A lightweight global awareness deep network model for flame and smoke detection*

XIAO Bowei and YAN Chunman**

School of Physics and Electronic Engineering, Northwest Normal University, Lanzhou 730070, China

(Received 5 March 2023; Revised 19 April 2023)

Aiming at the trouble of low detection accuracy and the problem of large model size, this paper proposes a lightweight flame-and-smoke detection model depending on global awareness of images. The proposed method replaces the Conv+BatchNorm+SiLU (CBS) module of original you only look once version 5 (YOLOv5) in the backbone with DSConv+BatchNorm+SiLU (DBS), and the C3 module with GC3, and thus constructs a lightweight backbone network. Besides, involution (InvC3) module is proposed to enhance the global modeling ability and compress the model size, and a module using adaptive receptive fields, named FConv, is proposed to enhance the model's perception capacity for foreground complex flame-and-smoke information in feature maps. Experimental results show that the proposed model increases the mean average precision of all categories at 0.5 IOU (mAP@0.5) to 70.8%, the mAP@0.5: 0.95 to 39.7%, reduces the number of parameters to 3.57M and the amount of calculation to 7.4 giga floating-point operations per second (GFLOPs) under the premise of ensuring the detection speed. It has been verified that the model can achieve high-precision real-time detection of flame and smoke.

Document code: A **Article ID:** 1673-1905(2023)10-0614-9

DOI https://doi.org/10.1007/s11801-023-3041-x

The occurrence of fire posed a significant risk to human life and property. So, in recent years, some countries have put a lot of effort into building fire protection systems to keep fires from happening and cut down on losses. In order to provide early warning of fire threats, accurate and quick detection of flame and smoke is essential. Since an actual fire is usually accompanied by smoke, the flame is typically utilized as the primary object in the task of detecting fire and smoke. The smoke complements the scenario information and will be detected at the same time. The detection results are output as two targets: fire and smoke.

The early-stage detection methods mainly used a detection system based on physical sensors that obtained physical information such as smoke, heat, and light in the space and compared the information with the threshold set in the program after digital-to-analog conversion to verify the actual detection situation. However, the system's huge volume and expensive price make it difficult to equip outdoors, in small spaces, or in multi-story buildings. Moreover, its detection speed is relatively slow, which means it cannot give early warning to dangerous situations in time.

With the improvement of storage and computing power of embedded monitoring equipment, researchers have put out a few methods for detecting flames and smoke using typical machine learning with manually created feature classifiers[1-4], which can detect early-stage fire and smoke in color, motion, and texture and perform better than physical sensor-based detectors in terms of deployment flexibility and detection quickness. However, there are issues with these types of detectors, including sluggish speed, a high false detection rate, poor generalization ability, and low adaptability to the environment.

When it comes to deep learning, object detection algorithms based on convolutional neural networks (CNNs) are frequently utilized in cutting-edge fields such as autonomous driving[5] and medical image processing[6]. The classical CNN-based object detection models include faster region-based CNN (faster R-CNN)[7] as two-stage models, single shot MultiBox detector (SSD)[8], you only look once (YOLO)[9], and RetinaNet[10] as one-stage models. Among them, YOLO better balances the relationship between detection accuracy and speed. With the emergence of feature pyramid networks (FPN)[11] and path aggregation networks (PAN)[12], the YOLO series models have completed the change from YOLOv1 to YOLOv5[13-15]. In terms of better detection accuracy, speed, and robustness, researchers have widely used CNN-based models to perform flame and smoke detection tasks. CHAOXIA et al[16] put out a faster R-CNN based detection model with a color-guided anchoring strategy, which uses color features of the flame

* This work has been supported by the National Natural Science Foundation of China (No.61961037), and the Gansu Provincial Department of Education 2021 Industry Support Program (No.2021CYZC-30).

** E-mail: yancm2022@163.com

to limit anchor point positions, thereby improving the detection accuracy. However, there exist problems of large model volume and slow detection speed. SHI et al[17] proposed a SSD detection model based on the DenseNet, which has the ability to detect small targets and reduces the missed detection rate for flame and smoke targets. Although the model has a fast detection speed, the false positive rate is high. HU et al[18] proposed a multi-directional flame and smoke detection model based on YOLOv5 that added a value conversion attention mechanism module and a mixed-NMS module. However, it is difficult to deploy the model due to its large size. CAI et al[19] improved the residual model using the channel attention mechanism, added DropBlock after each convolutional layer, and proposed a high-precision smoke detection model based on YOLOv5. But the model size is huge, and the detection effect is poor when the smoke and the background color are relatively close. ZHANG et al[20] proposed a YOLOv5-based flame and smoke detection model that combines swin transformer and weighted splicing modules, which improves detection accuracy by enhancing feature extraction and fusion capabilities. Nevertheless, the self-attention mechanism[21] brings a huge amount of calculation, and the model has a poor detection effect on scenes where flames and smoke cover each other.

Although the above-mentioned CNN-based flame and smoke detection models have achieved high detection accuracy and speed, there still exist several problems. The model structures are relatively complex, and the amount of parameters and calculations is large, which leads to a large amount of redundant information in the feature extraction process. In actual flame-and-smoke detection scenario, both of them have indefinite shapes, while the smoke also has a certain degree of transparency, which might make the smoke blend within the background of the scene, and sometimes the flame and smoke might also shield each other, which puts high demands on the global space modeling ability of the detection model.

Considering the existing problems, this paper proposes a lightweight YOLOv5-improved flame and smoke detection model based on global awareness (global modeling + adaptive receptive field): global-awareness and lightweight YOLOv5 (GAL-YOLOv5). The following is a summary of this paper's significant contributions.

A lightweight feature extraction network (DS-Conv+BatchNorm+SiLU (DBS) and GC3 net, DGNet) is proposed. The C3 in CSPDarknet53 is replaced with the GC3 composed of the ghost model, and the Conv+BatchNorm+SiLU (CBS) is replaced with the DBS composed of depth-wise separable convolution. DGNet compresses the model volume to 63% of YOLOv5s while ensuring the detection accuracy, which greatly reduces the large amount of redundant information generated during the feature extraction process.

A lightweight module involution C3 (InvC3) with global modeling capabilities is proposed. Involution is introduced into the C3 in the original neck, which further compresses the model volume while improving the accuracy of flame and smoke detection.

An adaptive receptive field module (FRelu convolution (FConv)) is proposed. FRelu is introduced into the CBS in the original neck, which strengthens the model's ability to perceive and process complex foreground flame and smoke information in the feature map. And FConv further optimizes the global modeling ability of InvC3, and the combination of the two gives the model global awareness ability of the flame and smoke characteristics.

Among the four models of YOLOv5s, YOLOV5m, YOLOv5l, and YOLOv5x, YOLOv5s has the fastest detection speed and the smallest model size. Therefore, this paper uses it as a benchmark model, improves it, and obtains GAL-YOLOv5 that takes into account both detection accuracy and speed. GAL-YOLOv5 is mainly composed of three parts, DGNet, improved neck and head, as shown in Fig.1.

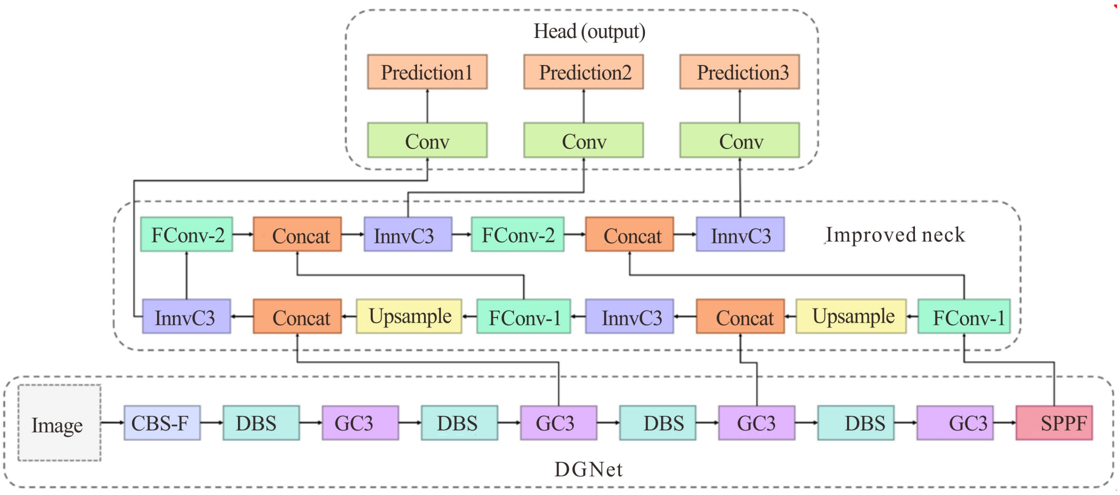


Fig.1 Network structure of GAL-YOLOv5

The backbone of YOLOv5s has a large volume and generates a lot of redundant information in the process of feature extraction. Therefore, we designed a new lightweight backbone network, DGNet. Among them, CBS-F represents the downsampling module formed by concatenating ordinary convolution with a kernel size of 6 and a step size of 2 with BatchNorm and SiLU, which achieves the same function as focus in the original backbone and has a lower amount of calculation.

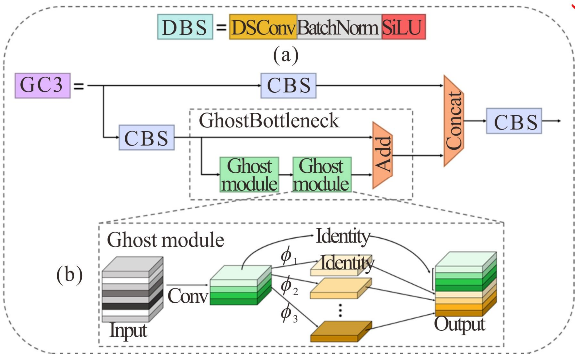


Fig.2 (a) Structure of DBS module; (b) Structure of GC3 module

In DGNet, the C3 module in the original backbone is replaced with the GC3 module, and the model volume is greatly reduced while the accuracy of detection is maintained. In GC3, two CBS modules with a convolution kernel size of 3 and a step size of 1 are first used to reduce the dimensionality of the input feature map channel. After that, one of the outputs will go through GhostBottleneck. The ghost module in it is a lightweight feature extraction method proposed by HAN et al in GhostNet[22], which can solve the problem of a large amount of redundant information generated by ordinary convolution during feature extraction. The calculation of the ghost module is mainly divided into three steps. Firstly, the input feature map is transformed into an intrinsic feature map by using a few convolution kernels. Then, use the cheap operation ϕ to effectively convert the intrinsic feature map into a ghost feature map. And finally, the intrinsic feature map is stitched with the ghost feature map.

Use the Concat operation to stitch the output of GhostBottleneck with the output of another channel in the channel dimension, and finally, use the same CBS module to adjust the number of channels and output the feature map.

In addition, the DBS module in this article also takes on the role of the CBS module in the original backbone. Among them, the ordinary convolution of the CBS module is replaced with a depth-wise separable convolution[23] with fewer parameters and calculations. Depth-wise separable convolution is divided into two parts. The first part is depth-by-depth convolution, which applies a convolution kernel to each of the input feature map's channels and then splices the output of all convolution kernels to obtain the intermediate feature maps of channel separation. The second part is point-by-point convolution, which fuses channels separated from each other in the intermediate feature map. DBS assists GC3 in feature extraction while achieving efficient 2-fold downsampling of feature maps, further compressing the model volume and improving detection accuracy.

In order to improve the model's global space modeling ability, this paper shows an improved neck based on YOLOv5's neck, which further compresses the model's volume while solving the problems in the actual flame and smoke detection.

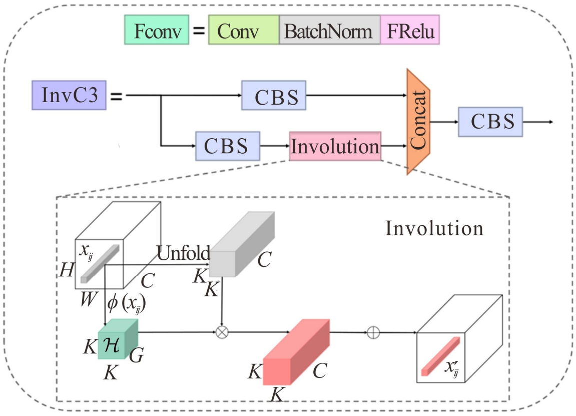


Fig.3 (a) Structure of FConv module; (b) Structure of InvC3 module

In the improved neck, first replace all C3 modules in the original neck with InvC3 modules. Involution in InvC3 is a lightweight operator proposed by LI et al[24] with space specificity and channel independence. Compared with ordinary convolution, involution has the following advantages. Contextual information can be summarized over a wider spatial range. It can adaptively assign different weights to different spatial positions, which is beneficial for finding visual elements that contribute more to the foreground in the spatial domain. The same global feature extraction and modeling capabilities as the self-attention mechanism can be obtained without position encoding. It has fewer parameters and calculations. The calculation process of involution is mainly divided into two parts. Firstly, two convolution operations $\phi(x_{ij})$ are performed on the input feature map to obtain the involution kernel $\mathcal{H}$, and the kernel size is K. Then, the involution kernel is multiplied and added to the unfolded input feature to generate the involution output feature map. G is the number of output channel groups, and the kernel can only be shared by output channels belonging to the same group.

Thanks to involution's excellent global modeling capabilities, InvC3 can more accurately extract flame and smoke feature information in the feature map space domain. Moreover, its calculation amount and parameter amount are lower than that of C3, which solves the above problems and satisfies the lightweight model's design requirements.

After that, replace the CBS module in the original neck with the FConv module. Due to the spatial insensitivity of the SiLU activation function[25] in CBS, it cannot adaptively acquire spatial dependencies. Therefore, replace it with the FRelu activation function[26] to form the FConv module, where 1 and 2 represent the convolution step size. FRelu can provide the model with pixel-level modeling capabilities and add an adaptive receptive field to the model, making it easy to extract the spatial structure of foreground objects. After FRelu obtains the surrounding information of each pixel, it performs a weighted summation.

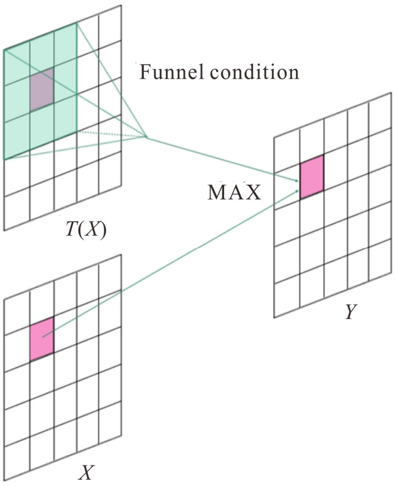


Fig.4 Calculation process of FRelu

Compared with Relu[27] (MAX(x, 0)), FRelu transposes the discriminant condition into the funnel condition $T(x)$, uses the MAX(·) function to obtain the maximum value between x and the discriminant condition, and (·) represents the dot product. The specific implementation of FRelu is as follows

$$f\left(x_{c,i,j}\right)=\mathrm{MAX}\left[x_{c,i,j},T\left(x_{c,i,j}\right)\right], \tag{1}$$

$$T\left(x_{c,i,j}\right)=x_{c,i,j}^{w}\cdot p_{c}^{w}, \tag{2}$$

where $x_{c,i,j}$ represents the input pixel of the nonlinear activation function $f(\cdot)$ on the cth channel at the 2D spatial position (i, j); $x_{c,i,j}^{w}$ is the parameter pooling window centered around $x_{c,i,j}$; moreover, p_{c}^{w} is the parameter shared in the same channel on this window.

FConv enables the model to perceive and process more complex foreground information in the feature map. Moreover, the adaptive receptive field provided by FConv further optimizes the global modeling ability of InvC3, and the two together increase the global awareness ability of the flame and smoke characteristics for the model, thereby further enhancing the detection accuracy of the model.

Due to the lack of flame and smoke datasets with accurate annotation and strong completeness, in order to ensure the robustness and generalization ability of the trained model, we have established a flame-smoke dataset with high-quality images and annotations, as shown in Fig.5. When building this dataset, web crawlers were used to crawl some images on the Internet, and cameras were used to manually collect some images in daily life. The images are labeled by LabelImg, and a total of 15 944 color-labeled pictures with resolutions ranging from 112×112 to 2 048×2 048 are obtained. The self-built dataset contains almost all possible scenes of smoke and flames, including 8 065 urban scenes, such as buildings, factories, communities, roads, and vehicles, and 7 879 outdoor scenes such as forests and grasslands. 9 073 images are in the daytime, and 6 871 are at night. It contains a total of 17 013 fire objects and 9 048 smoke objects. Abundant training samples will greatly improve the robustness and versatility of the algorithm. In this study, the data set was randomly divided into a training set and a verification set in the ratio of 8: 2.

Fig.5 Example images of the dataset

The computer hardware platform used for the work in this paper is Intel(R) Xeon(R) CPU E5-2680 v4, the memory capacity is 64 GB, and the GPU is NVIDIA RTX3090. The operating system is Windows 10, the Pytorch1.8.2 deep learning framework is adopted, and the programming language is Python.

In this paper, random initialization for model parameters is used in the training. The input image size during training is 640×640, the training batch size is 32, the initial learning rate is 0.001, and the total number of training epochs is 150. The iterative optimization algorithm adopts stochastic gradient descent (SGD), and the momentum coefficient is 0.937.

The total loss of GAL-YOLOv5 is mainly obtained by linearly superimposing the regression loss ($Loss_{\mathrm{box}}$), the classification loss ($Loss_{\mathrm{cls}}$), and the confidence loss ($Loss_{\mathrm{obj}}$) with weights α, β, and γ.

$$Loss=\alpha Loss_{\mathrm{box}}+\beta Loss_{\mathrm{cls}}+\gamma Loss_{\mathrm{obj}}. \tag{3}$$

Since the shapes and positions of flame and smoke are relatively random, and often cover each other, we set α to 0.5 so that the model can accurately locate the objects to be detected. Secondly, we set β to 0.3, and thus the model can accurately make classifications under the premise of precise positioning. Finally, γ is set to 0.2 to provide a high confidence value for detected objects. After many experiments and comparisons, it is found that the model trained with the loss function configured with the above parameters has the highest detect accuracy.

In order to verify the performance of GAL-YOLOv5, precision, recall, mean average precision of all categories at 0.5 intersection over union (IOU) (mAP@0.5), mAP@0.5: 0.95, params (the sum of each layer's parameters in the network structure), FLOPs (the number of floating-point operations performed by the model), FPS (the number of images the model detects per second), and model size (the weight size of the trained model) are used as measurement indicators.

Precision indicates the probability of correct prediction in the predicted positive samples, and recall indicates the probability that the actual positive samples are correctly predicted. Its definition is as follows

$$Precision = \frac{TP}{TP + FP}, \tag{4}$$

$$Recall = \frac{TP}{TP + FN}, \tag{5}$$

where *TP* denotes the amount of pictures that are rightly predicted as positive samples, *FP* denotes the amount of pictures that are erroneously predicted as positive samples, and *FN* denotes the amount of pictures that are erroneously predicted as negative samples.

mAP@0.5 indicates the mean average precision (AP) of all categories at 0.5 IOU. mAP@0.5: 0.95 indicates the average value of mAP under varied IOU thresholds, with a 0.05 step size:

$$AP = \int_0^1 P(R)\mathrm{d}R, \tag{6}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N}. \tag{7}$$

In order to explore the impact of DGNet, FConv and InvC3 on the performance of flame and smoke detectors, different combination methods are designed for ablation experiments in this section. The experimental results for 640×640 input images are displayed in Tab.1. Among them, precision and recall represent the average value of the evaluation indicators corresponding to the two types of objects, flame and smoke.

It can be seen from Tab.1 that the use of DGNet effectively reduces the redundant information in the model, and the params, FLOPs, and model size are reduced to 36.4%, 42.5%, and 35% of the original ones. The time complexity is also reduced, and the detection speed is increased by 4 frames. The values of mAP@0.5, mAP@0.5: 0.95, and precision are increased by 1%, 3.4%, and 4.1%, respectively. Even though the value of recall decreases, it is slight and negligible. DGNet backbone improves detection speed and accuracy while reducing model volume, and it is set as the benchmark for our model design. FConv and InvC3 are added in subsequent ablation experiments for comparison.

When only FConv is added, the params and time complexity of the model are slightly increased because a certain number of convolutional layers are used in FRelu.

Tab.1 Results of ablation experiments

YOLOv5s	DGNet	FConv	InvC3	mAP@0.5	mAP@0.5: 0.95	Precision	Recall	Params (M)	FLOPs (G)	FPS	Weights (M)
✓				0.693	0.381	0.702	**0.636**	7.03	16.0	52	13.7
✓	✓			0.700	0.394	0.731	0.629	4.47	9.2	**56**	8.9
✓	✓	✓		0.704	**0.399**	0.734	0.624	4.48	9.2	53	8.9
✓	✓		✓	0.703	0.394	0.730	0.63	**3.55**	7.4	54	**7.1**
✓	✓	✓	✓	**0.708**	0.397	**0.743**	0.626	3.56	**7.4**	52	7.2

However, this structure brings an adaptive receptive field to the model and improves its ability to perceive and process complex flame and smoke scenes in the foreground of the feature map, increasing mAP@0.5, mAP@0.5: 0.95, and precision by 0.5%, 1.2%, and 0.4%, respectively.

When InvC3 is added only, the model volume is compressed more, and the amounts of params, FLOPs, and model size are reduced by 20.6%, 19.6%, and 20.2%, respectively, while mAP@0.5 rose 0.4%, which illustrates that the involution in the InvC3 module has a lower computational complexity compared to ordinary convolution, and the global modeling ability of the model is strengthened.

In order to obtain a lightweight model that is sensitive to scenes such as flame and smoke with uncertain shapes, mutual occlusion, and high smoke transparency, we added the above two modules to DGNet at the same time to construct the model proposed in this paper, GAL-YOLOv5.

Compared with the benchmark (YOLOv5s), GAL-YOLOv5 significantly reduced the model volume, and mAP@0.5, mAP@0.5: 0.95 and precision increased by 1.1%, 0.8%, and 1.6%, respectively, which proves that the adaptive receptive field provided by FConv further optimizes the global modeling ability of InvC3, and the two together bring the model the global awareness ability of flame and smoke characteristics, which makes up for the shortcomings of the original model that caused poor detection accuracy due to spatial insensitivity and poor global feature modeling capabilities.

GAL-YOLOv5 is based on YOLOv5s. By replacing DGNet, InvC3, and FConv, the model volume is greatly reduced and the detection accuracy is improved while ensuring the detection speed.

In order to further explore the comprehensive detection performance of GAL-YOLOv5, it is applied to the flame and smoke detection task with several classic object detection models and lightweight object detection models based on YOLOv5s, and comparisons are made. For fair results, the above models are trained and verified using the self-built dataset. The input image size is 640×640, and 150 training epochs are executed on this experimental platform. All the models in the comparison experiments are trained by the transfer learning method of loading pre-trained weights. The experimental results are shown in Tab.2, where (B) represents the model structure where only the backbone of YOLOv5s is replaced and GAL-YOLOv5(C) represents the model structure after the positions of InvC3 and GC3 in GAL-YOLOv5 are exchanged.

It can be seen from Tab.2 that the GAL-YOLOv5's mAP@0.5 and mAP@0.5: 0.95 are better than other modules in the comparison group. Compared with the classic modules of YOLOv5s, SSD, RetinaNet, and faster R-CNN, mAP@0.5 has increased by 2.2%, 16.4%, 1.3%, and 10%, respectively, and mAP@0.5: 0.95 increased by 4.2%, 47%, 6.7%, and 28.5%, respectively. We think this is due to the model's excellent global awareness ability, so that it can better learn the characteristics of flame and smoke.

It can be seen from Fig.6 that the slope of the accuracy curve (red dot-dash line) of GAL-YOLOv5 maintains a large value at the beginning of training and then tends to keep stable after the model converges. Moreover, the model's accuracy is much higher than that of the group used as a comparison. In addition, the accuracy curve (purple dash) of YOLOv5 will appear overfitting after 50 epochs of training. By comparing the two curves of GAL-YOLOv5 and YOLOv5s, it can be seen that the improved model designed here not only improves the accuracy, but also solves the overfitting problem of the original model and improves the robustness of the algorithm.

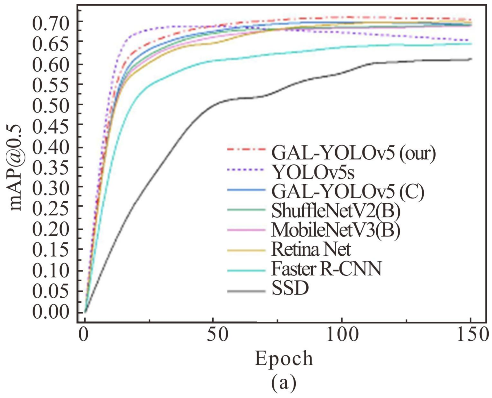
Tab.2 Results of comparative experiments

Method	Backbone	mAP@0.5	mAP@0.5: 0.95	Precision	Recall	Params (M)	FLOPs (G)	FPS	Weights (M)
RetinaNet	ResNet50	0.699	0.372	0.855	0.533	36.40	163.8	26	139.0
Faster R-CNN	VGG16	0.644	0.309	0.348	**0.780**	136.71	401.7	24	108.2
SSD	VGG16	0.608	0.270	**0.868**	0.312	23.70	273.6	**59**	91.1
YOLOv5s	CSPDarkNet53	0.693	0.381	0.702	0.636	7.02	16.0	52	13.7
ShuffleNetV2(B)	ShuffleNetV2[28]	0.69	0.372	0.717	0.623	**3.39**	7.4	49	6.8
MobileNetV3(B)	MobileNetV3[29]	0.689	0.372	0.726	0.61	3.53	**6.1**	46	**7.0**
GAL-YOLOv5(C)	—	0.697	0.387	0.727	0.613	3.87	8.7	49	7.8
GAL-YOLOv5	DGNet	**0.708**	**0.397**	0.743	0.626	3.57	7.4	52	7.1

In addition, compared with YOLOv5s, SSD, RtinaNet, and faster R-CNN, the params of GAL-YOLOv5 has decreased by 49.1%, 85.3%, 89.3%, and 97.3%, respectively. The FLOPs has decreased by 53.8%, 97.3%, 95.5%, and 99.8%, respectively, and the model size has decreased by 48.2%, 92.3%, 94.9%, and 93.4%, respectively. We think this is due to the fact that DGNet reduces a substantial amount of redundant information generated during the feature extraction process and compresses the model volume to a lower value.

When compared with YOLOv5s, the precision of GAL-YOLOv5 has increased by 5.8%. Although the recall has decreased slightly, the impact of this change is negligible. The precision of GAL-YOLOv5 has doubled that of faster R-CNN. This is because faster R-CNN is more suitable for dealing with problems with fewer cross-features, but the actual flame and smoke are in an overlapping relationship in most cases, which leads to a significant increase in cross-features, resulting in a lower precision of the model. The model we proposed solves this problem well. When compared with SSD and RetinaNet, which are both single-stage detection models, the precision of GAL-YOLOv5 drops slightly. This is due to the significant computational complexity of SSD, and the FocalLoss[10] used in RetinaNet which improves the balance of positive and negative samples will improve the precision of the model. However, the detection speed of RetinaNet is only 27 frames, which makes it hard to achieve real-time detection, and the huge model volume of SSD cannot meet the light-weight requirements of the detector, so GAL-YOLOv5 has superior overall performance compared to the other two.

In order to further test the performance of GAL-YOLOv5, we compared it with the lightweight object detection model also based on YOLOv5s. GAL-YOLOv5 has faster detection speed and higher



(a)

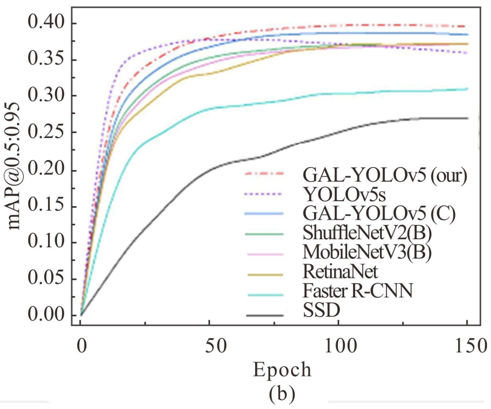


Fig.6 (a) Change curve of mAP@0.5 during model training; (b) Change curve of mAP@0.5: 0.95 during model training

detection accuracy while maintaining the same model volume as ShuffleNetV2(B) and MobileNetV3(B). By comparing GAL-YOLOv5 and GAL-YOLOv5(C), it is proved that the reasonable design and optimization of the model structure in this paper make it closer to the design requirements while obtaining optimal performance.

In order to verify the generality of the model, first train GAL-YOLOv5, faster R-CNN, SSD, RetinaNet, and YOLOv5s on the self-built dataset, and then use the two public flame and smoke datasets (Dataset 1, Dataset 2) proposed by @AbimbolaOO[30] and @gengyanlei[31] to test the above model. The results are shown in Tab.3.

Tab.3 Test results under two public datasets

Method	Dataset 1		Dataset 2	
	mAP@0.5	FPS	mAP@0.5	FPS
RetinaNet	0.732	56	0.750	50
Faster R-CNN	0.718	29	0.719	29
SSD	0.611	75	0.680	62
YOLOv5s	0.596	98	0.690	94
GAL-YOLOv5	**0.737**	**101**	**0.803**	**97**

As can be seen from the table, GAL-YOLOv5 has the highest mAP@0.5 and FPS on both datasets compared to the other four classic object detection models, which proves that it has good generality and makes it more advantageous in actual detection tasks.

To sum up, GAL-YOLOv5 balances the relationship between detection accuracy, speed, and model size well and is currently a lightweight flame and smoke detection model with excellent comprehensive performance.

In order to observe the detection performance of GAL-YOLOv5 more directly, we selected some actual flame and smoke pictures for detection. For the more prominent targets in Fig.7, GAL-YOLOv5 accurately detects flames and smoke in various scenes.

For the suspected targets accompanied by environmental interference in Fig.8, GAL-YOLOv5 still has good detection performance. When detecting smoke, GAL-YOLOv5 was not affected by the clouds around the smoke, and completed the detection accurately. Even if there were clouds in the positioning frame, the model did not misjudge them as smoke. When detecting flames, GAL-YOLOv5 did not make a misjudgment, even though the light emitted by the bulb has very similar characteristics to the flame.

From the above results, it can be seen that thanks to the excellent global awareness ability of GAL-YOLOv5, even small-scale flames, smoke that is almost integrated with the background, and smoke and flames that are irregular in shape and cover each other can be accurately detected by GAL-YOLOv5. This makes it possible to give early warning in the early stages of fire, effectively suppressing its occurrence.

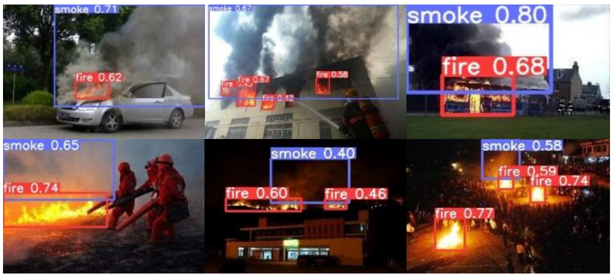


Fig.7 GAL-YOLOv5 detection results for prominent flame and smoke targets

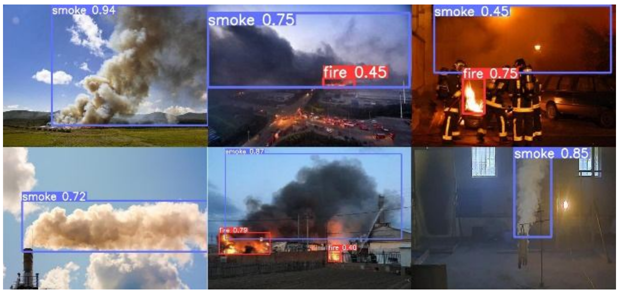


Fig.8 GAL-YOLOv5 detection results of suspected flame and smoke targets accompanied by environmental interference

Aiming at the problems existing in current CNN-based flame and smoke detection models, GAL-YOLOv5 is proposed. DGNet is proposed to greatly reduce the redundant information generated in the feature extraction process, compress the model's volume, and improve the model's ability to extract flame and smoke features. Then InvC3 is proposed to increase the model's global modeling ability for flame and smoke information in feature maps and further reduce the model's volume. Furthermore, FConv is proposed to expand the model's adaptive receptive field, enhance its ability to perceive complex foreground flame and smoke information in the feature map, and further optimize InvC3's global modeling capability. The model's global awareness ability of flame and smoke features was enhanced by the two. The experimental results show that compared with the benchmark model and other deep learning object detection models, GAL-

YOLOv5 shows good performance in terms of speed, accuracy, and model size. Meanwhile, the ablation experiments verify the effectiveness of the presented modules. It can be proved from the actual test results that the model proposed in this paper solves the flame-smoke detection problems better in a balanced way.

Ethics declarations

Conflicts of interest

The authors declare no conflict of interest.

References

[1] GAUR A, SINGH A, KUMAR A, et al. Video flame and smoke based fire detection algorithms：a literature review[J]. Fire technology, 2020, 56：1943-1980.

[2] FOGGIA P, SAGGESE A, VENTO M. Real-time fire detection for video-surveillance applications using a combination of experts based on color, shape, and motion[J]. IEEE transactions on circuits and systems for video technology, 2015, 25(9)：1545-1556.

[3] WANG T, BU L, YANG Z, et al. A new fire detection method using a multi-expert system based on color dispersion, similarity and centroid motion in indoor environment[J]. IEEE/CAA journal of automatica sinica, 2019, 7(1)：263-275.

[4] XIONG W. Research on fire detection and image infor mation processing system based on image processing[C]//2020 International Conference on Advance in Ambient Computing and Intelligence (ICAACI), September 12-13, 2020, Ottawa, ON, Canada. New York：IEEE, 2020：106-109.

[5] LI C, CAO Y, PENG Y. Research on automatic driving target detection based on YOLOv5s[C]//Journal of Physics：Conference Series 2022, November 12-14, 2021, Beihai, Guangxi, China. Bristol：IOP Publishing, 2022, 2171(1)：012047.

[6] LI S, LI L. DRT-Unet：a segmentation network for aiding brain tumor diagnosis[J]. Security & communication networks, 2022.

[7] REN S, HE K, GIRSHICK R, et al. Faster R-CNN：towards real-time object detection with region proposal networks[J]. Advances in neural information processing systems, 2015, 28.

[8] LIU W, ANGUELOV D, ERHAN D, et al. SSD：single shot multibox detector[C]//2016 European Conference on Computer Vision (ECCV), October 11-14, 2016, Amsterdam, Netherlands. Cham：Springer International Publishing, 2016：21-37.

[9] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once：unified, real-time object detection[C]//Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition, June 26-July 1, 2016, Las Vegas, NV, USA. New York：IEEE, 2016：779-788.

[10] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[C]//Proceedings of 2017 IEEE International Conference on Computer Vision, October 22-29, 2017, Venice, Italy. New York：IEEE, 2017：2980-2988.

[11] LIN T Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C]//Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, Hawaii, USA. New York：IEEE, 2017：2117-2125.

[12] LIU S, QI L, QIN H, et al. Path aggregation network for instance segmentation[C]//Proceedings of 2018 IEEE Conference on Computer Vision and Pattern Recognition, June 18-22, 2018, Salt Lake City, Utah, USA. New York：IEEE, 2018：8759-8768.

[13] REDMON J, FARHADI A. YOLO9000：better, faster, stronger[C]//Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, Hawaii, USA. New York：IEEE, 2017：7263-7271.

[14] REDMON J, FARHADI A. YOLOv3：an incremental improvement[EB/OL]. (2018-04-08) [2023-01-14]. http：//arxiv.org/abs/1804.02767.

[15] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4：optimal speed and accuracy of object detection[EB/OL]. (2020-05-28) [2023-01-14]. http：//arxiv.org/abs/2004.10934.

[16] CHAOXIA C, SHANG W, ZHANG F. Information-guided flame detection based on faster R-CNN[J]. IEEE access, 2020, 8：58923-58932.

[17] SHI L, ZHANG H F, YANG J F. Video-based fire and smoke detection based on improved SSD[J]. Computer applications and software, 2021, 38(12)：161-167. (in Chinese)

[18] HU Y, ZHAN J, ZHOU G, et al. Fast forest fire smoke detection using MVMNet[J]. Knowledge-based systems, 2022, 241：108219.

[19] CAI W, WANG C, HUANG H, et al. A real-time smoke detection model based on YOLO-SMOKE algorithm[C]//2020 Cross Strait Radio Science & Wireless Technology Conference (CSRSWTC), December 17-18, 2020, Fuzhou, China. New York：IEEE, 2020：1-3.

[20] ZHANG S G, ZHANG F, DING Y, et al. Swin-YOLOv5：research and application of fire and smoke detection algorithm based on YOLOv5[J]. Computational intelligence and neuroscience, 2022.

[21] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[22] HAN K, WANG Y, TIAN Q, et al. Ghostnet：more features from cheap operations[C]//Proceedings of 2020 IEEE Conference on Computer Vision and Pattern Recognition, June 14-19, 2020, Virtual. New York：IEEE, 2020：1580-1589.

[23] HOWARD A G, ZHU M, CHEN B, et al. Mobilenets：efficient convolutional neural networks for mobile vision applications[EB/OL]. (2017-06-18) [2023-01-14]. http：//arxiv.org/abs/1704.04861.

[24] LI D, HU J, WANG C, et al. Involution：inverting the inherence of convolution for visual recognition[C]//Proceedings of 2021 IEEE Conference on Computer Vision and Pattern Recognition, June 19-25, 2021, Virtual. New York：IEEE, 2021：12321-12330.

[25] AVENASH R, VISWANATH P. Semantic segmentation of satellite images using a modified CNN with hard-swish activation function[C]//Proceedings of 2019 the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, February 25-27, 2019, Prague, Czech Republic. Seattle：Semantic Scholar, 2019：413-420.

[26] MA N, ZHANG X, SUN J. Funnel activation for visual recognition[C]//2020 European Conference on Computer Vision (ECCV), August 23-28, 2020, Glasgow, UK. Cham：Springer International Publishing, 2020：351-368.

[27] GLOROT X, BORDES A, BENGIO Y. Deep sparse rectifier neural networks[C]//Proceedings of 2011 the 14th International Conference on Artificial Intelligence and Statistics, April 11-13, 2011, Ft. Lauderdale, USA. Cambridge：JMLR, 2011：315-323.

[28] MA N, ZHANG X, ZHENG H T, et al. Shufflenet v2：practical guidelines for efficient CNN architecture design[C]//2018 European Conference on Computer Vision (ECCV), September 8-14, 2018, Munich, Germany. Cham：Springer International Publishing, 2018：116-131.

[29] HOWARD A, SANDLER M, CHU G, et al. Searching for mobilenetv3[C]//Proceedings of 2019 IEEE/CVF International Conference on Computer Vision, October 27-November 2, 2019, Seoul, Korea. New York：IEEE/CVF, 2019：1314-1324.

[30] Fire-flame-dataset：version 1.0[EB/OL]. (2019-06-28) [2023-01-14]. https：//github.com/DeepQuestAI/Fire-Smoke-Dataset.

[31] Fire-smoke-detect-YOLOv4-v5 and fire-smoke-detect-dataset：version 1.0[EB/OL]. (2022-12-03) [2023-01-14]. https：//github.com/gengyanlei/fire-smoke-detect-yolov4.