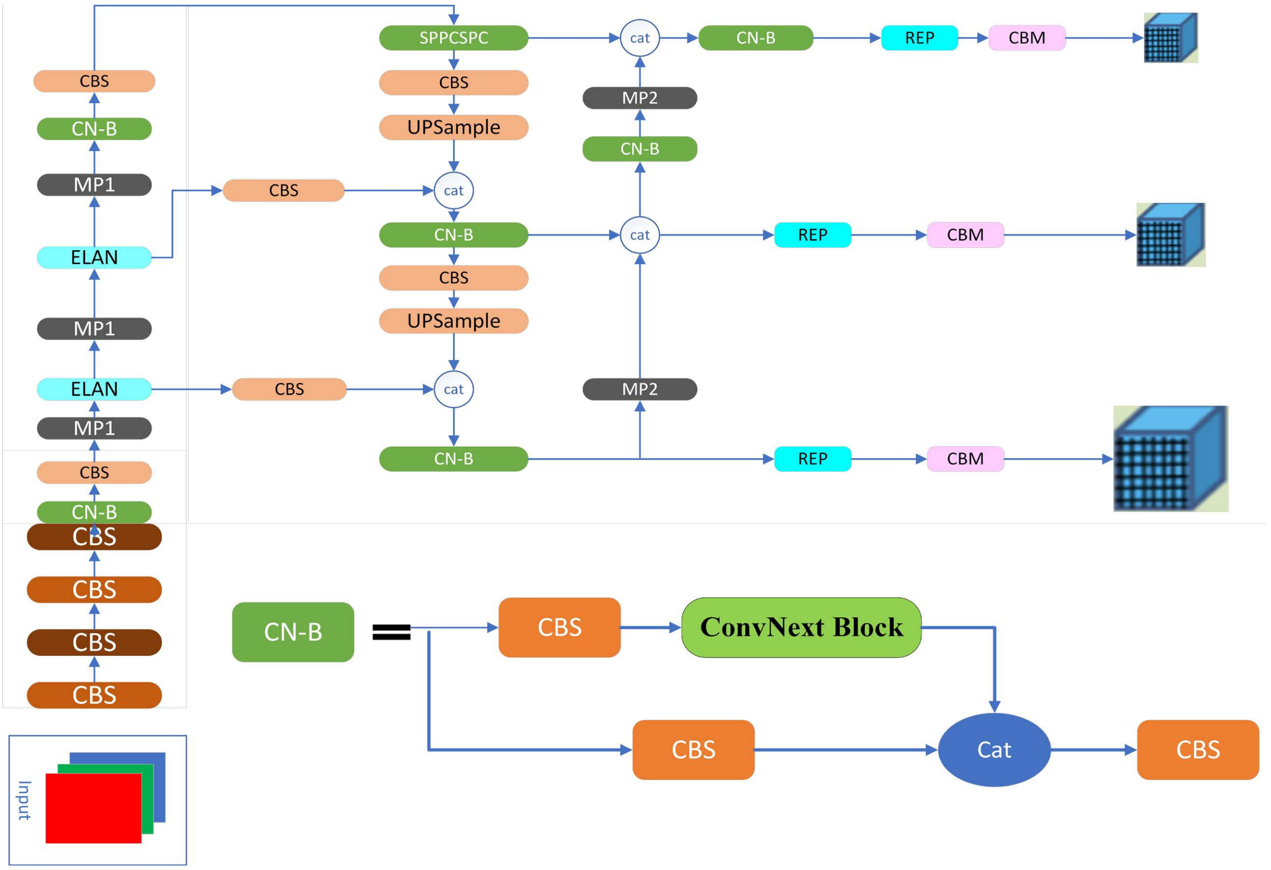


Figure 5. Network model of YOLOv7-CN-B.

4. Verification by Experiment

This paper adopts a small data set labeled with single flame. In order to evaluate the performance of the method, this paper uses the detection results of YOLO series to calculate evaluation indicators, including, PR curve, FPS, single detection speed and confusion matrix, as shown in Figure 6.

P (Precision) is used to reflect the model's ability to correctly predict the accuracy of positive samples. The higher the precision rate is, the better the model performance will be. Formula (9) is as follows:

$$\mathrm{TP} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{9}$$

TP: True class. The real category of samples is positive, and the result of model recognition is also positive. FP: The true class of the sample is a negative class, but the model recognizes it as a positive class.

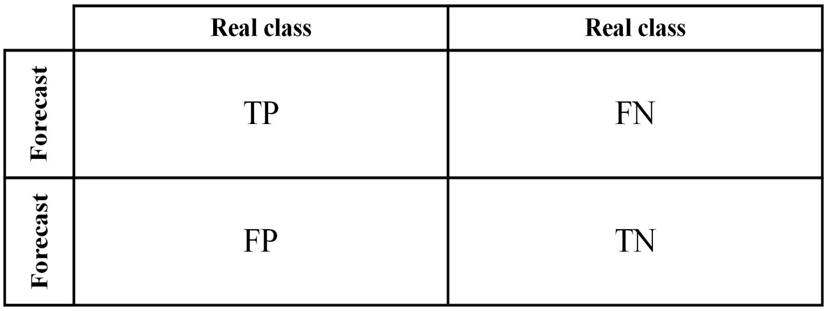
This paper calculates the Map (mean average precision) through the PR curve when most people agree that the IOU value is 0.5, which reflects the ability of the model to determine the correct category and the quantity TP of the correct category. The higher the mAP, the stronger the real prediction ability of the model. Formula (11) is as follows:

$$\mathrm{AP} = \int_0^1 P\left(\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}\right)\mathrm{d}R \tag{10}$$

$$\mathrm{mAP} = \frac{1}{C}\sum_{i=1}^{m}\mathrm{AP}_i \tag{11}$$

TP: True class. The real category of samples is positive, and the result of model recognition is also positive. FN: refers to the samples that are assigned as negative samples but incorrectly assigned, and represents the positive samples that are wrongly classified. AP: Average accuracy. Area enclosed by PR curve and coordinate axes, as shown in Formula (10) above. C Number of data set samples. In this paper, single flame labeling data is used, and C value is 1.

FPS: Used to assess the speed of flame detection, *i.e.* the number of images that can be processed per second. The more images, the faster the speed. In this paper, the default batch-size 32 detection speed of the YOLOv7 method is selected. The formula is as follows (12).



	Real class	Real class
Forecast	TP	FN
Forecast	FP	TN

Figure 6. Confusion matrix.

$$\text{FPS} = \frac{1(\text{ms})}{H} \tag{12}$$

4.1. Method of Experiment

In order to verify the effectiveness of the improved YOLOv7-CN-B method in flame detection, the original YOLOv7 method and YOLOv7-CN-B method were used in the experiment, and the experimental platform as shown in Table 1 was used for comparison experiments. The experiment adopted a public flame data set with 2059 flame images, which was annotated in flame YOLO format. Using this data set and excluding high quality data set, the improvement effect brought by the comparison experiment is completely dependent on the performance of the flame detection method itself. The training set and verification set were divided according to the ratio of 9:1, and the training was 250 rounds. Through the training weight of the training set, the accuracy and mAP values of the two are verified on the verification set. On batch-size 32, compare the FPS between the two: On batch-size1, compare the FPS between the two. The data set adopts the flame data set of the author: gengyanlei on Github. This flame data adopts a variety of scene flame data such as Night forest fire, Buying cabin fire, Road fire, smoke and fire, Daytime forest fire, candle fire, etc. As shown in Figure 7, there are partial data samples of the dataset.

4.2. Authors and Affiliations

As shown in Table 2, the accuracy of Yolov7-CN-B method reached 72% on the same experimental platform. Compared with the original YOLOv7 method, the accuracy of YOLOv7-CN-B method is improved by 5%. The detection speed on the default batch-size 32 was 149.25 FPS, an increase of 54 FPS over the 95.23 FPS of the original YOLOv7 method. The single detection speed reaches 11.9 (ms), which is 2.7 (ms) higher than the 14.6 (ms) of the original YOLOv7 method4.2. Identify the Headings.

Table 1. Experimental platform.

Device Name	Configuration
Operating system	Windows 10
CPU	AMD Ryzen 7-5800H@3.2 GHZ - 4.4 GHZ (Training load limit 3.8 GHZ)
GPU	NVIDIA GeForce RTX 3060 Laptop Power: 130 W
RAM	16 G
Memory	6 G
Deep learning Framework	Pytoch1.12.1
Environment of acceleration	CUDA11.6

Figure 7. Partial dataset. (a) Night forest fire; (b) Building caught fire; (c) Road fire; (d) smoke and fire; (e) Daytime forest fire; (f) Candle fire.

Table 2. Yolov7-CN-B Improved performance comparison.

Method	imagesize	P/%	mAP/%	FPS	Single sheet detection speed
Yolov7	640 × 640	67	67.6	95.23	14.6 ms
Yolov7-CN-B	640 × 640	72	69.7	149.25	11.9 ms

4.3. Experiment of Contrast

Yolov7-CN-B method Model Summary: 375 layers, 33,118,604 parameters, 33,118,604 gradients, 39.9 GFLOPS. Compared with the original YOLOv7 method, the Layers decreased by 9.6%, Parameters and Gradients decreased by 11.1%, and GFLOPS decreased by 62%. As shown in Figure 7, mAP of Yolov7-CN-B method reaches 69.7%, which is 2.1% higher than that of Yolov7-CN-B method. The results are shown in Figure 8.

5. Experiment of Contrast

The control variable method was used for comparison experiment. Under the same experimental conditions, the control of the number of training rounds is 250, the data set, the image input size is 640 × 640 and other variables remain unchanged, and only the experimental model is changed.

Experimental Conditions:

Hardware Conditions: Notebook CPU: R7-5800H (training limit frequency 3.8 G); GPU: NVIDIA RTX 3060 Laptop 6 G memory power 130 W).

Software Conditions: Pytoch1.12.1, CUDA11.6, Pycharm, Windows 10.

In order to verify the performance of YOLOv7-CN-B flame detection method by comparing YOLOv7-CN-B with advanced flame detection methods. The flame data set shown in Table 2 is still used. The general indexes of the deep learning network model, such as accuracy rate, mAP and single detection speed (ms), are selected to judge the superior performance.

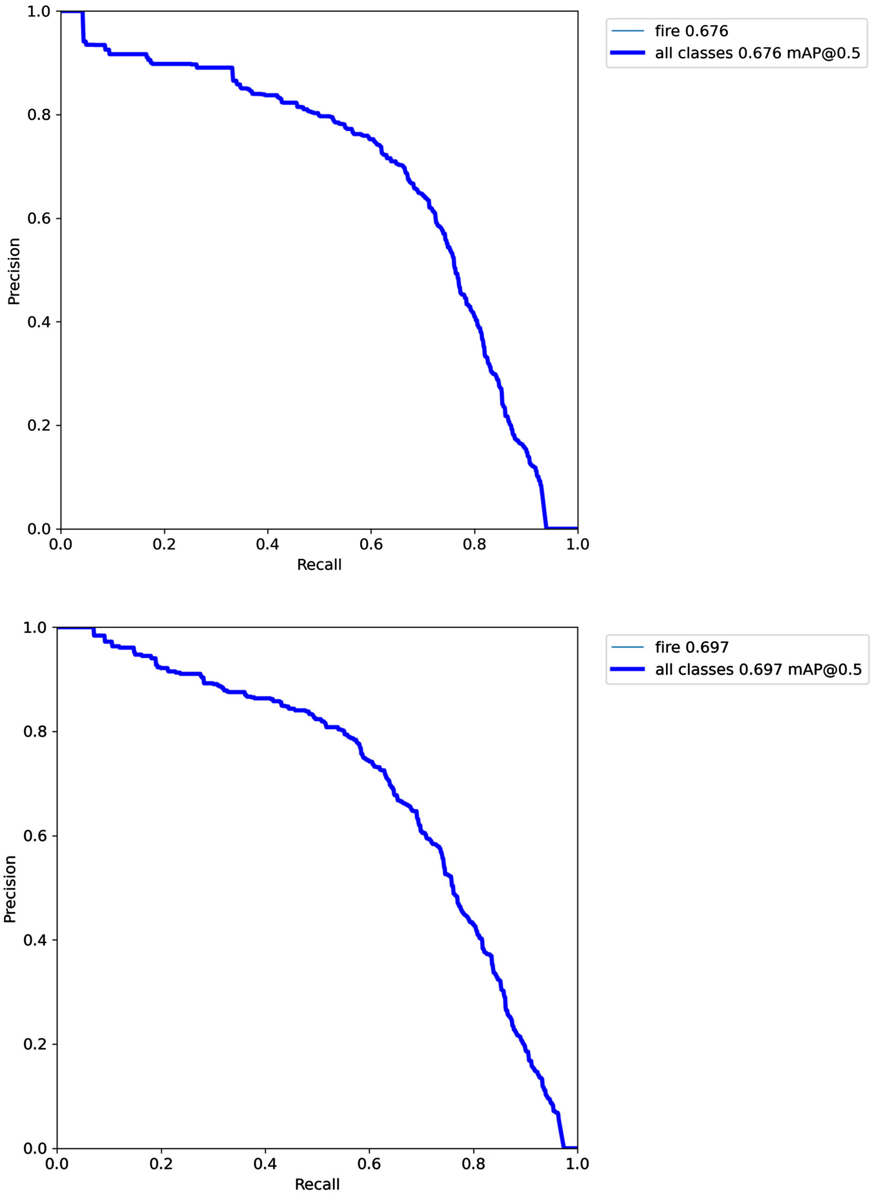


Figure 8. Comparison of PR curves.

As shown in Table 3, when the data size of the dataset image is 640 × 640, YOLOv7-CN-B method is nearly 10% higher than the mAP value of lightweight models such as YOLOv4-tiny, CenterNet, and YOLO5s, and its speed is also faster. Compared with Yolov3, Yolov4, Yolov7, Faster R-CNN and other models with high performance, YOLOv7-CN-B method is also about 6% higher on average. It is much faster than the above model in terms of single sheet detection speed. YOLOv7-CN-B method has the detection speed only with high mAP value and lightweight model. There is no doubt that YOLOv7-CN-B method is superior to other models in terms of detection accuracy, mAP value and detection speed. It can be found that YOLOv7-CN-B method is more efficient than other methods.

Table 3. Comparison experiment.

Method	Image size	mAP/%	Single sheet detection speed
Yolov3 [18]	640 × 640	62.58	32 ms
Yolov4 [10]	640 × 640	61.91	42 ms
Yolov4-tiny [19]	640 × 640	59.14	15 ms
Yolo5s [20]	640 × 640	61.5	14 ms
Faster R-CNN [12]	640 × 640	63.4	30 ms
EfficentDet-D0 [9]	640 × 640	62.4	39 ms
CenterNet [21]	640 × 640	50.13	19 ms
Yolov7 [15]	640 × 640	67.6	14.6 ms
Yolov7-CN-B	640 × 640	69.7	11.9 ms

6. Conclusions

Based on the most advanced target detector YOLOv7, the YOLOV7-CN-B flame detector is proposed. The YOLOv7 network model is lightweight, so the YOLOv7-CN-B method can obtain larger receptive field, higher precision and mAP, smaller Parameters, Gradients, Layers and less computation, breaking the inherent detection speed of YOLOv7. Through experimental verification, compared with the original YOLOv7 method, YOLOv7-CN-B method reduced Layers by 9.6%, Parameters and Gradients by 11.1%, and GFLOPS by 62%. Accuracy has been improved by 5% and mAP has been improved by 2.1%. The detection speed on the default batch-size 32 is 149.25 FPS, which is 54 FPS higher than the original YOLOv7 method's 95.23 FPS, and the single piece detection speed on the batch-size1 is also 11.9 ms. The comprehensive performance is better than that of the YOLOv7 method. By contrast experiment and advanced flame detector comparison. The YOLOv7-CN-B method proposed in this paper is superior to other models in terms of detection accuracy, mAP and detection speed. It can be found that the YOLOv7-CN-B method is more efficient than other methods.

The improved YOLOv7-CN-B method solves the problem that the traditional flame detection sensor is limited to a fixed and closed small space, and the detection ability of a single flame is limited. The conditions for outdoor flame detection and flame detection with temporal and spatial characteristics are satisfied. This paper provides a new high performance detection network model for flame detection in the world.

Acknowledgements

Fund project for publishing papers: Sichuan Key Science and Technology Research and Development Project (2019YFG0200); Sichuan Science and Technology Innovation (Miao Project) Cultivation Project (2022049).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Chen, J., He, Y.P. and Wang, J. (2009) Multi-Feature Fusion Based Fast Video Flame Detection. *Building and Environment*, **45**, 1113-1122. https://doi.org/10.1016/j.buildenv.2009.10.017

[2] Lasaponara, R., Cuomo, V., Macchiato, F.M., *et al.* (2003) A Self-Adaptive Algorithm Based on AVHRR Multitemporal Data Analysis for Small Active Fire Detection. *International Journal of Remote Sensing*, **24**, 1723-1749. https://doi.org/10.1080/01431160210144723

[3] Celik, T., Demirel, H., Ozkaramanli, H., *et al.* (2006) Fire Detection Using Statistical Color Model in Video Sequences. *Journal of Visual Communication and Image Representation*, **18**, 176-185. https://doi.org/10.1016/j.jvcir.2006.12.003

[4] Zhou, X.L., Yu, F.X., Wen, Y.C., *et al.* (2010) Early Fire Detection Based on Flame Contours in Video. *Information Technology Journal*, **9**, 899-908. https://doi.org/10.3923/itj.2010.899.908

[5] Zhao, Y., Zhu, J., Xie, Y., *et al.* (2021) Improved Yolo-v3 Video Image Flame Real-Time Detection Algorithm. *Journal of Wuhan University* (*Information Science Edition*), **46**, 326-334.

[6] Wang, G., Yan, Y., Gao, S., *et al.* (2022) Flame Detection Algorithm Based on Fire-YOLO. *Computer and Communication* (*Theory Edition*), **34**, 49-52.

[7] Hu, Y., Wang, H.Q., Huang, D.Y., *et al.* (2015) Adaptive Feature Selection of Flame Image Based on Improved GA-RS. *Computer Engineering*, **41**, 186-189.

[8] Lin, T.-Y., Goyal, P., Girshick, R., He, K.M. and Dollar, P. (2020) Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **42**, 318-327. https://doi.org/10.1109/TPAMI.2018.2858826

[9] Tan, M., Pang, R. and Le, Q.V. (2020) EfficientDet: Scalable and Efficient Object Detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seattle, 13-19 June 2020, 10778-10787. https://doi.org/10.1109/CVPR42600.2020.01079

[10] Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y.M. (2020) YOLOv4: Optimal Speed and Accuracy of Object Detection.

[11] Girshick, R.B. (2015) Fast R-CNN. https://doi.org/10.1109/ICCV.2015.169

[12] Ren, S.Q., He, K.M., Girshick, R.B. and Sun, J. (2015) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.

[13] He, K.m., Gkioxari, G., Dollár, P. and Girshick, R.B. (2017) Mask R-CNN. 2017 *IEEE International Conference on Computer Vision* (*ICCV*), Venice, 22-29 October 2017, 2980-2988.

[14] Yang, F., Zhang, X.L. and Liu, B. (2022) Video Object Tracking Based on YOLOv7 and DeepSORT.

[15] Wang, C.-Y., Bochkovskiy, A. and Liao, H.-Y.M. (2022) YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors.

[16] Vaswani, A., Shazeer, N., Parmar, N., *et al.* (2017) Attention Is All You Need. *Advances in Neural Information Processing Systems* 30: *Annual Conference on Neural Information Processing Systems* 2017, Long Beach, 4-9 December 2017, 5998-6008

[17] Liu, Z., Mao, H.Z., Wu, C.-Y., Feichtenhofer, C., Darrell, T. and Xie, S.N. (2022) A ConvNet for the 2020s.

[18] Redmon, J. and Farhadi, A. (2021) YOLOv3: An Incremental Improvement. https://arxiv.org/pdf/1804.02767.pdf

[19] Jiang, Z.C., Zhao, L.Q., Li, S.Y., *et al.* (2020) Real-Time Object Detection Method Based on Improved YOLOv4-Tiny.

[20] Ultralytics.YOLOv5. https://github.com/ultralytics/YOLOv5

[21] Zhou, X.Y., Wang, D.Q. and Krähenbühl, P. (2021) Objects as Points. https://arxiv.org/pdf/1904.07850.pdf