Improved Real-Time Smoke Detection Model Based on RT-DETR

Yuanpan ZHENG*, Zeyuan HUANG, Binbin CHEN, Chao WANG, Yu ZHANG
School of Computer Science and Technology, Zhengzhou University of Light Industry, Zhengzhou 450000, Henan, China

***Abstract*—Fire remains a major threat to society and economic activities. Given the real-time demands of smoke detection, most research in deep learning has focused on Convolutional Neural Networks. The Real-Time Detection Transformer (RT-DETR) introduces a promising alternative for this task. This paper extends RT-DETR to address challenges such as morphological variations and interference in smoke detection by proposing the Realtime Smoke Detection Transformer (RS-DETR). RS-DETR uses smoke images with concentration data as input and employs a deformable attention module to manage morphological changes, enabling robust feature extraction. Additionally, a Cross-Scale Smoke Feature Fusion Module (CS-SFFM) is integrated to enhance detection accuracy for small and thin smoke targets through multi-scale feature resampling and fusion. To improve convergence speed and stability, Efficient Intersection over Union (EIoU) replaces Generalized Intersection over Union (GIoU) in feature scoring. The improved model achieves an average precision of 93.9% on a custom dataset, representing a 5.7% improvement over the original model, and demonstrates excellent performance across various detection scenarios.**

Keywords—RT-DETR; smoke detection; deformable convolution; multi-scale feature fusion; EIoU; image enhancement; dark channel

I. Introduction

Fire is a highly destructive disaster that poses significant risks to human society and economic activities. In 2022, fires caused approximately 17,040 casualties globally, including 3,790 deaths—the highest number since 2013 [1]. Early fire warning systems are crucial for minimizing damage.

Traditional fire detection methods primarily rely on physical sensors to identify early-stage smoke. However, these approaches are less effective outdoors, require frequent maintenance, and offer limited coverage [2]. Such limitations often lead to false alarms and missed detections, underscoring the inadequacy of traditional methods for modern fire prevention.

Early image-based smoke detection used manual feature classifiers like SVM and Random Forests, but these had limited robustness due to hardware and design constraints. With advances in hardware, deep learning methods have become the standard, offering better robustness, generalization, and integration with existing surveillance systems [3]. CNN-based models have gained attention for their precision [4], but they often require complex post-processing, increasing optimization difficulty and computational load, which can compromise robustness [5].

The DETR (Detection Transformer) [6] series introduced a new solution by applying the Transformer architecture to computer vision. DETR leverages self-attention to model global contextual information, transforming object detection into a set prediction task, thereby simplifying the process and enabling end-to-end detection. However, the extensive use of attention mechanisms makes DETR models complex and less suitable for real-time tasks [7]. To address this, Zhao et al. [8] proposed the RT-DETR model, capable of real-time detection. RT-DETR introduces an attention-based intra-scale feature interaction module and a cross-scale feature fusion module, enhancing training speed and detection performance. The structure of the RT-DETR model is shown in Fig. 1.

RT-DETR, as the first real-time Transformer-based detection model, offers greater robustness and easier optimization compared to the YOLO series models [9-16], while avoiding the computational overhead of NMS. Recognizing its potential for fire smoke detection, this paper selects RT-DETR-r18 as the baseline and introduces improvements in smoke feature extraction and multi-scale feature fusion. Key contributions include:

1) To better evaluate the model's effectiveness in real fire detection scenarios, this paper addresses the shortcomings of existing datasets and common interference factors in smoke detection. A high-quality smoke target detection dataset was constructed by filtering unannotated images from existing datasets, collecting smoke images from the internet, and manually annotating them.

2) Smoke morphology often changes significantly over time and due to various interference factors. To capture these graphical features, this paper uses the dark channel prior method to process smoke data, setting the model input as a four-dimensional tensor that includes smoke concentration information. Additionally, a large kernel deformable convolutional attention mechanism based on channel priors is designed to extract robust smoke information, effectively handling variations in smoke's spectral characteristics and spatial distribution.

3) Early smoke targets with high detection value are usually small and have blurred edges. To address the baseline model's low accuracy in identifying small targets captured from a distance, this paper optimizes and improves the cross-scale feature fusion module of the model using methods such as feature map scaling strategies, 3D convolution, and 3D pooling. This resolves the issue of losing detailed feature information during feature fusion, enhancing the model's smoke detection accuracy while reducing the model parameters.

4) The baseline model's feature query loss function, which uses Generalized Intersection over Union (GIoU), suffers from slow convergence. To address this, this paper employs Efficient Intersection over Union (EIoU) as the regression loss in feature scoring. EIoU considers both bounding box coordinates and dimensions, accelerating model convergence, improving stability, and reducing redundant detections.

The structure of this paper is as follows: Section II reviews related work on fire smoke detection; Section III elaborates on the algorithmic optimization proposed for the fire smoke detection task; Section IV presents experiments and analysis of the proposed model, compares it with mainstream detection models, and validates the effectiveness of the proposed approach; Section V summarizes the main contributions of this paper and discusses future research directions.

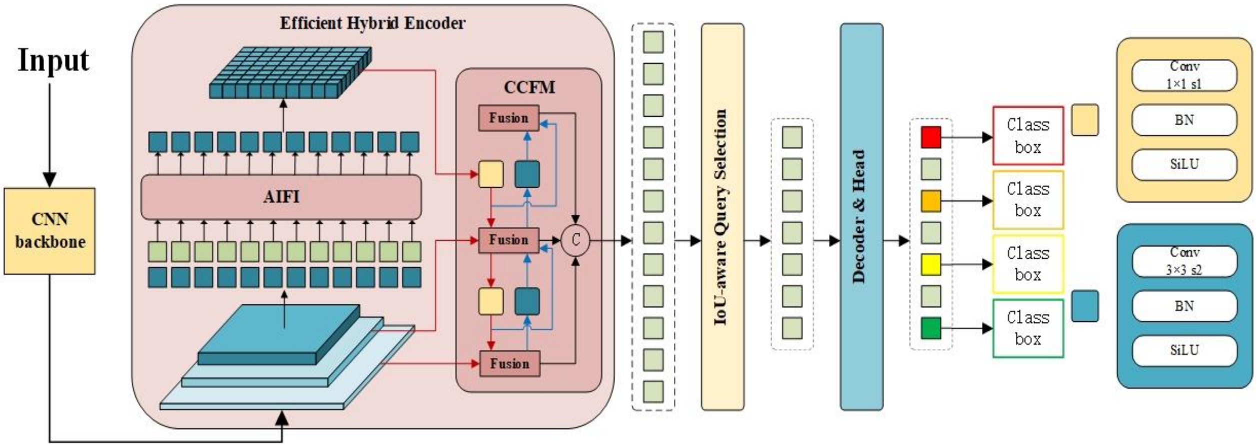

Fig. 1. The general architecture of RT-DETR.

II. Related Work

In recent years, significant progress has been made in fire smoke detection using deep learning. This section introduces related work on fire smoke detection based on different model architectures.

A. *CNN-based Fire Smoke Detection*

Frizzi et al. [17] utilized CNNs for smoke and flame detection by performing sliding window sampling on the feature map of the last convolutional layer instead of the original image, and recognizing smoke and flames in each block. Lin et al. [18] employed various backbone networks as feature extractors, combining them with Faster R-CNN [19], SSD [20], and R-FCN [21] frameworks for smoke detection. Zhang et al. [22] proposed a multi-scale convergence-coordinated feature pyramid network, which enhanced feature fusion efficiency and optimized NMS processing, thereby improving the accuracy and efficiency of detecting small and medium-sized fire smoke. Wang et al. [23] incorporated a self-attention mechanism into YOLOX [24] to enhance the model's ability to capture long-range dependencies. They also combined a self-collaborative mechanism with PAN [25] to achieve feature sharing and reduce redundant features, resulting in robust fire smoke detection. Zhan et al. [26] addressed the challenge of detecting highly transparent smoke by proposing a feature fusion scheme based on deconvolution and dilated convolution. This approach fused shallow visual information with deep semantic information along the channel dimension, enabling high-precision detection of distant aerial smoke. Sathishkumar et al. [27] introduced a transfer learning method based on lifelong learning to overcome the decline in model performance caused by insufficient training data, achieving efficient and accurate fire detection.

B. *Transformer-based Fire Smoke Detection*

To address these issues, Li et al. [28] leveraged the NMS-free algorithm concept from DETR and applied multi-scale deformable attention in the encoder of Deformable-DETR [29], along with lightweight optimizations. They also introduced a normalization-based attention mechanism, which accelerated network convergence and reduced deployment requirements. However, the model still suffers from repeated detections and insufficient detection accuracy. Similarly, Huang et al. [30] used Deformable-DETR as the baseline, integrating a multi-scale context contrast local feature module and a dense pyramid pooling module into the feature extraction module. This approach improved the detection accuracy for small and blurry smoke. However, the model's structure remains relatively complex, posing challenges for real-time detection tasks. Although these Transformer-based approaches eliminate the need for post-processing, the extensive stacking of attention mechanisms results in slow convergence and high deployment requirements, which still do not fully meet the practical demands of fire smoke detection.

III. Improvement Scheme

Feature extraction and feature fusion are two equally important components in object detection models. Feature extraction is responsible for deriving meaningful features from images, while feature fusion ensures the effective integration of these features, enabling the model to accurately detect objects in various complex scenarios. Given the susceptibility of smoke features to external interference and the uncertainty in scale, this chapter focuses on improving the baseline model in two key areas: robust smoke feature extraction and multi-scale feature fusion. The improved model is illustrated in Fig. 2.

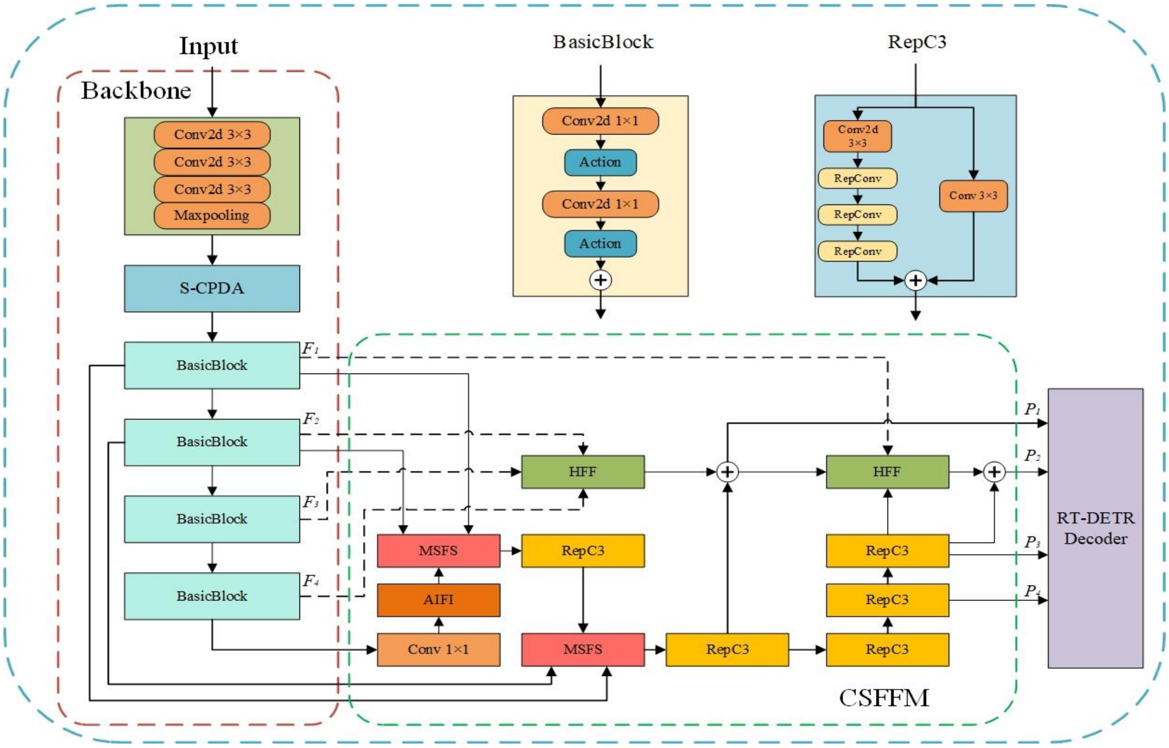

Fig. 2. Improved realtime smoke detection transformer, RS-DETR.

A. Robust Smoke Feature Extraction

Smoke is subject to significant variations in scale, shape, and spectral information due to environmental factors such as wind direction, wind speed, temperature, and humidity, as well as the thermodynamic and fluid dynamic properties of the smoke itself. These variations significantly impact the effectiveness of smoke detection tasks. Consequently, smoke detection models developed in the past often exhibited insufficient generalization capabilities, making it difficult to apply them across different scenarios.

This study addresses smoke’s graphical characteristics by focusing on both spectral features and spatial distribution variations. A four-dimensional vector, generated by combining the smoke's transmittance grayscale image and its RGB image, is used as the network input. Additionally, a specially designed attention mechanism is employed for robust smoke feature extraction to meet the demands of a highly generalizable network.

1) Smoke feature extraction aggregating concentration information: Smoke concentration is a crucial indicator of smoke intensity, exhibiting significant variation depending on the emission strength of the smoke. This variation greatly impacts the performance of neural network models in smoke detection tasks. Calculating the transmittance of smoke regions is a common and accurate method for estimating smoke concentration. To enhance the network's detection accuracy across different smoke concentrations and improve the algorithm's generalization capability in various scenarios, the network input is configured as a four-dimensional tensor generated by combining the smoke transmittance grayscale image, obtained through the dark channel prior method [31], with the smoke RGB image. The synthesized model input is illustrated in Fig. 3. The process of calculating the smoke transmittance image using the dark channel prior method can be described as follows:

$$A = I(y) \tag{1}$$

$$t(x) = 1 - \omega \min_{y\in\Omega(x)} \left(\min_{c\in\{r,g,b\}} \frac{I^c(y)}{A^c} \right) \tag{2}$$

$$J_{dark}(x) = \min_{y\in\Omega(x)} \left(\min_{c\in\{r,g,b\}} I^c(y) \right) \tag{3}$$

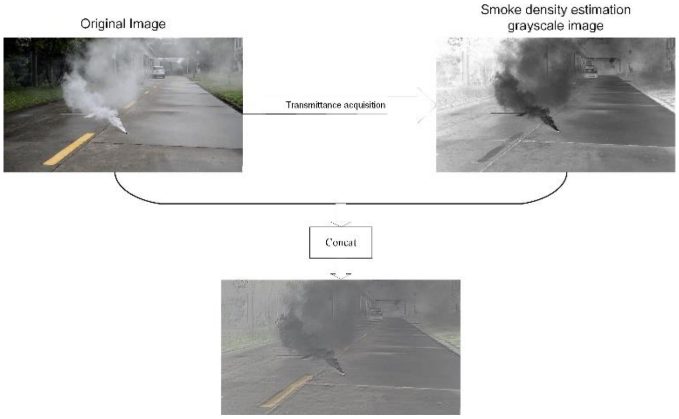
Original Image
Smoke density estimation grayscale image
Transmittance acquisition
Concat

Fig. 3. Concentration Feature Aggregation (CFA).

where, I represents the input image, and c denotes the color channel. Eq. (1) defines the dark channel, where $\Omega(x)$ is a window centered at x. Eq. (2) provides the atmospheric light estimation, where A is the estimated atmospheric light value, and $I(y)$ is the pixel value at the position with the highest intensity in the original image. Eq. (3) estimates the transmission rate, where $I^c(y)$ represents the intensity value of the c-th color channel at position y in the input image I, and A^c represents the value of atmospheric light in the c-th color channel.

2) Smoke feature extraction attention: The attention mechanism consists of two components: a channel attention module and a spatial attention module. These modules are integrated into the backbone network to enhance the extraction of smoke features. The structure of the attention mechanism is shown in Fig. 4. The overall process can be described as Eq. (4):

$$Output = CA(F) + SA(CA(F)) \tag{4}$$

where CA represents the channel attention, and SA represents the spatial attention.

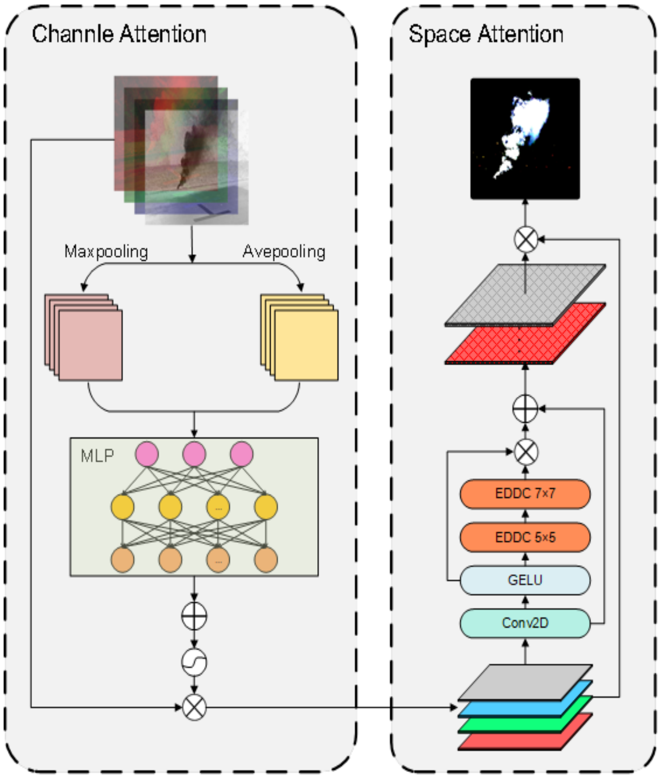

Fig. 4. Smoke channel-prior large-kernel deformable attention (S-CLDA).

a) Channel attention: The smoke channel information is extracted using the channel attention module. The Convolutional Block Attention Module (CBAM) [32] is employed, which aggregates the spatial information of the image through both average pooling and max pooling. This aggregation method produces a series of spatial indicators that capture the core attributes of smoke. These indicators are then processed by a simplified shared multi-layer perceptron (MLP), and the channel attention map is obtained by summing the MLP outputs. The MLP includes a hidden layer, designed to maintain the model's expressive capability while minimizing the number of parameters. The size of the hidden layer is specifically set to:

$$H = \frac{inchannels}{r} \times 1 \times 1 \tag{5}$$

where, H represents the size of the hidden layer, and r denotes the reduction ratio. This design carefully balances the model's lightweight nature with its ability to learn complex inter-channel relationships. By performing element-wise summation of the average pooling and max pooling results processed by the MLP, a detailed and expressive smoke channel attention map is generated.

The channel attention component can be expressed as:

$$F_{channel} = \sigma(MLP(F_{AVG}) + MLP(F_{MAX})) \tag{6}$$

$$F' = F \otimes F_{channel} \tag{7}$$

where, σ represents the sigmoid function, F denotes the input feature map, F' represents the output feature map, and AVG and MAX represent the average pooling and max pooling operations, respectively.

This strategy not only enhances the model's focus on critical channels within the smoke color features but also optimizes the use of computational resources by adjusting the hidden layer dimensions. This approach enables efficient aggregation of diverse channel information related to smoke, thereby improving the accuracy, robustness, and generalization capability of the smoke detection model.

b) Spatial attention: In the spatial attention module, the complex geometric variations exhibited by smoke due to its diffusive nature present a challenge. Using traditional convolutional kernels with fixed geometric structures across different smoke locations lacks adaptability to the dynamic morphology and scale changes of smoke. This limitation impedes the precise capture of spatial distributions and their variations. Additionally, valuable features in the thin edge regions of smoke may be suppressed during convolution operations due to their weaker signal strength, which can adversely affect the network's ability to learn the overall morphological characteristics of smoke.

To address these issues, we introduce deformable convolution [33] technology within the spatial attention module. This approach utilizes an additional convolutional layer to adjust the sampling region, resulting in adaptive convolutional kernels that improve the representation of smoke. To prevent the suppression of weak features, parallel convolution operations are employed, as shown in Fig. 5. A large kernel strategy [34] is implemented using depthwise convolution, dilated convolution, and 1×1 convolution to achieve a larger receptive field, allowing for the extraction of complete smoke region features. The enhanced deformable convolution can be expressed as Eq. (8):

$$y_{ed}\left(p_0\right)=\sum_{k=1}^{K^2} w_k \square\left(x\left(p_0+p_k\right)+x\left(p_0+p_k+\Delta p_k\right)\right) \tag{8}$$

p_0 represents the output position, w_k denotes the weight at the k-th position in the convolutional kernel, p_k is the standard positional offset of the kernel, and Δp_k is the learnable offset.

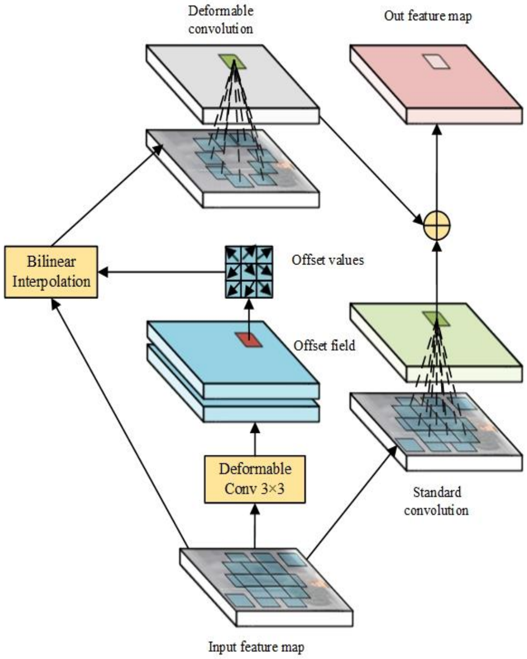

Fig. 5. Enhanced Deformable Depthwise Convolution (EDDC).

The equation for determining the kernel size of a $K \times K$ convolutional kernel in depthwise convolution and depthwise dilated convolution is:

$$DW=(2d-1)\times(2d-1) \tag{9}$$

$$DW_D=\left\lceil \frac{K}{d} \right\rceil \times \left\lceil \frac{K}{d} \right\rceil \tag{10}$$

where, K represents the kernel size, and d denotes the dilation rate.

The spatial attention component can be expressed as:

$$X'=GELU(X) \tag{11}$$

$$A=Conv_{1\times1}\left(EDDC\left(EDDC\left(X'\right)\right)\right) \tag{12}$$

$$Output=Conv_{1\times1}\left(A\otimes X'\right)+X \tag{13}$$

where, A represents the feature map processed by the EDDC module.

By introducing deformable convolution, the kernel offsets are calculated using bilinear interpolation, and a large kernel convolution strategy is implemented through depth-wise convolution and related techniques. This approach expands the receptive field while controlling the increase in the number of parameters, enabling the model to adapt to the complex spatial distribution of smoke. The channel attention and spatial attention modules focus on the spectral features and spatial distribution features of smoke, respectively, allowing for targeted extraction of robust smoke features.

A. *Cross-Scale Smoke Feature Fusion Module*

Transformer-based detectors utilize self-attention mechanisms for object localization, allowing them to cover the entire image. However, these models often focus more on larger target regions, resulting in suboptimal performance when detecting small objects. A common solution to this issue is multi-scale feature fusion using feature summation or concatenation. However, simple addition or concatenation methods lack selectivity across scales and lead to relatively independent channels after fusion. Implementing dynamic scale attention or more complex fusion strategies could address these limitations, but they would significantly increase computational complexity, thereby affecting the model's detection efficiency.

Ming Kang et al. [35] proposed the use of Gaussian blur to simulate images at different observation scales, effectively preserving both image details and structural features, and facilitating the fusion of deep and shallow features. This approach mitigates the information loss that often occurs with traditional concatenation and stacking methods.

Building on this idea, this paper redesigns the cross-scale smoke feature fusion module by employing a nearest-neighbor interpolation scheme to supplement the detail information of feature maps at different scales. This approach enables multi-level feature fusion with minimal information loss. Given that early smoke often appears as small targets, a small object detection branch is added to enhance detection accuracy. The structure of the cross-scale smoke feature fusion module is illustrated in Fig. 6.

In this module, RepC3 is the native component from RT-DETR, enhancing feature extraction and representation by stacking residual convolutional blocks. This design improves the model's expressive capability.

The Multi-Scale Feature Scaling Module (MSFS) applies adaptive pooling and nearest-neighbor interpolation to both large- and small-scale feature maps, adjusting their sizes to match the medium-scale feature map before channel concatenation. This approach magnifies small target features while preserving edge clarity and background information, enabling the network to capture more precise detail features. This module can be described as:

$$\begin{aligned} l' &= adaptive\ max\ \ pool2d\left(l, size\right) \\ &+ adaptive\ avg\ \ pool2d\left(l, size\right) \end{aligned} \tag{14}$$

$$s'=interpolate\left(s, size, mode=nearest\right) \tag{15}$$

$$Out_{lms}=concat\left(\ l', m, s', dim=1\right) \tag{16}$$

where, l, m, and s represent the large, medium, and small-scale feature maps, respectively, $size$ refers to the size of the medium-scale feature map, and lms denotes the output after the concatenation of the feature maps.

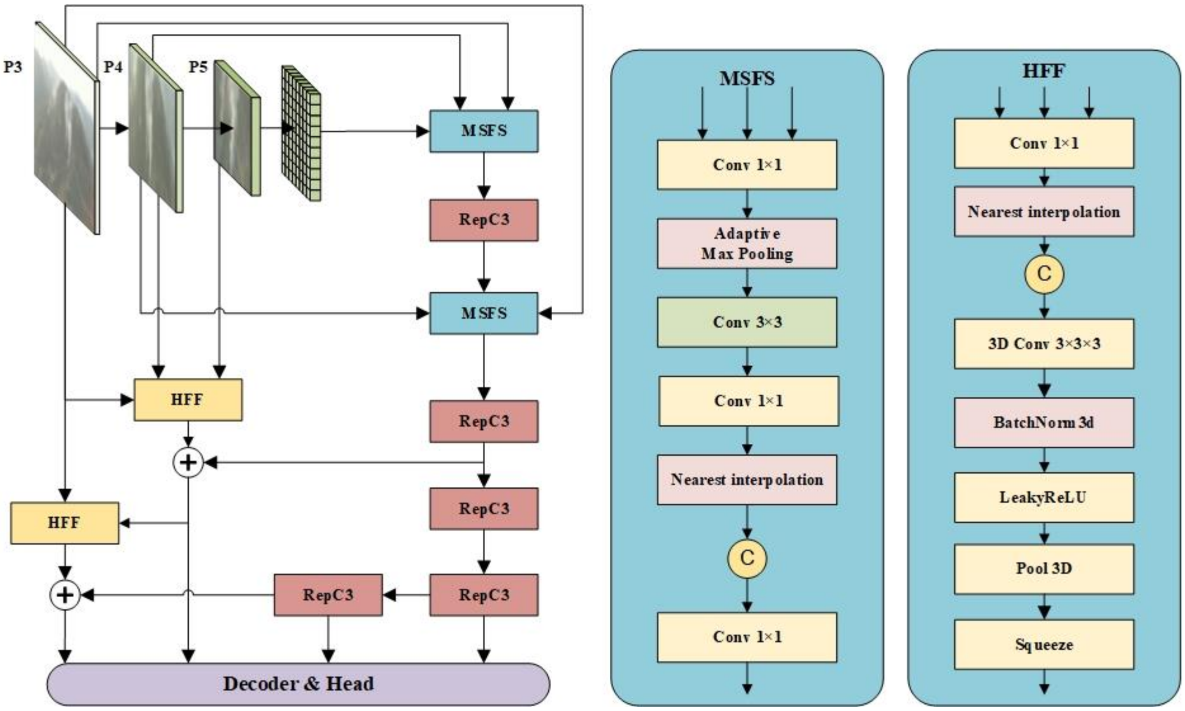

Fig. 6. Cross-Scale Smoke Feature Fusion Module (CS-SFFM).

The Hierarchical Feature Fusion Module (HFF) upscales the smaller-scale feature maps P_s and P_m to match the resolution of the larger-scale feature map P_l . Then, the three adjusted feature maps are fused using 3D convolution, followed by max pooling to process the output. This approach retains high-level semantic information while incorporating low-level detail, providing the network with more expressive features. The HFF can be described as:

$$P_s' = interpolate\ (P_m\ ,\ size\ of\ P_l) \tag{17}$$

$$combine = cat\left(unsqueeze\left(\ P_l\ ',\ P_m\ ',\ P_s\ '\right)\right) \tag{18}$$

$$conv3d = Conv3d\left(combine\right) \tag{19}$$

$$act = LeakyReLU\left(bn\left(conv3d\right)\right) \tag{20}$$

$$x = squeeze\left(MaxPool3d\left(act\right)\right) \tag{21}$$

where P represents the feature map, l , m , and s represent different feature map levels, and P' denotes the feature map after upsampling.

Compared to the CCFM module used in RT-DETR, the CS-SFFM module achieves cross-scale multi-level feature fusion through scaling and feature stacking, making more effective use of information from different scales. Furthermore, it constructs a micro-scale feature branch specifically for small object detection, utilizing the large, medium, and small-scale features. This design enhances the ability to detect early-stage, smaller-scale smoke, thereby improving detection accuracy.

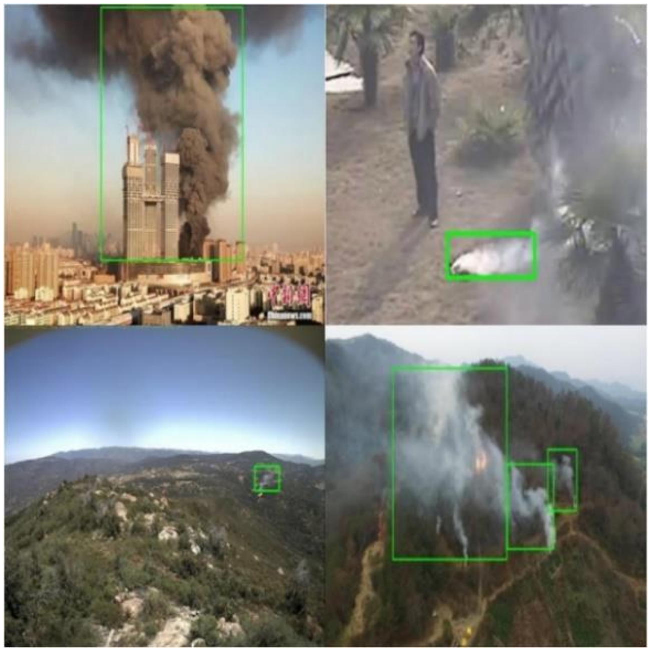

Fig. 7. Dataset annotation status.

B. Loss Function Optimization

In the label matching phase during training, RT-DETR utilizes a combination of Hungarian matching and IoU soft labels to align localization and classification, which allows the decoder to obtain higher-quality initial object queries. RT-DETR employs GIoU [36], which provides a more comprehensive evaluation by focusing on the minimum enclosing box of the predicted and ground truth boxes, addressing the issue when these boxes do not overlap. However, when the predicted box is entirely within the ground truth box, GIoU degrades to IoU, leading to slower regression speed. Additionally, due to the often irregular shape of smoke and its blurred boundaries, GIoU's focus on the pixel-level overlapping area makes it difficult to perform an effective evaluation.

EIoU [37] minimizes the height difference between the predicted and ground truth boxes while focusing on the minimum enclosing box, enabling more accurate box evaluation for detection targets with blurred boundaries and irregular shapes. EIoU can be divided into three components: IoU loss L_{IoU} 、 distance loss L_{dis} 、 and aspect ratio loss L_{asp}, The expressions are as Eq. (22):

$$\begin{aligned} L_{EIoU} &= L_{IoU} + L_{dis} + L_{asp} \\ &= 1 - IoU + \frac{1}{2}\left(\frac{d_{center}^2}{d_{diag}^2} + \frac{\alpha \cdot AR_{diff}^2}{AR_{sum}^2}\right) \end{aligned} \tag{22}$$

where, d_{center} is the Euclidean distance between the center points of the predicted box and the ground truth box. d_{diag} is the length of the diagonal of the enclosing box. AR_{diff} is the difference in aspect ratios between the predicted box and the ground truth box. AR_{sum} is the sum of the aspect ratios of the predicted box and the ground truth box. α is a coefficient used to adjust the weight of the aspect ratio loss.

Compared to GIoU, EIoU accounts for uncertainty by introducing a probability distribution to model the position of the bounding box. This approach more accurately reflects the relative position and size between bounding boxes, and through expectation calculations, it prevents the loss from being reduced to zero even when the predicted box is very close to the ground truth box, thus avoiding overfitting caused by perfect scores during training. Additionally, EIoU introduces a scaling factor that can dynamically adjust based on the target size. This ensures that even if the overlap between the predicted box and the ground truth box is small for small targets, the score will not be overly penalized, thereby improving the model's accuracy and robustness in detecting small objects. For these reasons, we chose EIoU to replace GIoU in the label matching phase.

IV. Experiment and Algorithm Performance Evaluation

To validate the effectiveness of the model improvements, we conducted ablation experiments on the enhanced model using a self-constructed dataset. Additionally, we tested the performance of several representative real-time object detection models, the original RT-DETR-r18 model, and the improved model on the same dataset to assess their ability to detect fire smoke in the shortest possible time. The performance of the improved model was thoroughly evaluated.

A. Dataset

Currently, the number of publicly available real-world outdoor fire smoke datasets is limited. To develop a detection model with optimal performance, it is crucial to consider the most challenging detection scenarios. These include interference factors such as backlighting, long distances, strong winds, color variations, dense targets, and complex backgrounds. We collected 7,834 smoke images from unannotated public datasets and the internet, further filtering them based on these interference factors. After ensuring that all types of interference were represented and removing low-quality images, we selected 1,868 images, each containing at least one smoke target. These images were manually annotated using the Labeliмg tool to create a custom YOLO-format smoke dataset. Fig. 7 shows dataset annotation status. The dataset was then randomly divided into training and test sets in an 8:2 ratio. All image sizes were adjusted to 640×640 to enhance detection speed.

Data preprocessing involved color space conversion and random mosaic processing to further augment the dataset and improve the model's robustness across different detection scenarios.

B. Implementation Details

1) Hardware and software environment: The hardware environment for the experiments in this paper is shown in Table I.

TABLE I. Experimental Environment

CPU	AMD EPYC 7773X @ 3.50GHz
GPU	GeForce RTX 4090
RAM	80G
Operating System	Ubuntu 20.04
Programming Language	Python 3.8
Deep Learning Framework	Pytorch 2.0.0
GPU Acceleration Library	Cuda 11.8

2) Training hyperparameter settings: The hyperparameters used in the experiments are listed in Table II.

TABLE II. Training Hyperparameter Settings

HYPERPARAMETER	**Value**
Optimizer	AdamW
Epochs	150
Batch size	16
Learning rate decay	cosine
Learning rate	0.0001
Weight decay	0.0001

3) Evaluation metrics: When evaluating the smoke detection performance of the model, we used five metrics: Recall, Average Precision (AP), model parameters (Params), Giga Floating Point Operations Per Second (GFLOPS), and Frames Per Second (FPS).

Recall is a crucial metric for assessing the detection capability of the model. It represents the proportion of correctly detected smoke instances out of all actual smoke samples. The calculation formula is as follows:

$$Recall = \frac{TP}{TP + FN} \tag{23}$$

where, TP refers to the number of smoke instances correctly detected by the model, while FN refers to the number of smoke instances that the model failed to detect. A high recall indicates that the model can more comprehensively detect smoke.

AP represents the model's average detection accuracy across different confidence thresholds. The calculation formula is as follows:

$$Precision = \frac{TP}{TP + FP} \tag{24}$$

$$AP = \int_0^1 P\left(R\right) dR \tag{25}$$

where, AP represents the Average Precision, $P(R)$ denotes the precision value at a given recall R, N is the total number of classes.

Model parameters refer to the total number of trainable parameters in the model, which is an indicator of the model's complexity and storage requirements. A larger number of parameters typically implies a higher model complexity, potentially requiring more computational resources and storage space. The calculation formula is as follows:

$$Params = \sum_{l=1}^{L} Params_l \tag{26}$$

where, L is the total number of layers in the model, and $Params_l$ represents the number of parameters in the l -th layer.

Model computational cost refers to the total number of computational operations required during inference or training. It is an important metric for assessing the complexity and operational efficiency of a model. A higher computational cost typically indicates that the model requires more computational resources and time to complete inference or training.

$$Total\ GFLOPs = \sum_{i=1}^{L} GFLOPs_i \tag{27}$$

where L represents the total number of layers in the model, and $GFLOP_{Si}$ denotes the computational cost of the i -th layer.

FPS indicates the number of image frames a model can process per second during operation, serving as a key metric for evaluating the model's real-time performance. A higher FPS value signifies faster processing speed, making the model more suitable for real-time detection scenarios, such as video surveillance systems. The calculation of FPS typically considers the model's inference time and processing capability.

$$FPS = \frac{N}{T} \tag{28}$$

where N represents the number of processed image frames, and T is the total time taken to process these N frames.

Considering the concept of transfer learning, the experiments utilized pre-trained weights obtained from training on the VOC 2007 dataset. These pre-trained weights were used as initialization for training on the dataset in this study.

C. Ablation Experiments

1) Network input ablation experiment: To validate the effectiveness of the input aggregation strategy for concentration features and evaluate the efficacy of using CFA as network input, we conducted ablation experiments on both RT-DETR and YOLOV8m, focusing on their impact on model detection accuracy. The experimental results are shown in Table III. Using CFA as model input improves the detection accuracy for smoke. In RT-DETR, using CFA as input resulted in AP50 and AP95 scores of 0.882 and 0.585, respectively, representing an improvement of 1.2% and 0.9% compared to using RGB input, which achieved scores of 0.870 and 0.574. In YOLOV8m, using CFA as input yielded AP50 and AP95 scores of 0.856 and 0.602, respectively, reflecting increases of 1.1% and 2.1% over RGB input. These results indicate that replacing RGB images with CFA images as network input can effectively enhance the model's performance in smoke detection tasks.

TABLE III. ABLATION STUDY ON INPUT TYPES

Model	**Input Type**	AP_{50}	AP_{95}
RT-DETR	RGB	0.870	0.574
	CFA	0.882	0.585
YOLOV8m	RGB	0.845	0.581
	CFA	0.856	0.602

2) Effectiveness of improvements: To evaluate the benefits of each improvement in the enhanced network model for smoke detection tasks, we conducted six ablation experiments focusing on the three main improvements. To ensure that the experiments accurately reflect the impact of the network structure improvements and eliminate additional interference, all experiments used CFA images as the network input and were trained for 150 epochs on the custom dataset. Table IV presents the experimental results of the models under different configurations. First, we tested the baseline model, and then we sequentially added different improvement schemes. The specific experiments are as follows:

TABLE IV. ABLATION STUDY ON IMPROVED MODULES

Experiment Number	**Improvement Scenarios**			**Evaluation Indicators**					
	EIoU	**S-CLDA**	**CSFFM**	$Recall(\%)$	$AP_{50}(\%)$	$AP_{95}(\%)$	$Params(M)$	$GFLOPs$	$FPS(Hz)$
1	✕	✕	✕	0.860	0.882	0.585	20.18	57.3	**65**
2	√	✕	✕	0.873	0.887	0.592	20.18	57.3	65
3	√	√	✕	0.884	0.920	0.616	20.48	67.3	48
4	√	✕	√	0.878	0.915	0.627	**15.07**	59.7	52
5	✕	√	√	0.889	0.933	0.641	15.47	68.3	46
6	√	√	√	**0.895**	**0.939**	**0.648**	15.47	68.3	45

a) In Experiment 1, the baseline model was used without any improvement schemes. The results were: recall of 0.867, AP50 of 0.882, AP95 of 0.585, with a parameter count of 20.18 million and a computational requirement of 57.3 GFLOPS.

b) In Experiment 2, EIoU was incorporated, increasing recall to 0.873, AP50 to 0.887, and AP95 to 0.592, while the parameter count and computational load remained nearly unchanged.

c) In Experiment 3, building on the addition of EIoU, the S-CLDA was further introduced. The inclusion of attention mechanisms significantly improved the model's responsiveness and accuracy in detecting smoke targets, with recall rising to 0.902, AP50 reaching 0.920, and AP95 increasing to 0.616. The parameter count slightly increased to 20.48 million, and due to the integration of deformable convolutions and depth-wise separable convolutions, the computational cost modestly rose to 67.3 GFLOPS.

d) In Experiment 4, the CCFM was replaced with CS-SFFM, in addition to EIoU. This new strategy provided more refined cross-scale feature fusion with minimal information loss, significantly enhancing the model's ability to accurately localize targets. Recall increased to 0.897, AP50 reached 0.915, and AP95 rose to 0.627. Since CS-SFFM uses 3D convolutions and 3D pooling for feature fusion instead of multiple stacked convolutional layers, the computational cost slightly increased to 59.7 GFLOPS, while the parameter count significantly decreased to 15.07 million.

e) In Experiment 5, both S-CLDA and CS-SFFM were applied, while EIoU was omitted from the loss function. Despite the absence of EIoU optimization, the introduction of the remaining improvement modules still considerably enhanced the model's smoke detection performance. Recall rose to 0.911, AP50 reached 0.933, and AP95 increased to 0.641.

f) In Experiment 6, all improvement schemes were implemented simultaneously. This configuration yielded the best model performance, with recall increasing to 0.923, AP50 reaching 0.939, and AP95 rising to 0.648. The parameter count and computational cost were maintained at 15.47 million and 68.3 GFLOPS, respectively.

The experimental results demonstrate that replacing GIoU with EIoU enhances model accuracy without increasing additional parameters or computational load. The application of S-CLDA and CS-SFFM positively impacts both recall rate and detection accuracy in smoke detection tasks.

These improvements enhance detection performance while effectively controlling the growth in computational cost and significantly reducing the number of model parameters. In summary, the proposed improvements effectively enhance the model's performance in executing smoke detection tasks.

D. Performance Comparison Experiments

To validate the effectiveness of the proposed algorithm, four mainstream real-time object detection algorithms—YOLOv5m, YOLOv6m, YOLOv7, and YOLOv8m—were selected for comparison, along with the baseline model RT-DETR-r18.

1) Comparison of evaluation metrics: In the fire smoke detection task, we compared the training curves of the mainstream YOLO series algorithms with the baseline algorithm and our proposed model. The corresponding curves were plotted to provide a more intuitive observation of their training progress and differences, as shown in Fig. 8.

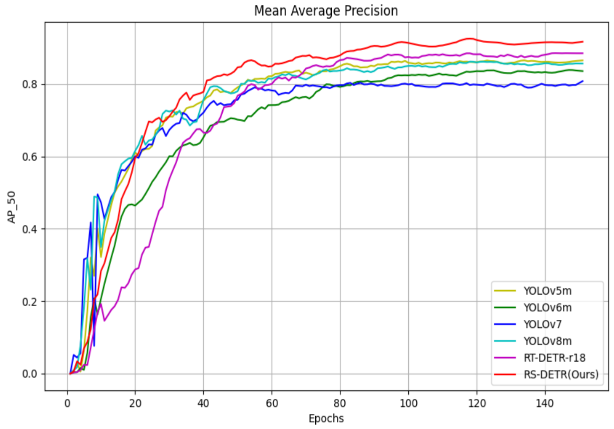

Fig. 8. Different algorithms' AP50 variations during training.

All models achieved convergence within 150 epochs. Our model demonstrated excellent performance in terms of accuracy and maintained high stability throughout the entire training process. Although the baseline model's accuracy was only slightly lower than that of our model, it exhibited significant fluctuations during training and had a slower convergence rate, falling behind the other algorithms. Overall, the improved model outperformed the baseline model in both detection accuracy and training stability.

The test results of each model on the self-built dataset are shown in Table V. Our algorithm achieved the best accuracy with 15.47 million parameters, an AP50 of 0.939, and an AP95 of 0.648. This success can be attributed to the more targeted and accurate smoke feature extraction enabled by the S-CLDA attention mechanism, as well as the refined multi-level feature fusion facilitated by CS-SFFM, which preserves more low-level features through adaptive pooling and interpolation.

TABLE V. COMPARATIVE EXPERIMENTS ON SELF-BUILT DATASET

Compare Models	**Evaluation Indicators**				
	Recall	AP_{50}	AP_{95}	*Param*	*GFLOPs*
YOLOv5m	0.839	0.863	0.564	21.2	64.6
YOLOv6m	0.802	0.834	0.544	24.85	161.7
YOLOv7	0.782	0.805	0.537	36.9	104.7
YOLOv8m	0.825	0.856	0.602	25.85	79.3
RT-DETR-r18	0.860	0.882	0.585	20.18	57.3
RS-DETR(ours)	**0.895**	**0.939**	**0.638**	**15.47**	68.3

2) Multi-scale smoke detection comparison experiments: Fire smoke undergoes significant scale variations at different stages, with early-stage smoke, which is often of high detection value, typically being smaller in size. To assess the model's detection performance across different stages of smoke, we designed a multi-scale smoke detection experiment. To visually represent the model's effectiveness in detecting smoke targets of varying scales, we applied the K-means [38] algorithm to cluster the test set and then divided the test set based on the clustering results. The clustering outcomes are shown in Fig. 9, where the centroids of the large, medium, and small target clusters correspond to [0.107, 0.131], [0.246, 0.325], and [0.333, 0.587], respectively.

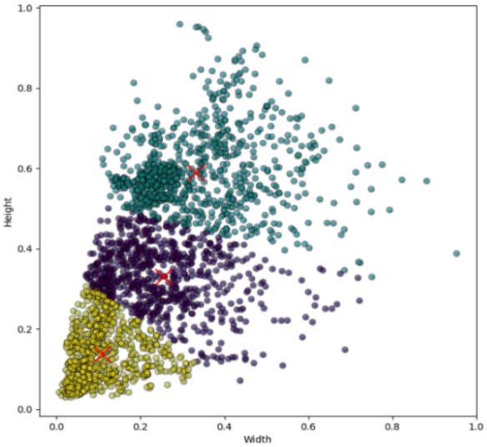

Fig. 9. Analysis of smoke scale distribution.

As shown in Table VI, after filtering and dividing the dataset, the test set of 374 images includes 145 images with small smoke targets, 152 images with medium smoke targets, and 77 images with large smoke targets.

TABLE VI. TEST SET TARGET SEGMENTATION RESULTS

Target Scale	Small Objects	Medium Objects	Large Objects
Target Quantity	145	152	77

Fig. 10 presents the statistical results of each algorithm's performance in detecting smoke targets of different scales. The results indicate that in the multi-scale smoke detection comparison, RS-DETR outperformed other mainstream real-time detection algorithms across all smoke scales. Compared to the YOLO series detection models, the improved model demonstrated particularly outstanding performance in detecting small-scale smoke. This improvement can be attributed to the CS-SFFM module in RS-DETR, which employs bilinear interpolation for scaling, effectively mitigating the loss of fine-grained feature details. Consequently, the model demonstrates enhanced sensitivity in capturing the distinctive characteristics of small-scale smoke targets, thereby reducing the likelihood of misclassification as background.

3) Comparison of detection results under interference factors: To validate the model's detection performance under various interference factors, we selected smoke images with strong wind, backlighting, long distances, color differences, and dense targets for comparison. The detection results are shown in Fig. 11. The results indicate that YOLOv5m performed poorly in detecting smoke targets with color differences and exhibited repeated detections when faced with distant, backlit targets. YOLOv6m and YOLOv7 both experienced missed or false detections in scenarios involving background interference, dense small targets, and backlighting. YOLOv8m also showed missed detections when detecting smoke targets with color differences. The baseline model encountered missed detections and false detections when dealing with dense small targets and backlit targets, likely due to the NMS-free strategy's inability to accurately determine whether to retain detection boxes for adjacent targets. The improved algorithm presented in this paper was able to correctly detect smoke targets in all these challenging scenarios, demonstrating higher detection accuracy and robustness, thereby meeting the practical application requirements for fire smoke detection.

4) Heatmap comparison: Heatmaps are a visualization technique used to display the intensity distribution of objects detected by a model within an input image. They typically indicate the location and confidence of the detected targets, with brighter areas representing higher confidence levels. We compared the heatmaps generated by the baseline model and the improved model, as shown in Fig. 12. The heatmap on the left corresponds to the baseline model, RT-DETR-r18, showing that the model primarily focuses on the central region of the smoke, with lower attention to the edges. In contrast, the second heatmap corresponds to our improved model, where the highlighted areas cover both the main body and the diffuse portions of the smoke, nearly encompassing the entire smoke region. Additionally, the heatmap of our model demonstrates higher attention to the overall structure of the smoke and effectively responds to thin smoke, indicating greater confidence in detecting smoke targets. These observations confirm that the improved model outperforms RT-DETR-r18 in smoke detection tasks.

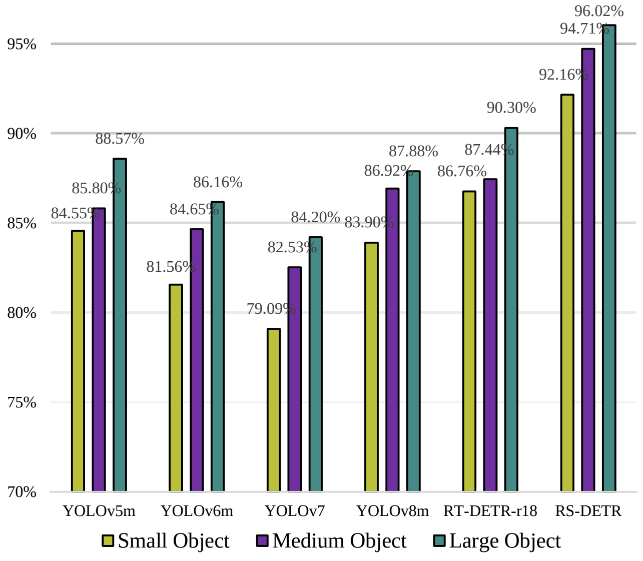

Fig. 10. Multi-scale object detection performance comparison.

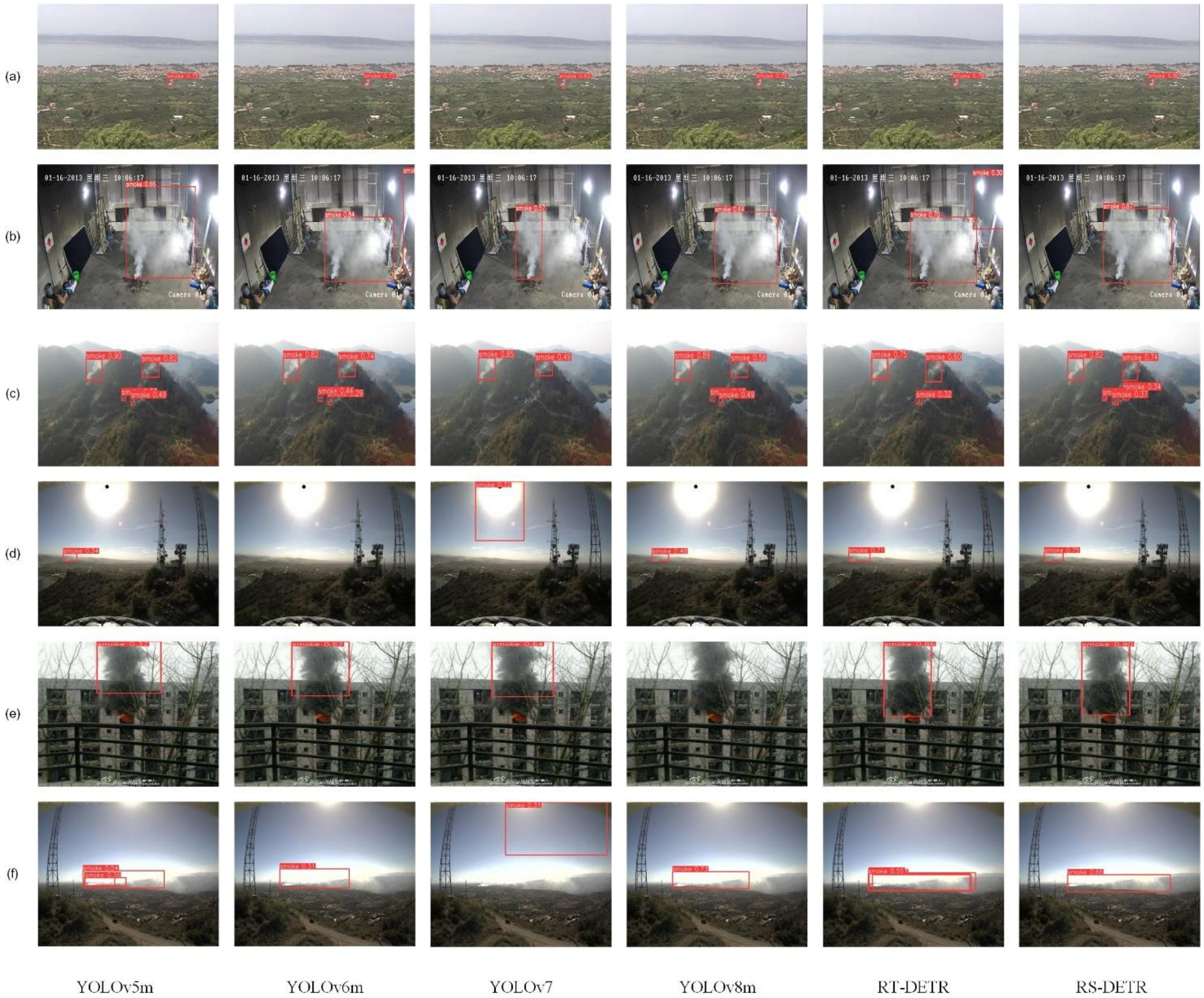

Fig. 11. Detection results under interference.

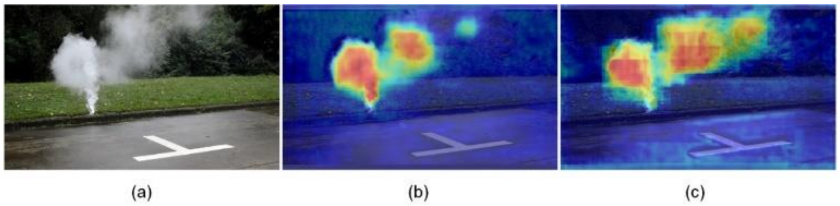

Fig. 12. The comparison results of the heatmap: (a) Original Image (b) RT-DETR-r18 (c) RS-DETR

V. Conclusion

This paper introduces an enhanced fire smoke detection algorithm based on RT-DETR, focusing on improving accuracy, real-time performance, and robustness against various interference factors. Key improvements include using the dark channel prior method for smoke concentration input, integrating the S-CLDA attention mechanism for robust feature extraction, and optimizing multi-scale feature fusion through the CSFFM module with 3D convolution and interpolation. The EIoU loss function further enhances detection accuracy for small targets and reduces redundant detections. Experiments on a self-made smoke detection dataset show that the improved model outperforms mainstream YOLO models and the RT-DETR-r18 baseline in AP50 and AP95 metrics while maintaining high detection speed. Specifically, the model achieved a 5.9% increase in AP50 and a 5.7% increase in AP95 with a 23.3% reduction in parameters, balancing accuracy and efficiency. This study confirms the potential of RT-DETR in fire smoke detection and demonstrates the effectiveness of the proposed improvements. Future work will focus on further optimizing the model, exploring advanced feature extraction and fusion strategies, and validating the model's robustness across diverse datasets and real-world scenarios to provide more reliable and efficient fire detection technology.

ACKNOWLEDGMENT

This study was supported by the Key Scientific Research Project Plan for Higher Education Institutions of Henan Province, China (No.25A520033).

REFERENCES

[1] “In the first half of 2023, there were over 3,000 fires per day on average nationwide.” National Fire and Rescue Administration, July. 2023. https://www.119.gov.cn/qmxfgk/sjtj/2023/38420.shtml.

[2] J. He, H. Lin, and G. Xu, “Overview of Research on Smoking Detection Methods in Computer Vision.”Computer Engineering and Applications, vol.60, no.1, pp. 40-56. 2024.

[3] Wahyono, A. Harjoko, A. Dharmawan, F. D. Adhinata, G. Kosala, and K.H. Jo, "Real-Time Forest Fire Detection Framework Based on Artificial Intelligence Using Color Probability Model and Motion Feature Analysis," Fire, vol. 5, no. 1, p. 23, 2022.

[4] Y. Al-Smadi, "Early wildfire smoke detection using different yolo models," Machines, vol. 11, no. 2, p. 246, 2023.

[5] Y. Zhou, L. Xia, J. Zhao, R. Yao, and B. Liu, "Efficient convolutional neural networks and network compression methods for object detection: A survey," Multimedia Tools and Applications, vol. 83, no. 4, pp. 10167-10209, 2024.

[6] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in European conference on computer vision, 2020, pp. 213-229.

[7] T. Shehzadi, K. A. Hashmi, D. Stricker, and M. Z. Afzal, "2d object detection with transformers: a review," arXiv preprint arXiv:2306.04670, 2023.

[8] Y. Zhao et al., "Detrs beat yolos on real-time object detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 16965-16974.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Look Only Once," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779-788.

[10] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 7263-7271.

[11] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.

[12] A. Bochkovskiy, C. Wang, and H. Liao, "Yolov4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, 2020.

[13] “Yolov5.” Ultralytics, 2021. https://github.com/ultralytics/yolov5.

[14] C. Li et al., "YOLOv6: A single-stage object detection framework for industrial applications," arXiv preprint arXiv:2209.02976, 2022.

[15] C. Wang, A. Bochkovskiy, and H. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 7464-7475.

[16] “YOLO by Ultralytics.” Ultralytics, 2023. https://github.com/ultralytics/ultralytics.

[17] S. Frizzi, R. Kaabi, M. Bouchouicha, J. Ginoux, E. Moreau, and F. Fnaiech, "Convolutional neural network for video fire and smoke detection," in IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, 2016, pp. 877-882.

[18] Z. Lin ,Y. Shen," Research on Fire Warning Algorithm Based on Deep Convolutional Neural Network " Information &Commmunications, no. 5, pp. 38-42, 2018.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 2017.

[20] W. Liu et al., "Ssd: Single shot multibox detector," in Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, 2016, pp. 21-37.

[21] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," Advances in neural information processing systems, vol. 29, pp. 379-387, 2016.

[22] L. Zhang, C. Lu, H. Xu, A. Chen, L. Li, and G. Zhou, "MMFNet: Forest Fire Smoke Detection Using Multiscale Convergence Coordinated Pyramid Network With Mixed Attention and Fast-Robust NMS," IEEE Internet of Things Journal, vol. 10, no. 20, pp. 18168-18180, 2023.

[23] J. Wang, X. Zhang, K. Jing, and C. Zhang, "Learning precise feature via self-attention and self-cooperation YOLOX for smoke detection," Expert Systems with Applications, vol. 228, p. 120330, 2023.

[24] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," arXiv preprint arXiv:2107.08430, 2021.

[25] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8759-8768.

[26] J. Zhan, Y. Hu, G. Zhou, Y. Wang, W. Cai, and L. Li, "A high-precision forest fire smoke detection approach based on ARGNet," Computers and Electronics in Agriculture, vol. 196, p. 106874, 2022.

[27] V. E. Sathishkumar, J. Cho, M. Subramanian, and O. S. Naren, "Forest fire and smoke detection using deep learning-based learning without forgetting," Fire Ecology, vol. 19, no. 1, p. 9, 2023.

[28] Y. Li, W. Zhang, Y. Liu, R. Jing, and C. Liu, "An efficient fire and smoke detection algorithm based on an end-to-end structured network," Engineering Applications of Artificial Intelligence, vol. 116, p. 105492, 2022.

[29] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," arXiv preprint arXiv:2010.04159, 2020.

[30] J. Huang, J. Zhou, H. Yang, Y. Liu, and H. Liu, "A small-target forest fire smoke detection model based on deformable transformer for end-to-end object detection," Forests, vol. 14, no. 1, p. 162, 2023.

[31] X. Zhuang, F. Tan, Z. Li, L. Li, " Image Defogging Algorithm Based On Dark Channel Priorand Optimized Auto-Color," Computer Applications and Software, vol. 38, no. 7, pp. 190-195, 2021.

[32] S. Woo, J. Park, J. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in Proceedings of the European conference on computer vision, 2018, pp. 3-19.

[33] J. Dai, "Deformable convolutional networks," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 764-773.

[34] X. Ding, X. Zhang, J. Han, and G. Ding, "Scaling up your kernels to 31x31: Revisiting large kernel design in cnns," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 11963-11975.

[35] M. Kang, C. Ting, F. Ting, and R. C. W. Phan, "ASF-YOLO: A novel YOLO model with attentional scale sequence fusion for cell instance segmentation," Image and Vision Computing, vol. 147, p. 105057, 2024.

[36] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 658-666.

[37] Y. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, and T. Tan, "Focal and efficient IOU loss for accurate bounding box regression," Neurocomputing, vol. 506, pp. 146-157, 2022.

[38] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," Information Sciences, vol. 622, pp. 178-210, 2023.