FSD-Net: Multi-Scale Outdoor Fire and Smoke Detection via Swin Transformer-Enhanced YOLOv11s with Lightweight Feature Fusion and Inner-CIoU Loss

Yuanpan Zheng[1*], Yu Zhang[1], Chao Wang[1], Xuhang Liu[1], Wenbin Zhong[1]
[1]School of Computer Science and Artificial Intelligence, Zhengzhou University of Light Industry, Zhengzhou 450000, China
E-mail: ypzheng@zzuli.edu.cn

Keywords: Fire smoke detection, YOLOv11s, multi-scale feature extraction, feature fusion, attention mechanism, lightweight network

Received: March 17, 2026

In this paper, a multi-scale fire and smoke detection model, named FSD-Net, is established upon the YOLOv11s model to overcome the challenges in outdoor fire scenarios, such as large variations in target shapes, complicated backgrounds, and the difficulty of detecting small objects. First, a multi-scale feature extraction module, C3Swin, is designed by integrating the Swin Transformer into the backbone network. This strengthens the model's capability to capture fine-grained fire and smoke features. Second, we incorporate shallow features into the FPN-PAN architecture to improve small-object detection performance. Third, we developed a lightweight fusion module, GS-A2C2f, to enhance the model's feature fusion capabilities.. Furthermore, GhostConv is employed to decrease computational complexity. An auxiliary bounding-box loss function, Inner-CIoU, is introduced to better balance detection performance and inference speed. Experimental results on a self-built dataset comprising 11,672 images evaluated following the COCO metrics, suggest that, compared to baseline model, FSD-Net achieves improvements of 2.5%, 2.3%, and 1.1% in precision, recall, mAP@50 and mAP@95, respectively, while lessening the number of parameters by 1.3 million and increasing FPS by 3.8%. These results confirm that FSD-Net achieves accurate fire smoke detection and localization, providing an effective solution for outdoor fire detection in complex environments.

Povzetek: Predstavljen je model za zaznavanje ciljev za zgodnje opozarjanje na požarni dim na prostem. Model temelji na izboljšanem modelu YOLOv11s in omogoča natančnejše ter učinkovitejše prepoznavanje in lokalizacijo požarnega dima v kompleksnih okoljih.

1 Introduction

Fire has consistently threatened social safety and economic security. In 2025, the wildfires in Los Angeles, California, USA, resulted in economic losses of tens of billions of dollars[1]. Outdoor fires generally feature high intensity and rapid spread, frequently bringing about devastating consequences. Therefore, effective fire and smoke detection in outdoor environments is of great significance for early warning.

Although traditional fire smoke detection methods have been widely applied, they rely heavily on manual feature derivation and handcrafted feature design, leading to high expenses and restricted flexibility in complex environments[2]. In recent years, deep learning-based fire smoke detection methods have become a useful solution, attributed to their capability of automatic feature extraction. These approaches are typically classified into two-stage object detection methods, one-stage object detection methods and Transformer-based detection methods.

Two-stage detection methods, represented by Faster R-CNN[3] and Mask R-CNN[4], require region proposals followed by feature extraction. While they achieve high detection accuracy, their detection speed is relatively slow. In contrast, one-stage detection methods, represented by YOLO[5], can locate detection targets and simultaneously determine the detection area, providing a favorable equilibrium between accuracy and real-time performance. Thus, they are suited for fire smoke detection operations. Xiao et al.[6] improved YOLOv5 by introducing depth-wise separable convolution, Ghost convolution (GhostConv), and adaptive receptive field convolution (FConv) to enhance feature representation while minimizing model complexity, even though the recall rate decreased. Chen et al.[7] introduced a bidirectional feature pyramid network (BiFPN) into YOLOv8 and combined it with a lightweight mixed local attention mechanism (MLCA) to improve the model's capability of extracting small smoke features. Nonetheless, the model's performance remained limited in complex scenarios. Zhao et al. [8] proposed a dual-path residual attention mechanism in YOLOv11 and employed a detection head incorporating attention mechanisms to compensate for feature loss in smoke-obscured regions. The YOLO framework, despite its high efficiency, still has some limitations in extracting multi-scale features, such as insufficient sensitivity to small objects [9].

In recent years, vision Transformers have demonstrated strong performance in computer vision tasks[10] and have been widely applied in fire smoke detection research (Zheng et al. [11], Sun et al. [12], and Wang et al. [13]) However, these methods rely on attention mechanisms and hence typically involve high computational costs.

Regardless of the rapid advancement of related research, fire smoke detection in complex outdoor environments still faces numerous challenges, such as complex backgrounds, interfering objects, and difficulties in detecting small targets. In recent years, multi-scale features have effectively enhanced a model's adaptability to complex environments. For example, Qureshi et al. [14] demonstrated that a multi-scale convolutional neural network feature extraction strategy tailored to variations in Region of Interest (RoI) scale improves the efficiency of classification models in multi-scale scenarios with high effectiveness. Yin et al. [15] incorporated dilation convolutions with varying dilation ratios into the YOLOv5 backbone network, contributing to the enhanced multi-scale feature capture capability of the C3 module. Li et al. [16] integrated their self-developed Multi-Scale Convolutional Attention (MSCA) mechanism into the neck network of YOLOv11 and strengthened the model's ability to capture multi-scale spatial features using multi-branch deep convolutions. While these studies have yielded encouraging results, they largely focus on a single dimension: either enhancing the model's multi-scale feature extraction capabilities or improving its multi-scale feature fusion capabilities.

In this study, the challenges in outdoor fire detection, such as complex environmental interference and the difficulty of detecting small targets are addressed by using YOLOv11s as the baseline model and proposing a fire and smoke detection algorithm combining multi-scale feature extraction and fusion. This approach aims to improve detection accuracy while lowering model size and computational costs. Table 1 presents a comparison of the aforementioned studies.

Table 1: Summary of existing studies on fire smoke recognition and localization

Author/Year	Model Used	Datasets	mAP@ 50(%)	Precision (%)	Recall (%)	param (M)	Main Weaknesses
Xiao et al. (2023)[6]	YOLOv5s+ Involution Operator	15,944 images, flame and smoke	70.8	74.3	62.6	3.56	The effectiveness of the testing has not been validated for small-scale targets.
Chen et al. (2025)[7]	YOLOv8n+BiFP N+MLCA	11,130 images forest fire and smoke	80.1	N/A	N/A	2.0	Detection capabilities are limited in more complex scenarios such as foggy conditions or when objects are obstructed.
Zhao et al. (2025) [8]	YOLOv11+C3k2-DWR+SEAMHea d+W-SIoU	10,641 images, flame and smoke	69.2	77.9	64.8	14	The model has a large number of parameters; the effectiveness of the loss function has not been validated on a real dataset.
Li et al. (2025)[16]	YOLOv11s+Multi -Scale Convolutional Attention	21,527 images, flame and smoke	77.4	79.7	69.8	N/A	No number of arguments was provided; this may involve heavy computation.

2 Materials and methods

2.1 Architecture of proposed method

2.1.1 Baseline model

The YOLO series algorithms are representative one-stage object detection methods and have been widely applied in the field of fire smoke detection recently[17]. YOLOv11s[18], as the latest model released by Ultralytics, adopts a more lightweight framework while achieving improved detection performance.

The network structure of YOLOv11s is composed of the backbone network (Backbone), the neck network (Neck), and the detection head (Head). The backbone network enhances local feature extraction capability by introducing the C3k2 module. This module contains cross-layer connections and split operations while supporting customizable convolution kernel sizes, thereby facilitating flexible adaptation to different task scenarios. Additionally, the C2PSA module is incorporated to provide YOLOv11 with a strong spatial perception capability through a spatial attention mechanism[19]. The neck network adopts the Feature Pyramid Network (FPN)[20] and Path Aggregation Network (PAN) [21], referred to as the FPN-PAN structure, to achieve multi-level semantic information fusion. The head utilizes a decoupled detection head, which separates the original detection head into an object detection head and a classification head. Two depthwise separable convolutions (DSConv[22]) are integrated into the classification head to reduce the number of parameters and computational cost.

While YOLOv11s currently performs well in most o bject detection tasks, it encounters difficulties in outdoor fire smoke detection. Particularly, the model may produc e false detections and missed detections in outdoor scenes with complicated backgrounds and intertwined fire and s moke features. Given these issues, a multi-scale outdoor f ire detection model, named FSD-Net, is established in ou r study based on YOLOv11s. (In this paper, the YOLOv1 1s model is implemented based on the open-source reposi tory released by Ultralytics https://github.com/ultralytics/ ultralytics)

2.1.2 FSD-Net network

The architecture of FSD-Net is illustrated in Figure 1. In the backbone, a multi-scale feature extraction module,

named C3Swin, is proposed. The module augments the model's multi-scale feature extraction capability by introducing the Swin Transformer and combining it with convolutional channel improvements. Furthermore, the shallow feature layer P2 is integrated into the FPN-PAN structure to strengthen the model's ability to detect small-scale targets.

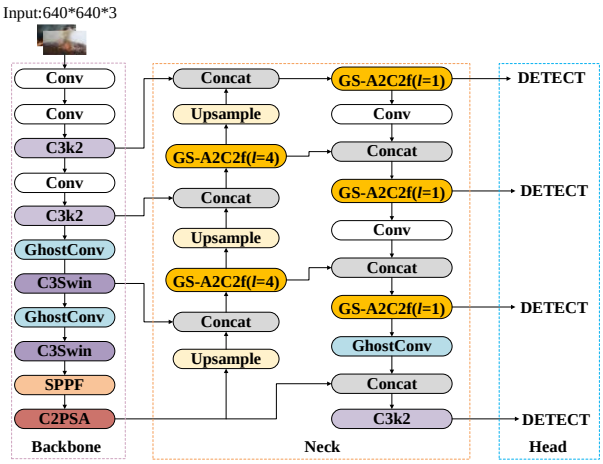


Figure 1: Structure of FSD-Net

In the neck, a lightweight feature fusion module, named GS-A2C2f, is constructed to improve the effectiveness of feature fusion. A GhostConv replacement strategy is designed to replace part of the standard convolution operations, allowing for the optimization of model parameters and achievement of a balance between detection effectiveness and inference speed. Furthermore, the loss function Complete Intersection over Union (CIoU), which measures the overlap between predicted and true bounding boxes, is replaced with Inner-CIoU, an adaptation incorporating information from an efficient auxiliary bounding box. This modification alleviates gradient attenuation (slowed learning) within regions with high intersection-over-union values while improving the stability of the model during training.

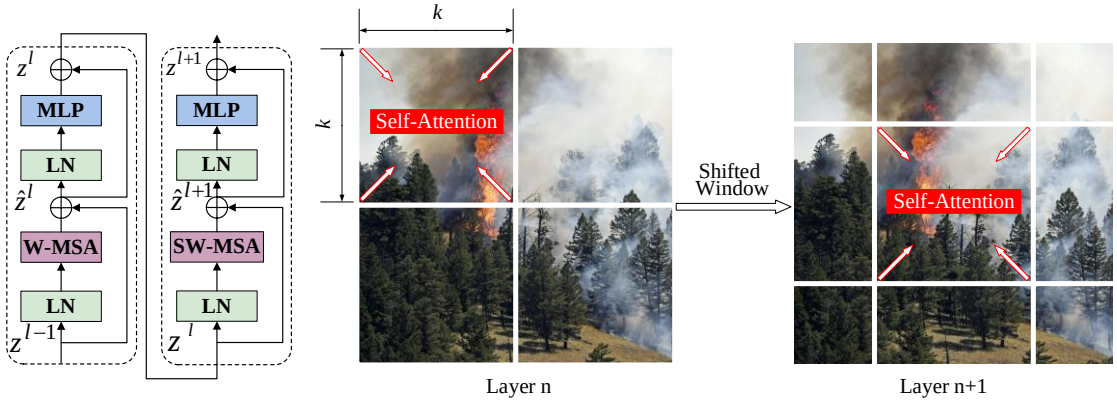
The structure of the proposed modules will be detailed in following sections.

2.2 Core improvements of FSD-Net

2.2.1 C3Swin module

The C3k2 module in YOLOv11s introduces a variable convolution structure, C3k, to improve the flexibility of the network. Nonetheless, the contextual information modeling capability provided by variable convolution remains limited. Through the Shifted Window Multi-head Self-Attention mechanism (Shifted Window Multi-head Self-Attention, (S)W-MSA), Swin Transformer[23] retains richer contextual information with relatively low computational complexity. In recent years, it has been widely applied in fire smoke detection tasks and has demonstrated strong performance[13,24,25]. Inspired by this approach, a Swin Transformer layer is introduced into C3k in this study to improve the channel convolution in C3k2, enabling the design of the C3Swin module.

Figure 2(a) illustrates the specific structure of the Swin Transformer layer used in this study, where LN and MLP denote the normalization layer and the multilayer perceptron, respectively. It is adopted to enhance nonlinear feature transformation capability. W-MSA and SW-MSA represent the window-based self-attention mechanism and the shifted window self-attention mechanism, respectively. The two are connected through residual connections, collectively referred to as (S)W-MSA. The variables z^{l-1} embodies the input features in Swin Stage; $(\hat{z}^{l}, \hat{z}^{l+1})$ and (z^{l}, z^{l+1}) signify the output features of (S)W-MSA and MLP, respectively.



(a) Structure of Swin Stage.

(b) Feature interaction process of the Shifted Window Multi-head Self Attention across adjacent layers

Figure 2 : (a) Schematic illustration of Swin Stage structure; (b) Shifted Window Multi-head Self Attention.

In this paper, structure in Figure 2(a) is referred to as a Swin Stage. The computation process of a feature map within a Swin Stage is expressed as:

$$\hat{z} = \text{W-MSA}(\text{LN}(z^{l-1})) + z^{l-1} \quad (1)$$

$$z^{l} = \text{MLP}(\text{LN}(\hat{z})) + \hat{z} \quad (2)$$

$$\hat{z}^{l+1} = \text{SW-MSA}(\text{LN}(z^{l})) + z^{l} \quad (3)$$

$$z^{l+1} = \text{MLP}(\text{LN}(\hat{z}^{l+1})) + \hat{z}^{l+1} \quad (4)$$

The process by which the n-th layer feature map passes through a Swin Stage is depicted in Figure 2(b). The feature map is first divided into four equal windows in the W-MSA layer, and self-attention is computed

independently within each window. In the next layer, SW-MSA performs a complete shift of the window partition to generate nine new irregular windows. When self-attention is computed again, each new window obtains more information from the four windows of the n-th layer. This design establishes information interaction among windows, enabling the module to capture richer contextual information within a relatively small receptive field.

The structure of C3Swin is displayed in Figure: 3. A new module named C3ST is formed by replacing the bottleneck layer (Bottleneck) of C3k with a Swin Stage. In the original version of the C3k module, the parameter k controls the variable convolution kernel size (*Kernel Size=k*). In C3ST, k is mapped to the attention receptive field size of (S)W-MSA (*Window Size=k*), allowing the W-MSA and SW-MSA window sizes to be directly controlled through the base parameter of C3k. The number of attention heads for the (S)W-MSA is set to 4, the attention window size to 7, and the depth to 2.

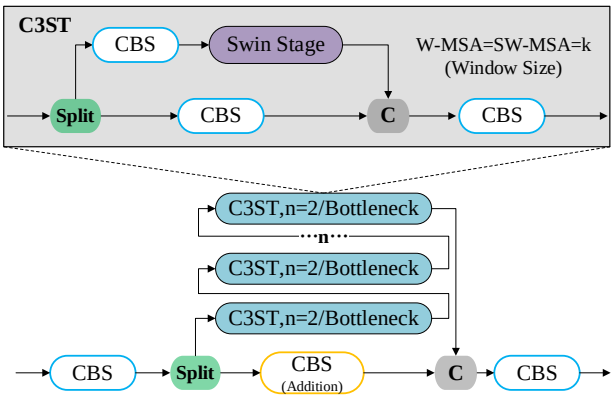


Figure: 3 C3Swin module

The introduction of the Swin Stage enables the channel containing C3ST to extract features with richer abstract information, leading to a large semantic gap compared with the split features in the other branch. For the purpose of avoiding overfitting and information redundancy during feature concatenation, a 1×1 convolution layer is added to the branch channel that does not pass through C3ST. In this way, the channel features are aligned and remapped to improve the quality of the concatenated features. Additionally, the switching mechanism between the C3ST module and Bottleneck is retained in the C3Swin module to accommodate scenarios where partial detection performance may be sacrificed in exchange for higher detection speed and reduced model capacity.

To sum up, the C3Swin modules significantly improve multi-scale feature extraction capability by introducing the Swin Transformer while maintaining controllable model parameters and computational complexity. Therefore, the can be efficiently applied to fire smoke detection tasks in complex outdoor environments.

2.2.2 Shallow feature fusion

Targets in outdoor fire smoke scenarios commonly feature irregular shapes and large variations in scale. Shallow feature information plays a crucial function in preserving semantic details and spatial information of fire and smoke, especially for small targets[26,27]. In YOLOv11s, the P2 layer is the shallow feature layer with the highest resolution and the smallest receptive field. Therefore, the P2 layer is introduced into the Feature Pyramid Network–Path Aggregation Network (FPN-PAN) structure to expand the feature fusion scale of the model and improve the detection capability for small targets.

Figure: 4 presents the FPN-PAN structure after the introduction of the P2 layer. In the FPN stage, P2 participates in upsampling and multi-scale feature fusion, generating a new feature map P19, which is subsequently connected to the newly added small-object detection head. In the PAN stage, P19 is further fused with higher-level features through a bottom-up aggregation process such that shallow features effectively participate in the generation of the final detection results.

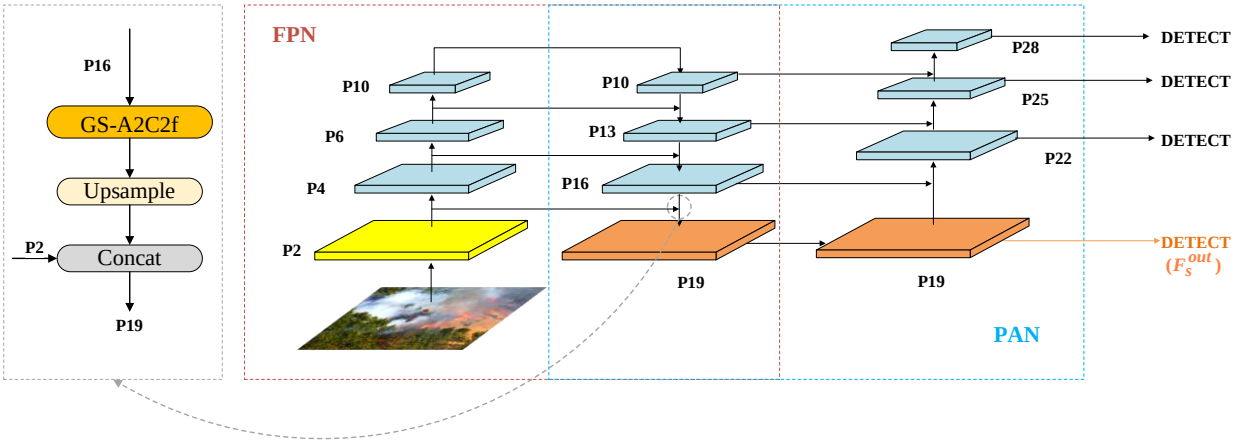


Figure: 4 Schematic diagram of FPN-PAN with P2 Layer integration

Consequently, the model can learn richer small-object features during the training process. The output formulas of the newly added detection head F_{19}^{out} is:

$$F_{19}^{out} = GSA2(Concat[\alpha \cdot F_{16}^{Up}, \beta \cdot P_2]) \quad (5)$$

$$F_{16}^{Up} = Upsample(P_{16}, s=2) \quad (6)$$

where *Concat* denotes the feature concatenation operation; *Upsample* signifies the upsampling operation. P_2 represents the shallow feature from the P2 layer; F_{16}^{Up} embodies the upsampled feature from the P16 layer; s refers to the scale parameter; α and β indicate the normalized weights of the two features, respectively; GSA2 stands for the lightweight feature fusion module GS-A2C2f proposed in this paper, which will be detailed in Section 2.2.3.

2.2.3 GS-A2C2f Module

Based on Section 2.2.2, a lightweight feature extraction module, GS-A2C2f, is constructed by building upon the A2C2f module proposed in YOLOv12[28]. By integrating the Area Attention mechanism (AAttn) and a module lightweighting strategy, GS-A2C2f effectively establishes cross-scale semantic correlations of targets in spatial space and improves the suppression of non-target regions while attaining efficient feature fusion. Its structure is displayed in Figure: 5.

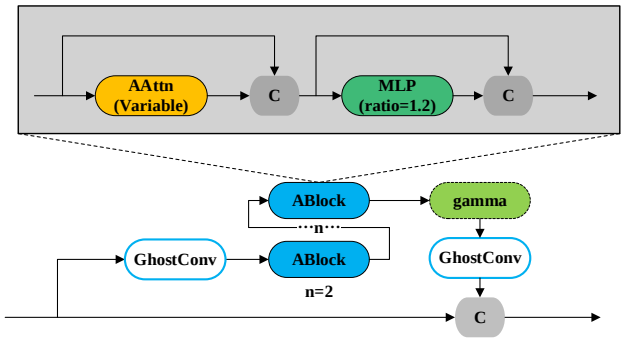


Figure: 5 GS-A2C2f module

The attention window division of AAttn is illustrated in Figure: 6. For each operation, a feature map of resolution (H, W) is divided into l segments of size (H/l, W) or (H, W/l), and self-attention is computed within each segment. The parameter l is typically set to 4 to ensure a global receptive field. Under this configuration, the computational cost of AAttn is reduced from n^2hd to $1/4n^2hd$, where h denotes the number of attention heads, d embodies the channel dimension per head, and $n = H \times W$ signifies the number of pixels in the feature map. This strategy eliminates explicit window partitioning and substantially reduces the computation of the attention mechanism while retaining global perceptual capability.

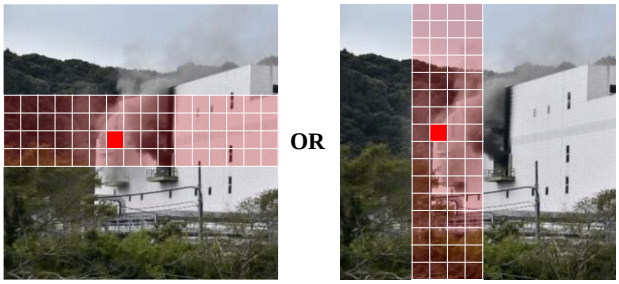


Figure: 6 Area Attention Window partitioning diagram

Since directly introducing A2C2f into the expanded network of Section 2.2.2 would still incur considerable computational cost, a lightweight version of the module, GS-A2C2f (Ghost-A2C2f), is proposed in this study.

The improvements of GS-A2C2f lie in the following aspects:

GhostConv Replacement: Two standard convolution layers in the module are replaced with GhostConv to lessen memory consumption during backpropagation. As revealed in Figure: 7, GhostConv[29] decomposes a conventional convolution into two steps. First, a standard convolution generates k feature maps. Next, additional feature maps are generated by applying inexpensive linear transformations to each channel. Finally, the two sets of feature maps are concatenated to produce the output feature map. This design lowers both the number of parameters and computation while maintaining feature representation capability.

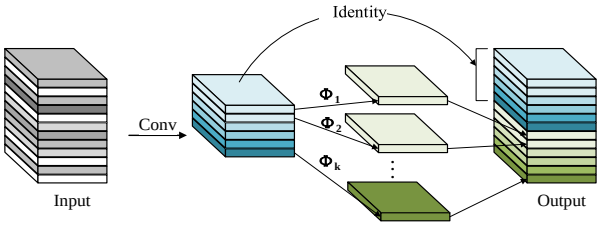


Figure: 7 GhostConv

MLP Ratio Adjustment: The multilayer perceptron (MLP) in the ABlock is a major source of model intricacy. The MLP ratio directly determines the number of hidden channels, consequently influencing the overall number of parameters and computational overhead. A single-layer MLP's parameters in ABlock are calculated by:

$$Params_{MLP} = C_{in} \times C_{out} \quad (7)$$

$$C_{in} = dim, C_{out} = r \times dim \quad (8)$$

where C_{in} and C_{out} represent the input and output channels, respectively; dim denotes the hidden dimension; r signifies the MLP ratio. In this paper, the MLP ratio of A2C2f is reduced from the default value of 2.0 to 1.2, leading to an approximately 40% decrease in computational parameters.

AAttn Window Division Strategy: The division strategy of AAttn windows in different layers of GS-A2C2f is adjusted to balance computational effectiveness and feature representation capability. As illustrated in Figure 1, the window division of AAttn is set to 4 for mid-level high-resolution feature layers such as P13 and P16 to preserve area attention; the window division is set to 1 for deep low-resolution layers such as P19, P22, and P25, suggesting that no window partitioning is applied, and global attention is computed.

2.3 Other improvements

2.3.1 GhostConv replacement strategy

The improvements described above yield a moderate increase in the overall number of parameters in the model.

For the purpose of keeping the model size within a reasonable range, GhostConv is further employed to replace part of the standard convolution operations in the network. Nevertheless, the linear transformation of GhostConv may restrict its ability to represent high-density features, bringing about a decline in model accuracy. Given this issue, a GhostConv replacement strategy is designed in this study to evaluate the results of replacing GhostConv in different convolutional layers of the model. The experimental results are lsited in Table 2. The number of layers we ultimately selected for replacement is shown in Figure: 8.

Table 2: GhostConv replacement position comparison experiment

GhostConv Replacement Scheme	mAP@50(%)	Param(M)	FPS
Non	79.5	9.3	80.1
All	79.0	**7.8**	80.2
Backbone only	**79.5**	8.2	79.0
Neck only	79.3	8.8	76.6
Layers 20 and 23	79.1	9.1	82.6
Layers 5, 7, and 26	79.3	8.1	**85.2**

The comparative experiments suggest that replacing all standard convolutions with GhostConv contributed to a significant reduction in model parameters but also led to a substantial loss in detection accuracy. When only the basic convolutions in the backbone network were replaced, there was no loss in model accuracy, whereas the FPS dropped. When only the convolutions in the neck network were replaced, the model suffered a 0.2-point loss in accuracy, while the model parameters remained relatively high, and the FPS further declined.

Finally, we attempted to replace some of the high-level feature extraction layers and the high-channel feature fusion layers (namely, the standard convolutions in layers 5 and 7 of the backbone network and layer 26 of the neck network). At this point, GhostConv yielded the most significant improvement to the model, and the 0.2% accuracy loss was within an acceptable range. Meanwhile, the number of parameters was reduced by 1.1 million, and FPS was improved by 7%. In contrast, replacing only the narrow-channel convolutions in layers 20 and 23 brought about negligible improvements.

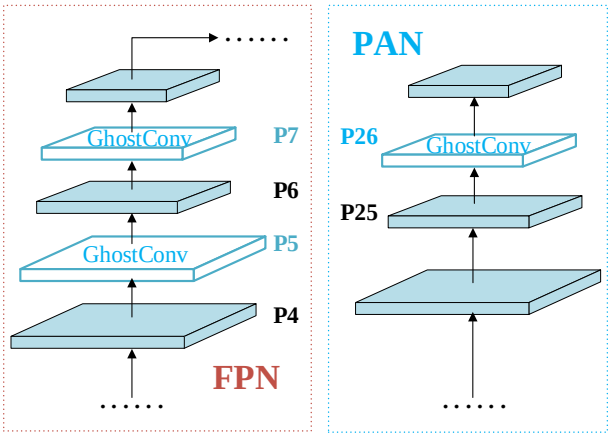


Figure: 8 GhostConv replacement layer

Overall, the model achieved the best overall performance when GhostConv was used to replace some of the standard convolutions (layers 5, 7, and 26) only in the high-level feature extraction layer and the high-channel feature fusion layer. Therefore, this replacement strategy was employed in our study.

2.3.2 Inner-CIoU

YOLOv11s employs Complete Intersection over Union (CIoU)[30] as the default loss function. CIoU can accurately measure the similarity between two bounding boxes. However, CIoU has the following limitations in the fire smoke detection process. When the width–height ratios of the predicted box and the confidence box are linearly correlated, the predicted box may considerably exceed the target box without being strongly constrained. This restricts the regression optimization process. Additionally, the gradient of CIoU tends to decay in zones with high intersection-over-union (IoU), leading to a clear decrease in the convergence speed. Concerning these issues, CIoU is replaced with the Inner-CIoU loss function based on an efficient auxiliary bounding box in this study. This function introduces the auxiliary bounding box algorithm of Inner-IoU[31] into CIoU to effectively improve the boundary regression accuracy of the model, expressed as:

$$b_l^{gt} = x_c^{gt} - \frac{w^{gt} * ratio}{2}, b_r^{gt} = x_c^{gt} + \frac{w^{gt} * ratio}{2} \quad (9)$$

$$b_t^{gt} = y_c^{gt} - \frac{h^{gt} * ratio}{2}, b_b^{gt} = y_c^{gt} + \frac{h^{gt} * ratio}{2} \quad (10)$$

$$b_l = x_c - \frac{w * ratio}{2}, b_r = x_c + \frac{w * ratio}{2} \quad (11)$$

$$b_t = y_c - \frac{h * ratio}{2}, b_b = y_c + \frac{h * ratio}{2} \quad (12)$$

$$Inter = (\min(b_r^{gt}, b_r) - \max(b_l^{gt}, b_l)) * (\min(b_b^{gt}, b_b) - \max(b_t^{gt}, b_t)) \quad (13)$$

$$Union = (w^{gt} * h^{gt}) * (ratio)^2 + (w * h) * (ratio)^2 - Inter \quad (14)$$

$$Inner\text{-}IoU = \frac{|B_{pred} \cap B_{gt}|}{|B_{pred}|} = \frac{Inter}{Union} \quad (15)$$

where and denote the ground truth bounding box (GT Box) and the anchor box (Anchor Box), respectively; (x_c^{gt}, y_c^{gt}) represent the center coordinates of the GT Box and its internal auxiliary box; (x_c, y_c) refers to the center coordinates of the Anchor Box and its internal auxiliary box; and embody the width and height of the GT Box, respectively; w and h signify the width and height of the Anchor Box, respectively. The parameter ratio is a scaling factor used to control the size of the auxiliary bounding box. Specifically, $ratio < 1$ indicates that the auxiliary bounding box is smaller than the actual bounding box. In this case, the effective regression range is smaller than that of the IoU loss, whereas the absolute gradient value is larger, which helps accelerate the convergence of samples

with high IoU. Besides, *ratio* > 1 reflects that the enlarged auxiliary bounding box expands the effective regression range and provides better guidance for bounding box regression in low-IoU cases.

Applying Inner-IoU to the CIoU loss function results in the Inner-CIoU loss, whose calculation method is expressed as:

$$Inner\text{-}CIoU = CIoU - Inner\text{-}IoU + IoU \tag{16}$$

To summarize, Inner-CIoU with the auxiliary bounding box retains the geometric constraint advantages of CIoU while effectively suppressing the gradient attenuation of CIoU in high-overlap regions. This accelerates the convergence of the loss function and improves the overall training performance of the model.

3 Experiments and analysis

3.1 Dataset

Existing public fire smoke datasets suffer from low-quality image clarity, missing annotations, or annotation deviations. For the purpose of overcoming the challenges of fire smoke detection in outdoor scenes, a dedicated outdoor fire smoke image dataset is constructed in this study. Figure: 9 exhibits some sample images from the dataset. The dataset contains 11,672 outdoor fire smoke images with different illumination levels, background colors, shapes, and sizes, and smoke densities, covering multiple outdoor fire scenarios, including wildfires, forest fires, remote-sensing fires, and urban fires. It yields a total of 5,858 images of smoke alone, 1,164 images of flames alone, and 4,641 images consisting of both flames and smoke. Among them, 80% of the images are derived from the D-Fire Dataset[32], and the remaining 20% were collected from the Internet. Search keywords include "Los Angeles fires," "outdoor fires," and "fire smoke." The dataset is divided into training set, validation set, and test set in an 8:1:1 ratio. It is annotated by the LabelImg tool and formatted according to the YOLO standard. Unlike the CoCo dataset format, the YOLO format typically separates data into image files and corresponding annotation text files. Furthermore, the dataset was preprocessed before training by the 'check_imgsz()' method in the Ultralytics framework to standardize the resolution of the preprocessed images to 640x640.

Figure: 9 Examples of the fire smoke dataset

3.2 Experimental environment

The experimental environment configuration used in this study is presented in Table 3.

Table 3: Configuration of experimental environment

Environment	Specification
Programming Language	Python V 3.8
Deep learning framework	Pytorch V 2.3.1
CPU	Intel(R) Xeon(R) Gold 6330
GPU	RTX 3090(24GB)
RAM	16 GB

The number of training epochs was set to 300, the batch size was 16, and the optimizer was Adam. The initial learning rate (lr0) was set to 0.001, and input image size (imgsz) was 640 × 640. All experiments in this paper were conducted under the above configuration to ensure consistency.

3.3 Evaluation indicators

In this study, *Precision*, *Recall*, mean Average Precision (*mAP*), number of parameters (*Params*), and frames per second (*FPS*) are adopted as evaluation metrics

for the model. These metrics adhere to the COCO protocol standards, and their calculation methods are expressed as:

$$Precision = \frac{TP}{TP + FP} \tag{17}$$

$$Recall = \frac{TP}{TP + FN} \tag{18}$$

$$AP = \int_0^1 p(r)dr \tag{19}$$

$$mAP = \frac{1}{N}\sum AP_i \tag{20}$$

$$Params = C_O \times (k^2 \times C_I + 1) \tag{21}$$

$$FPS = \frac{1}{Time} \tag{22}$$

where *TP*(True Positives) represents samples correctly identified as targets; *FP*(False Positives) denotes non-target samples incorrectly identified as targets; *FN*(False Negatives) embodies the number of missed target samples; *N* signifies the number of target categories; *i* refers to the i-th category; *k* stands for the convolution kernel size; C_O and C_I indicate the output and input channel numbers of the convolution layer, respectively. *Time*(s) describes the time required for the model to process each image frame, measured in seconds. *Precision*(*P*) and *Recall*(*R*) specify the proportion of correctly detected samples among all detected samples and all ground-truth samples, respectively.

Additionally, *AP* denotes the average precision, reflecting the prediction performance for a single category; *mAP* represents the mean average precision across all predicted categories, suggesting the overall detection performance of the model. In the following sections, *mAP@50* and *mAP@95* embody the mean average precision at IoU thresholds of 0.50 and 0.95, respectively. *Params* refers to the total number of trainable parameters in the neural network, and *FPS* indicates the number of image frames processed per second by the model, revealing its real-time performance. These performance metrics are defined following the COCO Protocol.

3.4 Ablation experiments and analysis

In this section, ablation experiments are performed with the six evaluation indicators described above to evaluate the improvement strategies proposed in Section 2. The results are presented in Table 4, where A, B, C, D, and E represent the C3Swin module, the P2 layer, the GS-A2C2f module, the GhostConv replacement strategy, and Inner-CIoU, respectively.

Table 4: Ablation experiment results

Models	*Precision*(%)	*Recall*(%)	*mAP@50*(%)	*mAP@95*(%)	*Param*(M)	*FPS*
baseline	80.0	71.6	77.6	46.0	9.4	86.0
+A	79.4	72.2	78.6	46.7	9.7	81.4
+B	79.2	73.5	79.0	46.9	9.5	80.2
+C	80.2	72.0	78.4	46.5	8.6	**91.7**
+A+B	78.7	73.3	79.0	46.8	9.9	77.1
+A+B+C	77.8	73.6	79.5	47.0	9.3	80.1
+A+B+C+D	79.6	73.9	79.3	47.0	8.1	85.2
+A+B+C+D+E	**80.5**	**74.1**	**80.0**	**47.1**	**8.1**	89.3

As observed from Table 4, the C3Swin module, the P2 layer, and the GS-A2C2f module are first introduced sequentially, resulting in performance improvements at different levels. When all three modules were introduced into the network simultaneously, the model's precision decreased by 2.2%, while recall, mAP@50, and mAP@95 increased by 2%, 1.9%, and 1%, respectively. This stems from the enhancement of multi-scale features and the introduction of shallow features, which improve the model's sensitivity to fire smoke targets. However, the increase in model parameters brings about a reduction in FPS. After the GhostConv replacement strategy and the Inner-CIoU loss function were applied, the number of model parameters dropped by 1.3 million, and FPS increased by 3.3%. Precision was ultimately improved by 0.5% compared to the baseline model. Meanwhile, recall, mAP@50, and mAP@95 were improved by 0.5%, 1%, and 0.1%, respectively. These results suggest that the lightweight structural optimization and the improved regression loss function improved the stability of feature representation while efficiently suppressing false detections and localization deviations. Consequently, a better balance between detection capability and prediction correctness was achieved.

To sum up, the proposed improvement strategies efficiently optimize the model effectiveness from a global perspective and considerably enhance its real-world applicability.

3.5 Comparative experiments and analysis

3.5.1 Comparative experiments of feature fusion modules

Performance of A2C2f module, GS-A2C2f module, and the commonly used feature fusion module BiFPN on our dataset was compared to validate the effectiveness of the feature fusion module GS-A2C2f proposed in this paper. The experimental results are illustrated in Table 5.

As suggested in the table, when BiFPN was introduced, mAP@50 improved by 1%, model parameters remained unchanged, and FPS decreased by 3.5. Upon the

introduction of GS-A2C2f, although the improvement in mAP@50 was smaller than that of BiFPN (0.7%), the number of parameters was effectively lowered by 0.8M, and FPS increased by 5.7. Furthermore, GS-A2C2f performed better on all three metrics compared to the original A2C2f. Overall, the improvement strategy of GS-A2C2f effectively reduced model parameters while enhancing model performance, achieving a better balance between detection accuracy and computational efficiency.

Table 5: Comparison experiment of GS-A2C2f module

Model	mAP@50(%)	Param(M)	FPS
YOLOv11s	77.6	9.4	86.0
+BiFPN	**78.6**	9.4	82.5
+A2C2f	78.2	8.7	84.7
+GS-A2C2f	78.4	**8.6**	**91.7**

3.5.2 Comparative experiments of loss functions

Performance of Inner-CIoU was compared with that of IoU, CIoU, and Inner-IoU in fire smoke detection based on our self-built dataset to validate the advantages of Inner-CIoU. The mAP@50 curves for the four loss functions during training are exhibited in Figure: 10.

Regarding convergence speed, as the number of training iterations increased, particularly after 150 epochs, the model loss of Inner-CIoU gradually converged, exhibiting a smooth trend. In contrast, the convergence curves of other loss functions continued to fluctuate. This demonstrates that the Inner-CIoU improvement effectively enhanced the model's convergence capability. Concerning mAP@50 performance, Inner-CIoU achieved a significantly higher average mAP value throughout the training process, implying superior fire detection performance.

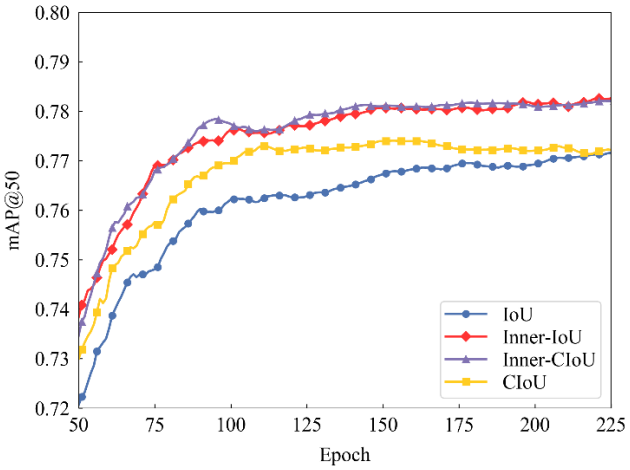


Figure: 10 Comparison of Inner-CIoU with other loss functions

3.5.3 Comparative experiments of different detection algorithms

In this section, comparative experiments were conducted between the proposed FSD-Net model and several mainstream object detection algorithms in recent years on the self-constructed fire smoke dataset to verify the suitability and application value of FSD-Net in fire smoke detection tasks. The detection results of different algorithms are detailed in Table 6.

Table 6: Comparative experimental results

Models	*Precision*(%)	*Recall*(%)	*mAP@50*(%)	*mAP@95*(%)	*Param*(M)	*FPS*
SSD	74.8	62.7	60.0	41.4	143	30.2
Faster-RCNN	74.3	65.1	69.5	41.6	60	20.7
RT-DETR	77.2	70.0	75.8	42.8	41.9	68.1
YOLOv5s	79.4	70.9	78.0	46.4	9.2	96.7
YOLOv8s	79.2	72.0	78.2	45.8	11.2	**105.6**
YOLOv9s	79.8	71.1	78.0	46.6	**7.3**	41.5
YOLOv10s	76.7	72.6	77.2	45.3	8.0	82.7
YOLOv11s(baseline)	80.0	71.6	77.6	46.0	9.4	86.0
FSD-Net	**80.5**	**74.1**	**80.0**	**47.1**	8.1	89.3

Apart from the baseline model YOLOv11s, the comparison models include SSD[33], Faster R-CNN, RT-DETR[34], YOLOv5[35], YOLOv8[36], YOLOv9[37], and YOLOv10[38]. Considering that the performance of the YOLO series models improves sequentially with model size (n, s, m, x), the s version was uniformly selected in our study to ensure a fair comparison of experimental results.

According to Table 6, SSD, Faster-RCNN, and RT-DETR fall short in accuracy and real-time performance for outdoor fire detection, even though they have larger network architectures compared to FSD-Net. From another perspective, models such as YOLOv5s and YOLOv8s demonstrate excellent efficiency in fire detection. However, their performance on mAP@50 is 2.0% and 1.6% lower, respectively, compared to our model, attributed to a lack of sensitivity to multi-scale fire smoke features and small objects.

While the FSD-Net model did not achieve optimal results in terms of parameter count and FPS, it outperformed the baseline model on all six evaluation metrics. It also demonstrated the best performance in precision, recall, mAP@50, and mAP@95. In other words, the model can detect fire smoke targets of varying sizes with greater accuracy and stability, offering broad application potential.Furthermore, the mAP@50 and

mAP@95 curves for FSD-Net and other models during training are illustrated in Figure 11.

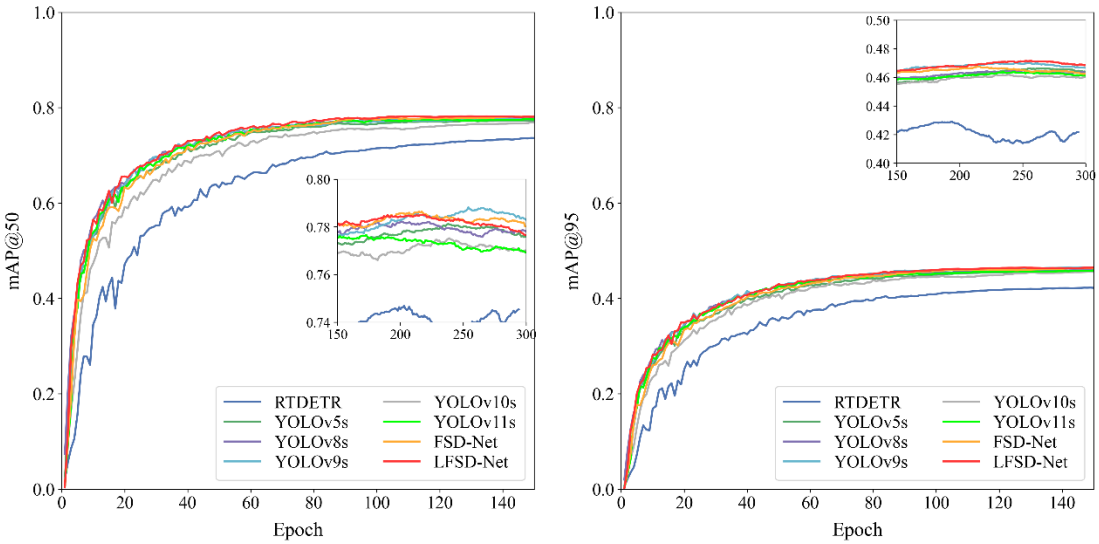


Figure 11: Comparison of FSD-Net's mAP against other models

Compared to other models, FSD-Net achieved faster convergence on the mAP@50 and mAP@95 curves during the first 150 epochs of training. During the late training phase (150~300 epochs) when model performance stabilized, FSD-Net continued to exhibit superior performance on both the mAP@50 and mAP@95 curves, with smoother trends. This result validates the effectiveness of the improvements achieved in this paper.

Four performance curves for FSD-Net (namely, the precision curve, recall curve, F1 curve, and Pr curve) are plotted in Figure 12 to verify the model's stability under different confidence thresholds. The sampling ranges for confidence, precision, and recall are all 0~1.

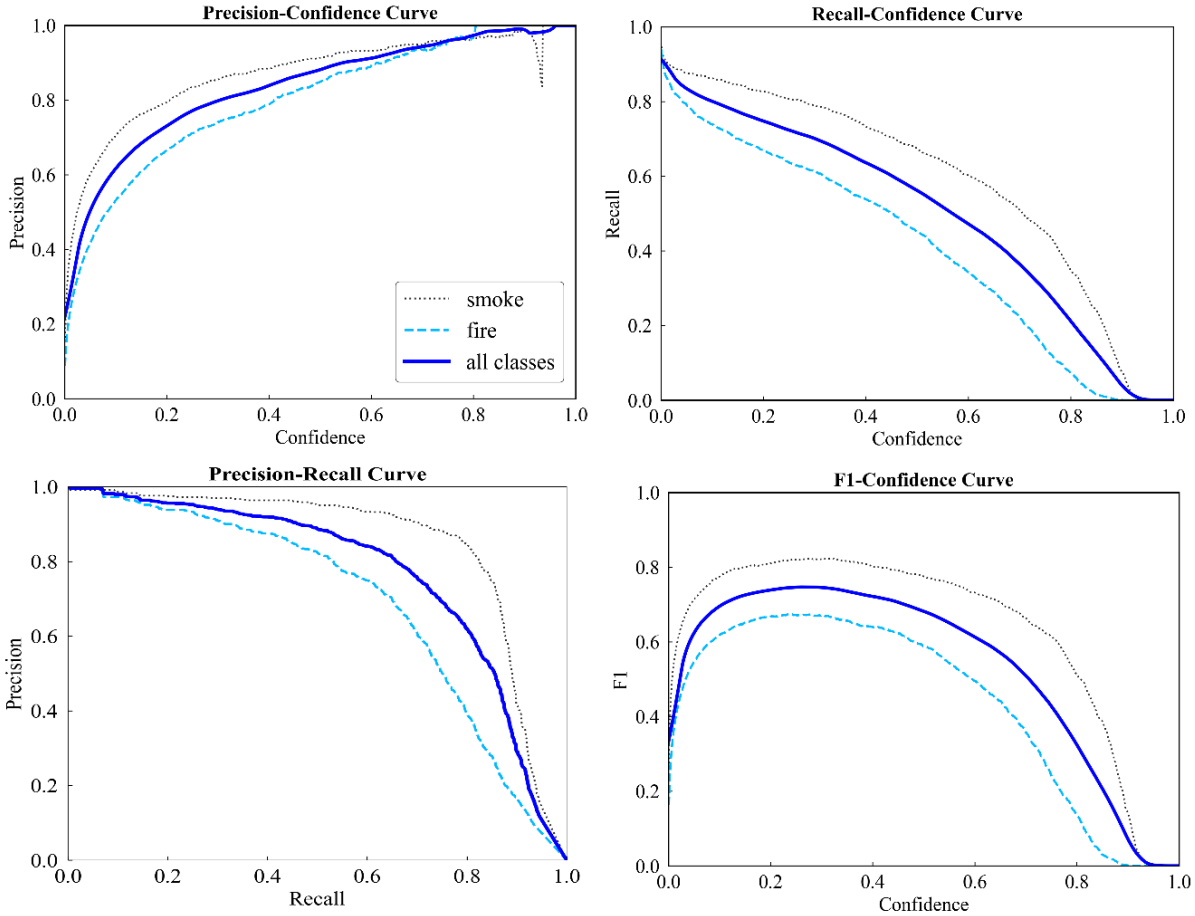


Figure 12: Four performance curves of FSD-Net

The figure reveals that the Precision-Confidence Curve maintains relatively smooth precision across the entire range of confidence thresholds, with only a slight drop observed when confidence exceeds 0.95. In other words, the improvement strategy proposed in this section effectively guarantees the model's detection stability across most confidence levels.

The Recall-Confidence Curve reflects that the model maintains high recall at confidence levels ranging from 0 to 0.8, particularly for smoke detection. Thus, the algorithm can sustain high recall even after filtering out low-confidence bounding boxes, specifying its strong performance in object detection tasks.

The F1-Confidence Curve provides an overall evaluation of recall and precision. FSD-Net achieves a favorable F1 score within the confidence threshold range of 0 to 0.8, reflecting that the model performs ideally in balancing precision and recall within this range.

Finally, the Precision-Recall Curve demonstrates the model's comprehensive capabilities. As precision and recall increase, the curve gradually moves toward the top-right corner, implying that FSD-Net can ensure high precision and recall while performing efficient detection.

Overall, FSD-Net effectively controls model size and computational expense while preserving high detection performance. Hence, it is suitable for fire smoke detection tasks in real outdoor environments.

3.6 Visualization of detection results

In this section, the detection results of the baseline model YOLOv11s and the proposed FSD-Net on different types of fire smoke images from the self-constructed dataset are compared to visually demonstrate the effectiveness of the proposed improvement strategies.

The detection results are divided into three scenarios: medium-to-large fire smoke targets, small fire smoke targets, and multi-target fire smoke scenes. The detection results for medium and large fire smoke targets are illustrated in Figure: 13~Figure 15.

Figure: 13 suggests that when the smoke density is uneven, the baseline model tends to incorrectly detect a single target as two separate parts. Additionally, missed detections may occur when flame and smoke targets coexist. In contrast, FSD-Net more effectively captures multi-scale feature information, contributing to significantly improved recognition precision and more detailed detection regions compared with the baseline model.

The detection results for small fire smoke targets are displayed in Figure 14. It can be observed that the feature fusion strategy of FSD-Net remarkably improves the model's perception capability for small-scale fire smoke targets. Under different backgrounds and lighting environments, the model accurately localizes small targets and achieves higher detection performance than the baseline model. In contrast, the baseline model is more prone to missed detections in such cases.

Detection scenarios containing multiple targets impose higher requirements on the detection performance of the model. The detection results for multi-target fire smoke scenes are presented in Figure 15. The detection performance of the baseline model is limited when multiple flame and smoke targets exist simultaneously in an image. Meanwhile, issues such as false detections, missed detections, and overlapping detection regions are more likely to occur.

The improvement strategies employed in FSD-Net effectively improve the model's generalization capability. Thus, it accurately detects multiple targets with high precision in different environments.

Figure: 13 Results of medium or large object fire smoke detection.

Figure 14: Results of the small object fire smoke detection test.

Figure 15: Results of multi-target fire smoke detection.

4 Discussion

Compared to a series of detection models, the improved model FSD-Net achieved higher mAP@50 (80.1%), mAP@95 (47.1%), precision (80.5%), and recall (74.1%), while reducing the number of parameters from 9.4M to 8.1M and increasing FPS from 86.0 to 89.3. The integration of Swin Transformer enhanced global feature perception. Meanwhile, the addition of the P2 layer and the GS-A2C2f module improved feature fusion performance, particularly for small objects. Additionally, the replacement strategy of GhostConv significantly lessened the number of parameters, and the inclusion of Inner-CIoU enabled faster inference. These modifications realized a balance between detection accuracy and model efficiency. The comparison images in Section 4.6 reveal that the model performed more robustly in various fire environments. With these improvements, the proposed model becomes more practical for outdoor fire smoke detection in complex environments, allowing early warning systems to more accurately identify and locate fire targets.

5 Conclusions

Outdoor fire-smoke detection suffers from large variations in target scale and morphology, blurred boundaries, complicated backgrounds, and the tendency for small targets to be missed. Given these challenges, FSD-Net was proposed in this paper. It is a multi-scale fire target detection algorithm improved from the YOLOv11s network. First, a C3Swin module based on the Swin Transformer was introduced to improve the model's capability, thereby recognizing multi-scale fire and smoke features. Second, shallow features from the P2 layer were incorporated into the PAN-FPN structure, and a lightweight feature fusion module GS-A2C2f was introduced to effectively improve feature fusion performance. Third, a local GhostConv replacement strategy was employed to lessen model parameters while maintaining detection performance. Finally, the conventional CIoU loss was replaced with the loss function Inner-CIoU, based on auxiliary bounding boxes, to alleviate gradient attenuation in high-IoU regions during training, contributing to improved training efficiency.

The above improvement strategies jointly generated a comprehensive optimization framework covering enhanced feature extraction, feature fusion optimization, computational effectiveness control, and training process improvement. Compared with the baseline model, FSD-Net improved precision, recall, mAP@50, mAP@95, and FPS by 0.5%, 2.5%, 2.3%, 1.1%, and 3.8%, respectively. Simultaneously, it lowered the number of parameters by 1.3M. Thus, it demonstrated better overall performance in comparison with several benchmark models.

In future work, the practical value of the proposed model can be further enhanced in several aspects. First, the proposed improvement strategies can be extended to newer YOLO frameworks with the continuous development of object detection technologies. Second, the adaptability of the model will be further improved because it still produces a small number of false positives under extreme conditions (such as in dark or very bright fire environments). Finally, considering practical application requirements, a real-time fire detection system suitable for edge devices (such as surveillance cameras, drones, and personal devices) will be further developed based on the proposed model and the Python framework.

Funding

This research was supported by the Key Scientific Research Project of Higher Education Institutions in Henan Province (No.25A520033).

Authors' contributions

It is hereby acknowledged that all authors have accepted responsibility for the manuscript's content and consented to its submission. They have scrupulously reviewed all results and unanimously approved the final version of the manuscript.

References

[1] J. P. Rafferty. (2025). *Los Angeles wildfires of 2025*. Available: https://www.britannica.com/event/Los-Angeles-wildfires-of-2025

[2] C. Jin *et al.*, "Video Fire Detection Methods Based on Deep Learning: Datasets, Methods, and Future D irections," *Fire*, vol. 6, no. 8, 2023.https://doi.org/10 .3390/fire6080315

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-C NN: Towards Real-Time Object Detection with Reg

ion Proposal Networks," *IEEE Transactions on Patt ern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 2017. https://doi.org/10.1109/TPAMI.2016.2577031

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980-2988. https://doi.org/10.1109/ICCV.2017.322

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779-788. https://doi.org/10.1109/CVPR.2016.91

[6] B. W. Xiao and C. M. Yan, "A lightweight global a wareness deep network model for flame and smoke detection," (in English), *Optoelectronics Letters*, vo l. 19, no. 10, pp. 614-622, Oct 2023. https://doi.org/10.1007/s11801-023-3041-x

[7] F. Y. Chen, M. Yang, and Y. Wang, "Recognition o f Forest Fire Smoke Based on Improved YOLOv8n Model," *FIRE TECHNOLOGY*, vol. 61, no. 5, pp. 3 351-3374, SEP 2025. https://doi.org/10.1007/s10694-025-01733-x

[8] C. M. Zhao, L. K. Zhao, K. Zhang, Y. H. Ren, H. Chen, and Y. H. Sheng, "Smoke and Fire-You Only Look Once: A Lightweight Deep Learning Model for Video Smoke and Flame Detection in Natural Scenes," *Fire-Switzerland*, vol. 8, no. 3, Mar 4 2025, Art. no. 104. https://doi.org/10.3390/fire8030104

[9] M. H. Ashraf, F. Jabeen, H. Alghamdi, M. S. Zia, and M. S. Almutairi, "HVD-Net: A Hybrid Vehicle Detection Network for Vision-Based Vehicle Tracking and Speed Estimation," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 8, p. 101657, 2023/09/01/ 2023. https://doi.org/https://doi.org/10.1016/j.jksuci.2023.101657

[10] Y. Liu *et al.*, "A Survey of Visual Transformers," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 6, pp. 7478-7498, 2024. https://doi.org/10.1109/TNNLS.2022.3227717

[11] Y. Zheng, G. Zhang, S. Q. Tan, Z. G. Yang, D. X. Wen, and H. S. Xiao, "A forest fire smoke detection model combining convolutional neural network and vision transformer," *FRONTIERS IN FORESTS AND GLOBAL CHANGE*, vol. 6, APR 17 2023, Art. no.1136969. https://doi.org/10.3389/ffgc.2023.1136969

[12] B. S. Sun and X. Cheng, "Smoke Detection Transformer: An Improved Real-Time Detection Transformer Smoke Detection Model for Early Fire Warning," *FIRE-SWITZERLAND*, vol. 7, no. 12, DEC 2024, Art. no. 488.https://doi.org/10.3390/fire7120488

[13] L. Wang, H. Li, F. Siewe, W. Ming, and H. Li, "Forest fire detection utilizing ghost Swin transformer with attention and auxiliary geometric loss," *Digital Signal Processing*, vol. 154, 2024. https://doi.org/10.1016/j.dsp.2024.104662

[14] M. E. Qureshi, M. H. Ashraf, M. W. Arshad, A. Kh an, H. Ali, and Z. U. Abdeen, "Hierarchical Feature Fusion With Inception V3 for Multiclass Plant Dise ase Classification," *Informatica*, vol. 49, no. 27, 202 5. https://doi.org/10.31449/inf.v49i27.8208

[15] D. X. Yin, P. L. Cheng, and Y. Huang, "YOLO-EPF: Multi-scale smoke detection with enhanced pool former and multiple receptive fields," *DIGITAL SIGNAL PROCESSING*, vol. 149, JUN 2024, Art. no. 104511.https://doi.org/10.1016/j.dsp.2024.104511

[16] Y. X. Li *et al.*, "Improving Fire and Smoke Detection with You Only Look Once 11 and Multi-Scale Convolutional Attention," *FIRE-SWITZERLAND*, vol. 8, no. 5, APR 22 2025, Art. no. 165. https://doi.org/10.3390/fire8050165

[17] D. Gragnaniello, A. Greco, C. Sansone, and B. Vento, "Fire and smoke detection from videos: A literature review under a novel taxonomy," *Expert Systems with Applications*, vol. 255, 2024. https://doi.org/10.1016/j.eswa.2024.124783

[18] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," *arXiv preprint arXiv:2410.17725*, 2024.

[19] H. Zhao *et al.*, "Psanet: Point-wise spatial attention network for scene parsing," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 267-283. https://doi.org/10.1007/978-3-030-01240-3_17

[20] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Harihara n, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *2017 IEEE Conference on Co mputer Vision and Pattern Recognition (CVPR)*, 20 17, pp. 936-944. https://doi.org/10.1109/CVPR.2017.106

[21] H. Li, P. Xiong, J. An, and L. Wang, "Pyramid Atte ntion Network for Semantic Segmentation," in *Proc . British Machine Vision Conference (BMVC)*, New castle, UK, Sep. 2018, p. 285. https://doi.org/10.2139/ssrn.5008171

[22] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251-1258. https://doi.org/10.1109/cvpr.2017.195

[23] Z. Liu *et al.*, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10012-10022. https://doi.org/10.1109/iccv48922.2021.00986

[24] D. M. Wang, Y. Qian, J. Y. Lu, P. Wang, Z. R. Hu, and Y. K. Chai, "Fs-yolo: fire-smoke detection based on improved YOLOv7," (in English), *Multimedia Systems*, vol. 30, no. 4, Aug 2024. https://doi.org/ARTN 215.10.1007/s00530-024-01359-z

[25] S. Choi, Y. Song, and H. Jung, "Study on Improving Detection Performance of Wildfire and Non-Fire Events Early Using Swin Transformer," *IEEE ACCESS*, vol. 13, pp. 46824-46837, 2025. https://doi.org/10.1109/ACCESS.2025.3528983

[26] C. Jin, A. Zheng, Z. Wu, and C. Tong, "Real-Time Fire Smoke Detection Method Combining a Self-Attention Mechanism and Radial Multi-Scale Feature Connection," *Sensors (Basel)*, vol. 23, no. 6, Mar 22 2023. https://doi.org/10.3390/s23063358

[27] J. Wang, X. Zhang, and C. Zhang, "A lightweight smoke detection network incorporated with the edge cue," *Expert Systems with Applications*, vol. 241, 2024. https://doi.org/10.1016/j.eswa.2023.122583

[28] Y. Tian, Q. Ye, and D. Doermann, "Yolov12: Attention-centric real-time object detectors," *arXiv preprint arXiv:2502.12524*, 2025.

[29] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. X u, "Ghostnet: More features from cheap operations," in *Proceedings of the IEEE/CVF conference on co mputer vision and pattern recognition*, 2020, pp. 15 80-1589. https://doi.org/10.1109/cvpr42600.2020.00165

[30] Z. Zheng *et al.*, "Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation," *IEEE Transactions on Cybernetics*, vol. 52, no. 8, pp. 8574-8586, 2022. https://doi.org/10.1109/TCYB.2021.3095305

[31] H. Zhang, C. Xu, and S. Zhang, "Inner-IoU: more effective intersection over union loss with auxiliary bounding box," *arXiv preprint arXiv:2311.02877*, 2023.

[32] P. Almeida, T. Rezende, A. Lisboa, and A. Barbosa, "Fire Detection based on Two-Dimensional Convolutional Neural Network and Temporal Analysis," *7th IEE LA-CCI, Temuco*, 2021. https://doi.org/10.1109/la-cci48322.2021.9769824

[33] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," in *Computer Vision – ECCV 2016*, Cham, 2016, pp. 21-37: Springer International Publishing. https://doi.org/10.1007/978-3-319-46448-0_2

[34] Y. Zhao *et al.*, "DETRs Beat YOLOs on Real-time Object Detection," in *2024 IEEE/CVF Conference o n Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 16965-16974. https://doi.org/10.1109/CVPR52733.2024.01605

[35] R. Khanam and M. Hussain, "What is YOLOv5: A deep look into the internal features of the popular object detector," *arXiv preprint arXiv:2407.20892*, 2024.

[36] R. Varghese and M. S, "YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness," in *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, 2024, pp. 1-6. https://doi.org/10.1109/ADICS58448.2024.10533619

[37] C.-Y. Wang, I. H. Yeh, and H.-Y. Mark Liao, "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information," in *Computer Vision – ECCV 2024*, Cham, 2025, pp. 1-21: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-72751-1_1

[38] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, "YOLOv10: Real-Time End-to-End Object Detection," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. https://doi.org/10.52202/079017-3429