2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

DRC-YOLO: An Improved Fire Detection Algorithm Based on YOLO11

Wentao Li[1*], Cunrui Zou[2], Zhiguo Zhou[3]

[1] Northeast Normal University, Changchun Jilin 130000, China
[2] Northeast Normal University, Changchun Jilin 130000, China
[3] Northeast Normal University, Changchun Jilin 130000, China
1561825068@qq.com

Abstract. Fire detection plays an important role in safety and loss reduction and is widely used in scenarios such as forests, industrial facilities and urban environments. However, fire detection faces many challenges, including the diversity of flame appearance, dynamic and unpredictable behaviour, and the complexity of distinguishing flames from similar visual phenomena. Existing fire detection algorithms generally suffer from low detection accuracy, slow processing speed, and poor adaptability to complex backgrounds. To address these limitations, we propose a fire detection algorithm called DRC-YOLO, an enhanced model based on YOLO11. First, we replace some standard convolution blocks with dynamic convolution layers, which improves the detection accuracy of irregular fire regions while maintaining the lightweight design of the model. Second, we integrated CBAM into the detection head and enhanced it through residual connections to further enhance the network's ability to localise fire-affected regions and improve robustness. Finally, we enhanced the spatial pyramid structure by simulating large-kernel convolution operations, significantly expanding the model's receptive field while improving multi-scale feature extraction capability and maintaining computational efficiency. Extensive experiments on the M4SFWD dataset show that DRC-YOLO improves the AP by 2.6%, the AR by 2.2% and the mAP@50 by 1.8%, which are significant advantages over the baseline model.

Keywords: Fire Detection, YOLO11, Dynamic Convolution, CBAM, LSKA.

1 Introduction

Fires, whether caused by natural phenomena or human activities, pose a serious threat to economic and biological health [1]. They have the potential to escalate and spread rapidly, often leaving very little time for people to react or even escape. High temperature flames can exacerbate the danger and lead to tragic loss of life. In addition, once out of control, fires cause incalculable economic losses and pose a major threat to social production and livelihoods. Timely detection and early warning of fires is therefore an important means of ensuring the safety of people and property, reducing the losses caused by accidents and improving the overall level of emergency management [2–4].

Smoke appears earlier than other fire indicators, so early smoke detection is extremely important to improve fire control and chances of survival [5]. However, traditional smoke sensors rely on smoke spreading to a certain concentration before triggering an alarm, which not only suffers from detection delays, but is also prone to false alarms due to factors such as water vapour and soot [6]. In complex background environments, the subtle characteristics of smoke and flame are often obscured by background clutter, coupled with the influence of light, weather, wind speed and other environmental factors, further exacerbating the risk of leakage and false alarms, missing the best time to control the fire. In the complex background environment, fire detection faces even greater challenges. The characteristics of smoke and flames in the early stages of a fire are typically weak and easily masked by the surrounding background clutter [7]. For example, reflected light from urban complexes and dappled sunlight in forests can interfere with the detection of fire-related features. In addition, the effects of environmental factors such as light variations, weather conditions (e.g. fog) and wind speed significantly weaken the performance of conventional detection devices [8]. The weak robustness of traditional detection methods in dealing with these complex scenarios often leads to frequent misses and false alarms, and more efficient means of fire detection are urgently needed.

Over the past few years, deep learning-based object detection methods have emerged as innovative solutions to critical challenges in fire detection. Representative models like Fast R-CNN [9] (two-stage detection), the YOLO family [10] (single-stage detection), and Transformer-based RT-DETR [11] have demonstrated remarkable progress in general object recognition through technological breakthroughs such as feature pyramid networks and self-attention mechanisms.

However, fire detection demands unique optimization criteria: algorithms must simultaneously resist environmental interference while maintaining real-time detection speed. Current solutions grapple with inherent trade-offs - Transformer-based approaches strengthen feature representation through global modeling at the cost of computational efficiency, whereas YOLO architectures prioritize detection speed but risk missing critical details like flame edges under flickering lights or smoke patterns in foggy conditions where low contrast prevails.

Therefore, in order to improve the accuracy and robustness of the fire detection algorithm and to reduce false detections, we propose a new detection method, the DRC-YOLO model.

We designed the C3k2-DC module (Dynamic Convolution), an improved version of the C3k2 module, to enhance the baseline YOLO11 model. The C3k2 module is characterised by the flexibility to choose between C2f and C3k structures internally, as well as its adaptability to different convolution kernel sizes. Based on these features, the C3k2-DC module significantly improves the feature extraction capability of the model and better handles the dynamic shapes of flame and smoke. Compared to the standard C3k2 module, the C3k2-DC module improves the detection accuracy of irregular fire regions by dynamically adjusting convolution parameters based on input features during each operation.

The integration of an enhanced CBAM (Residual Connection-improved Convolutional Block Attention Module, RC-CBAM) in the detector head further enhances the

model's ability to independently extract and fuse small-scale feature information. This improvement significantly enhances the model's anti-interference capability against challenging backgrounds, reducing missed and false detections. By dynamically recalibrating spatial and channel features, RC-CBAM effectively highlights key fire-related information, further improving the reliability of fire detection.

The construction of the FSKA-Pool module, based on the integration of Large Separable Kernel Attention (LSKA) into the SPPF module, significantly enhances the model's ability to extract and fuse multi-scale features. This allows for more effective interaction across scales, enabling the model to fully exploit different levels of semantic information. As a result, the detection of small smoke and flame targets is improved, along with the localisation accuracy of large targets. The enhanced cross-scale feature extraction further boosts the model's adaptability and performance in detecting fire in diverse and complex scenarios.

2 Related Work

2.1 Conventional Fire Detection Methods

Conventional fire detectors are mainly based on triggering an alarm when the smoke concentration or temperature reaches a preset threshold [12]. However, due to the delayed accumulation of smoke and temperature, this approach results in a delayed response from the early fire detection system, which is particularly noticeable in large spaces or ventilated environments; this limitation increases the risk of fire spread and is significantly detrimental to evacuation and fire control.

To improve sensitivity, several new techniques have been proposed in recent years. For example, a multi-bandpass filtered fire detection system based on spectral features [13] , which achieves accurate identification of flame radiation characteristics and significantly reduces the detection delay by optimising the filter design; a rhodium complex based CO detection material [14], which detects carbon monoxide through a colorimetric thin film and improves the reuse rate; and a distributed feedback semiconductor laser (DFB LD) combined with a harmonic detection technique [15], for radiation characterisation in the flame region, which demonstrates high sensitivity in complex backgrounds. In addition, fire detection devices based on chemical molecular materials, such as reduced graphene oxide paper (RGOP-NaCl) [16], effectively reduce the risk of electrical fires due to their self-destructing properties.

Although the above new methods have improved detection accuracy and sensitivity, they are still limited by the fixed threshold triggering mechanism, resulting in a delayed response when the initial characteristics of the fire are not obvious [17]. To overcome this shortcoming, research has gradually shifted to machine vision-based fire detection techniques, aiming to achieve more real-time and dynamic fire detection and early warning, thus improving comprehensive and real-time fire detection.

2.2 Machine Vision-Based Fire Detection Methods

Machine vision-based fire detection mainly relies on machine learning and image processing techniques, where the choice of colour space is critical. The YCbCr colour space [18] is used to extract flame pixels for real-time fire detection due to its advantage of luminance and chromaticity separation. In addition, the PreVM pixel accuracy method improves detection accuracy through L0-norm regularization constraints [19], performs well under non-rigid and colour uncertainty conditions of fire images, and is suitable for low computational resource environments.

Both methods perform well in flame region segmentation, offering higher accuracy and real-time performance compared to traditional fire detection methods. However, the adaptability and generalisability of these vision-based detection techniques in complex environments is still lacking compared to deep learning methods.

2.3 Deep Learning-Based Fire Detection Methods

As the technology develops, deep learning is gradually being applied to complex fire detection tasks. Early models such as VGG, AlexNet and DenseNet, although having some detection accuracy, have limited performance in real-time and large-scale fire monitoring scenarios due to the large number of parameters, high computational overhead and overfitting problems [20]. To address these issues, models such as YOLO and Faster R-CNN have significantly reduced the number of parameters and improved the detection speed by optimising the network structure and feature extraction capability, while performing well in the accurate positioning of flames and smoke, which has become a research hotspot in the field of fire detection.

The researchers have further optimised the deep learning model. For example, by introducing the inverse convolution and null convolution layers, the resolution of the feature map and the ability to capture contextual information are improved [21], which performs well in small target detection; combining RCNN with ResNet [22] improves the recognition ability of key features of flames and smoke; and adding a small target detection (STD) layer to YOLOv5 [23], significantly improves the detection of fine fire and smoke.

Despite the excellent performance of existing deep learning models in fire detection, there are still problems of insufficient ability to detect irregular fire regions and the balance between model lightness and accuracy [24]. To address these shortcomings, we propose an improved model based on YOLO11 called DRC-YOLO.

3 Approaches

In this section, the core components of DRC-YOLO are described in detail, including the dynamic convolution layer (DC), the DC-C3k2 module, the RC-CBAM, and the FSKA-Pool module. These modules work synergistically to effectively improve the model's detection accuracy and robustness to irregular fire regions.

3.1 DRC-YOLO

Based on a comprehensive comparison of the various versions and different specifications of the YOLO model, we choose YOLO11n as the base framework of the model due to the consideration of detection speed and accuracy. As a single-stage detection algorithm, YOLO11n is lightweight, but there is still room for improving its accuracy and robustness in fire detection tasks. For this reason, we improve the model's performance while minimising the increase in computational complexity and detection latency. The overall framework of DRC-YOLO is shown in **Fig. 1.** Overall architecture of DRC-YOLO. The improvement is divided into three parts: firstly, a dynamic convolutional layer is constructed and the original C3k2 module is replaced by the C3k2-DC module in the Backbone and Neck part, thus enhancing the detection capability of irregular fire regions while maintaining a lightweight design; secondly, the residual connection-improved CBAM (RC-CBAM) is introduced in the detection head part to enhance the feature integration capability to improve the accuracy and robustness of fire detection; finally, the original SPPF module is replaced by the FSKA-Pool module to improve multi-scale feature extraction ability.

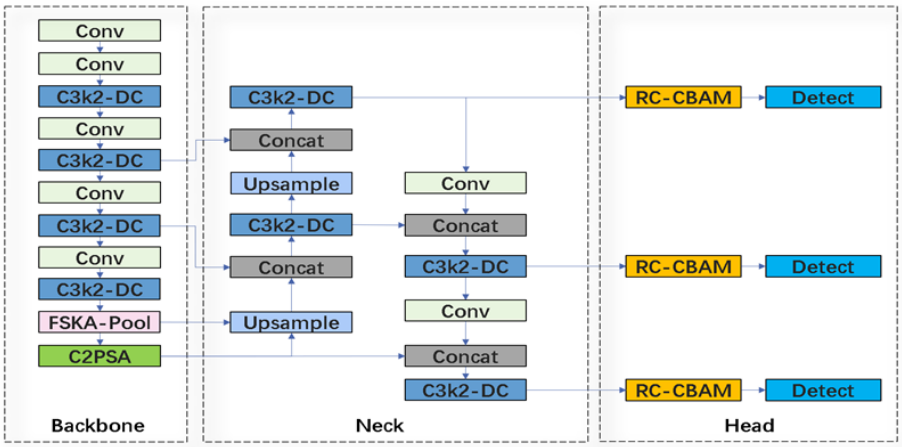

Fig. 1. Overall architecture of DRC-YOLO

3.2 Dynamic Convolution Layer and C3k2-DC

The main targets of fire detection are flames and smoke, which are highly irregular in shape and distribution. Therefore, traditional fixed shape convolutional kernels may have limitations in capturing these complex features. Dynamic convolution [25], on the other hand, allows the model to adaptively adjust the convolution kernel by introducing a weight generation mechanism based on the input features to better extract the key information in the input data. This dynamic feature provides a better option for more accurate flame and smoke detection.

The dynamic convolutional layer constructed in this paper is implemented by combining the input features to generate dynamic weights and conditional convolutional

operations. First, the weight generation module dynamically generates weight vectors from the input features, as shown in **Fig. 2.** Weight generation module of dynamic convolution , where each channel is downscaled to a 1 × 1 global feature after the adaptive average pooling operation. Subsequently, the weight vectors are obtained by the flatten operation and then mapped to [0, 1] by the sigmoid function. Next, the conditional convolution module, as shown in **Fig. 3.** Conditional convolution (example with 4 expert kernels), dynamically fuses the different convolutions by the weight vectors of the input features to achieve dynamic convolution. In (1), the contribution of each convolution kernel is determined by its weight, and the final convolution result is the result of weighting all expert convolution kernels.

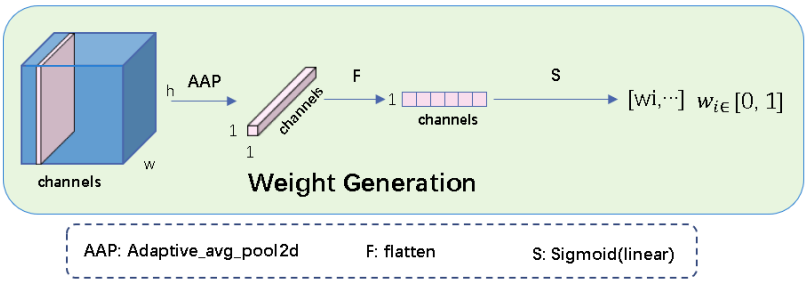

Fig. 2. Weight generation module of dynamic convolution

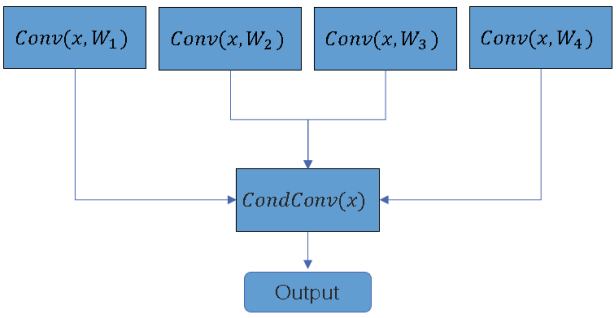

Fig. 3. Conditional convolution (example with 4 expert kernels)

$$CondCov(x) = \sum_{i=1}^{experts} (\alpha_i \times Conv(x, W_i)) \quad (1)$$

The constructed C3k2-DC module is shown in **Fig. 4.** Structure diagram of C3k2-DC , and the design idea is to improve the extraction capability of the model for features of irregular targets (e.g., flames and smoke) by introducing dynamic convolution based on the C3k2 module. The C3k2 module is characterised by a flexible structure: when the parameter C3k is true, the interior of C3k2 consists of multiple C3k modules con-

nected across phases; when the parameter C3k is false, C3k2 consists internally of multiple bottleneck modules connected across stages. The specific design to improve C3k2 is as follows:

- Bottleneck-DC module: in the Bottleneck module, this paper replaces the second convolutional layer with the constructed dynamic convolutional layer to obtain the new Bottleneck DC module.
- C3k-DC module: the C3k module is replaced by C3k-DC module, and the Bottleneck module is replaced by Bottleneck-DC module to construct the C3k-DC module.
- C3k2-DC module: the C3k module is replaced by C3k-DC module, and the Bottleneck module is replaced by Bottleneck-DC module to construct the final C3k2-DC module. The C3k2-DC module integrates the advantages of dynamic convolution and has a stronger feature extraction capability for irregular targets.

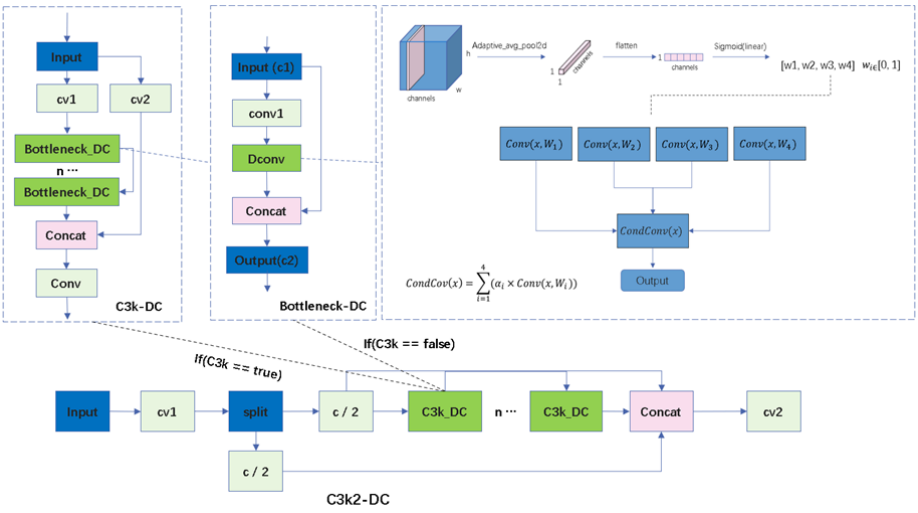

Fig. 4. Structure diagram of C3k2-DC

Previously, we proposed C3k2-DC with fused dynamic convolution, and it is worth noting that although dynamic convolution increases the number of parameters, the computational increase due to dynamic convolution is almost negligible compared to normal convolution. This is because dynamic convolution generates fusion weights from input features that linearly weight multiple sets of convolution kernels. In the inference phase, only one regular convolution operation is performed with the fused single convolution kernel, instead of multiple convolutions with multiple sets of kernels. And in the fusion process, although the weighting operation is required, it only involves weighting and accumulating the matrix by elements, and the computational complexity is much lower than the multiply-add operation of the convolution operation.

$$R_{\text{FLOPs}} = \frac{C_{\text{in}}^2 + C_{\text{in}} \cdot M + M \cdot C_{\text{out}} \cdot C_{\text{in}} \cdot K^2 + H' \cdot W' \cdot C_{\text{out}} \cdot C_{\text{in}} \cdot K^2}{H' \cdot W' \cdot C_{\text{out}} \cdot C_{\text{in}} \cdot K^2} \tag{2}$$

The conclusion can be further verified by theoretical calculations. The ratio R_{FLOPs} of the actual FLOPs of dynamic convolution compared to the ordinary convolution FLOPs is shown in (2). Since the number of the expert convolution kernels M is much smaller than the product of the height and width of the output feature map $H' \cdot W'$, which makes the computation of the weight generation and fusion stage a very small percentage, the additional overhead of dynamic convolution compared to ordinary convolution is negligible, and it can be introduced that $R_{FLOPs} \approx 1$.

3.3 RC-CBAM

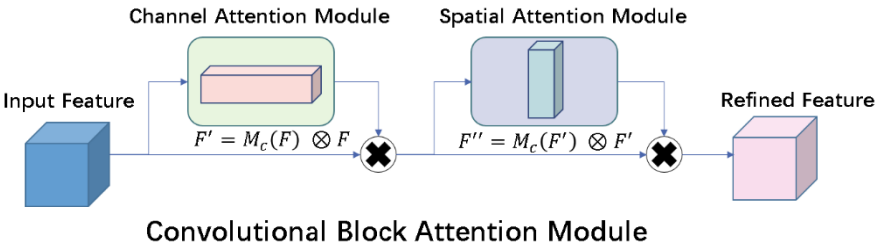

Fig. 5. Structure diagram of CBAM

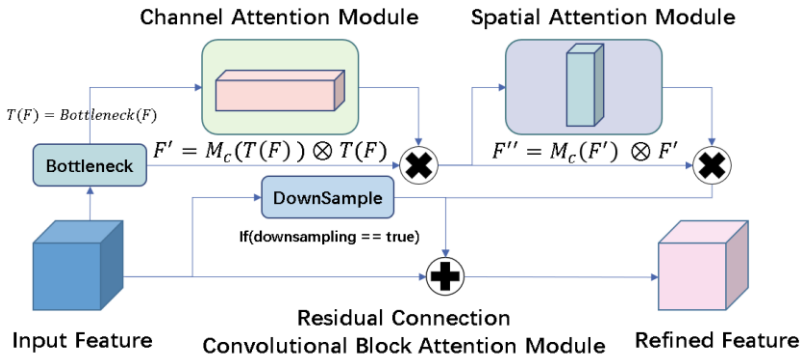

Fig. 6. Structure diagram of RC-CBAM

Convolutional Block Attention Module (CBAM) [26] is a lightweight and efficient attention mechanism that enhances the feature extraction capability. CBAM is able to judge the two independent dimensions of the attention graphs, channel and spatial, in turn, and adaptively optimise the input feature graph. As shown in **Fig. 5.** Structure diagram of CBAM, CBAM first models the relationships between different channels in the feature map through the channel attention module and assigns weights to each channel, thus highlighting information-rich channels and suppressing irrelevant channels;

second, the spatial attention module analyses the spatial dependencies within each channel and learns the key spatial regions.

Residual connection is a widely used technique in deep learning models, the core idea of which is to improve the efficiency of feature transfer by directly adding input features to output features to alleviate the problem of vanishing or exploding gradients during training. Compared to other forms of shortcut connections, residual connections are unique in that they can optimise the model more efficiently through residual functions rather than direct mapping.

In this paper, CBAM is improved with residual connection and RC-CBAM is constructed, the specific structure is shown in **Fig. 6.** Structure diagram of RC-CBAM. The input feature F is first processed by the bottleneck module to generate the transformed feature $T(F)$ (as formulated in (3)), which extracts the higher-level feature representation and effectively compresses the redundant information. Next, $T(F)$ is processed by the channel attention module M_c to obtain the feature F'(as formulated in (4)). Then, F' is further processed by the spatial attention module M_s to produce F''(as formulated in (5)). Meanwhile, as shown in (6)(7), the initial input features F are passed through the residual function to generate residuals to preserve the key information in the original features and provide supplements to the final results. Finally, F'' is summed with the residual and an activation function is applied to complete the entire RC-CBAM inference process. The design effectively fuses the deep representation of the features with the original information, improving the expressiveness and robustness of the model.

$$T(F) = Bottleneck(F) \tag{3}$$

$$F' = M_c\big(T(F)\big) \otimes T(F) \tag{4}$$

$$F'' = M_s(F') \otimes F' \tag{5}$$

$$residual = \begin{cases} downsample(F), if\ shape\_mismatch \\ F, otherwise \end{cases} \tag{6}$$

$$out = ReLU(F'' + residual) \tag{7}$$

3.4 FSKA-Pool

Spatial Pyramid Pooling Fast (SPPF) module improves on Spatial Pyramid Pooling (SPP) module by efficiently extracting multi-scale features in convolutional neural networks. It merges three successive pooling layers to fuse outputs for multi-scale fusion [27], while reducing computational complexity and increasing speed compared to SPP. However, the SPPF in the baseline model acquires features through pooling operations and lacks dynamic modelling capability, making it difficult to adapt to multi-scale target detection requirements in fire detection tasks.

Large Separable Kernel Attention (LSKA) [28] has a better ability to model long-range dependencies by decomposing a single convolutional kernel into a cascade of horizontal and vertical convolutional kernels, which reduces the optimisation difficulty and computational overhead while maintaining the ability to capture global infor-

mation. LSKA combines both spatial and channel attention mechanisms, is able to dynamically adjust the weights of features according to the contextual dependencies, thus improving the feature expression ability.

$$Z^C = \sum_{H,W} W^C_{(2r-1)\times 1} * \left(\sum_{H,W} W^C_{1\times(2r-1)} * F^C\right) \tag{8}$$

$$Z^C = \sum_{H,W} W^C_{\lfloor\frac{k}{r}\rfloor\times 1} * \left(\sum_{H,W} W^C_{1\times\lfloor\frac{k}{r}\rfloor} * Z^C\right) \tag{9}$$

$$A^C = W_{1\times 1} * Z^C \tag{10}$$

$$F^C = A^C \otimes F^C \tag{11}$$

To improve the capability of the baseline model in multi-scale feature extraction, this paper proposes to integrate LSKA into the SPPF module, and the constructed FSKA-Pool module is shown in **Fig. 7.** Structure diagram of FSKA-Pool. The input features are first spliced by three MaxPool operations to preserve the multi-scale context, and then cascaded by deep separable convolution to obtain a larger range of spatial context as well as attention to capture the global semantics and highlight the key information.

The detailed process of large kernel decomposition convolution within the FSKA-Pool module is as follows: first, the input features F^C are processed using a $1 \times (2r-1)$ horizontal convolution and a $(2r-1) \times 1$ vertical convolution, resulting in intermediate feature Z^C(as formulated in (8)). This step essentially decomposes the large kernel into depth-wise separable convolution operations. After obtaining Z^C, another large-kernel convolution with a scale of $\frac{k}{r}$ (k: kernel size, r: decomposition factor) is applied and similarly decomposed to further expand the receptive field (as formulated in (9)). Finally, a 1×1 convolution is used to generate the attention map A^C(as formulated in (10)). The attention map A^C is then multiplied element-wise with the original feature F^C to produce the final output (as formulated in (11)).

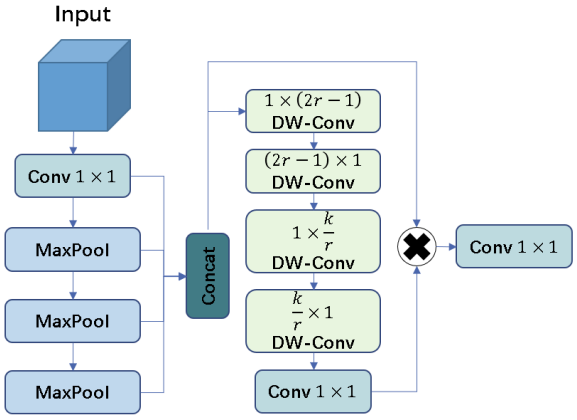

Fig. 7. Structure diagram of FSKA-Pool

4 Experiments

4.1 Datasets and experiment environment

M4SFWD: The M4SFWD synthetic wildfire dataset with multi-scenario, multi-weather, multi-light and multi-fire targets was selected for this experiment [29]. This dataset introduces complex terrain, different weather conditions and texture effects, and simulates light density at different times of day using real-time ray tracing. This diversity is very important for investigating the robustness of the model under complex conditions such as light and weather changes. In addition, M4SFWD contains 3974 images and 17763 bounding boxes covering multi-scale flame and smoke targets, providing a more challenging and representative test environment for fire detection algorithms.

The experiments were conducted under the Windows 11 operating system environment with a hardware configuration of an NVIDIA RTX 4080 laptop GPU with 12 GB of video memory. The experiments utilized Python 3.12.3, CUDA12.7 to provide GPU acceleration support, and the deep learning framework PyTorch 2.5.1. The experimental parameters were set as follows: the input image size was 640×640, the epochs were 400, and the batch size was set to 64. The pretrained and cache was set to False, and Automatic Mixed Precision (AMP) was set to True.

4.2 Comparison Experiments

Comparison of Different Attention Mechanisms

Table 1. Experimental Results of Different Attention Mechanisms

	mAP@50(%)	AP(%)	AR(%)
YOLO11n (Baseline)	86.8	82.3	80.8
YOLO11n +EMA	87.0 (+0.2)	83.4 (+1.1)	79.1 (-1.7)
YOLO11n +ECA	87.2 (+0.4)	82.4 (+0.1)	80.3 (-0.5)
YOLO11n +CBAM	87.3 (+0.5)	83.6 (+1.3)	81.1(+0.3)
YOLO11n +RC-CBAM	**87.8 (+1.0)**	**83.8 (+1.5)**	**81.5(+0.7)**

To demonstrate the effectiveness of the RC-CBAM constructed in this paper for fire detection. We conducted experiments on the M4SFWD dataset and used YOLO11n as the baseline model to compare it with EMA [30], ECA [31]and CBAM. The experimental results are shown in Table 1. The model integrated with RC-CBAM achieved improvements of 1%, 1.5% and 0.7% in mAP@50, AP(Average Precision) and AR(Average Recall) respectively compared to the baseline model. Compared to EMA and ECA, RC-CBAM improved the AP metric by 0.4% and 1.4% respectively. Unlike the trade-off observed in EMA, ECA where "AP increases but AR decreases" or "AP decreases but AR increases", RC-CBAM achieved improvements in both AP and AR metrics, demonstrating greater robustness.

Comparison of Different Detection Models

Table 2. Experimental Results of Different Detection Models

	mAP@50(%)	AP(%)	AR(%)	Params(M)	FLOPS(G)
RT-DETR	83.6	79.1	80.2	4.1	125.6
YOLOv5n	85.6	82.9	79.6	**2.2**	**5.9**
YOLOv8n	86.1	82.6	80.9	3.1	8.9
YOLO11n	86.8	82.3	80.8	2.6	6.4
DRC-YOLO	**88.6**	**84.9**	**83.0**	4.8	8.3
YOLO11s	88.1	84.6	83.0	9.4	21.6

We conducted comparative experiments on the fire detection task using the M4SFWD dataset, evaluating YOLOv5n, YOLOv8n, YOLO11n/s, RT-DETR, and our improved DRC-YOLO model. The model selection rationale includes three key considerations: First, improved models mentioned in Chapter 2 [21-23] were excluded from comparison due to unavailability of open-source code for reproduction. Second, the classical lightweight versions YOLOv5n and YOLOv8n were chosen as performance benchmarks due to their extensive deployment validation in industrial applications. Third, the novel baseline YOLO11n was introduced for its C3k2 module's flexibility and detection head's lightweight design, theoretically combining high accuracy with low computational costs. Additionally, RT-DETR is selected as a comparative model because it represents a state-of-the-art real-time detection method based on the Transformer architecture. This choice allows us to validate the performance advantages of DRC-YOLO in cross-methodology scenarios (CNN vs. Transformer) while also evaluating the model's comprehensive competitiveness in both real-time efficiency and detection accuracy.

The experimental results validate the technical rationale for selecting YOLO11n as the baseline model:

- Hardware cost: With 2.6M parameters, YOLO11n exhibits an 18% increase compared to v5n (2.2M) but remains 16.1% lower than v8n (3.1M).
- Computational efficiency: At 6.4G FLOPS, it achieves a 28% reduction compared to v8n (8.9G) while maintaining comparable efficiency to v5n (5.9G).
- Performance: The model achieves a cross-generational breakthrough with 86.8% mAP@50, surpassing v5n (85.6%, released in 2019) by 1.2% and outperforming v8n (86.1%, 2023 release) by 0.7%.

These three-dimensional advantages—controlled parameter growth, sustained computational efficiency, and leading detection accuracy—solidify YOLO11n's technical validity as a novel baseline for fire detection tasks.

The data in Table 2 demonstrates that the YOLO series comprehensively outperforms the Transformer-based RT-DETR in fire detection tasks: DRC-YOLO achieves a significant 5.0% lead in mAP@50 (88.6% vs. 83.6%) while compressing computational costs from 125.6G to 8.3G FLOPS (merely 6.6% of RT-DETR's). This "higher

accuracy, lower energy consumption" characteristic validates the adaptive superiority of the YOLO architecture in fire detection scenarios.

Compared to the baseline model YOLO11n, DRC-YOLO achieves three-dimensional improvements through C3k2-DC, RC-CBAM, and FSKA-Pool modules: mAP@50: 86.8% → 88.6% (+1.8%), AP: 82.3% → 84.9% (+2.6%), AR: 80.8% → 83.0% (+2.2%).

Notably, despite increased parameters from architectural enhancements, DRC-YOLO maintains substantial efficiency advantages over the larger YOLO11s variant. With only 51.1% of YOLO11s' parameters (4.8M vs. 9.4M) and 38.4% of its computational costs (8.3G vs. 21.6G FLOPS), DRC-YOLO surpasses YOLO11s in both AP and mAP@50 metrics.

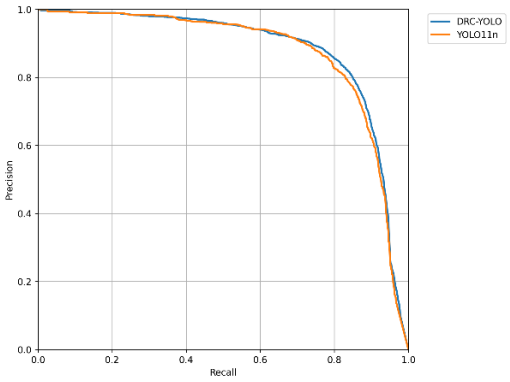

Fig. 8. Precision-recall curve

As shown in **Fig. 8.** Precision-recall curve, comparing the Precision-Recall (P-R) curve of DRC-YOLO with the baseline model YOLO11n, it can be seen that DRC-YOLO is almost always on top of the baseline model, indicating that the improved model has a higher precision for the same recall, implying that DRC-YOLO is able to detect more positive samples with a lower false detection rate. As shown in **Fig. 9.** F1-confidence curve , the F1-confidence plots of DRC-YOLO versus the baseline model YOLO11n show that the F1 scores of DRC-YOLO for different confidence thresholds are always higher than those of the baseline model, which also shows that the model has a better balance between precision and recall.

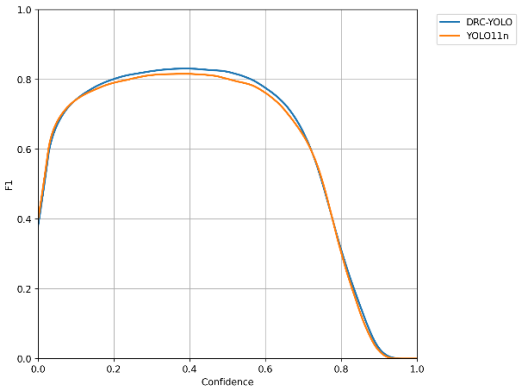

Fig. 9. F1-confidence curve

Fig. 10. Partial detection results (top two-YOLO11n, bottom two-DRC-YOLO)**Fig. 10.** Partial detection results (top two-YOLO11n, bottom two-DRC-YOLO) shows a comparison of some of the YOLO11n and DRC-YOLO detection results on the test set, where the top two are the YOLO11n detection results and the bottom two are the DRC-YOLO detection results. The test samples were deliberately selected from images with both daytime and nighttime backgrounds in order to fully evaluate the model's performance. It can be seen that in the daytime background, with the interference of the forest background and lighting conditions, YOLO11n failed to detect, while DRC-YOLO successfully detected all the targets in this image. It can also be seen that the confidence level of DRC-YOLO in detecting targets is also higher than that of YOLO11n in the nighttime background image, which also verifies the detection capability of DRC-YOLO in low light conditions.

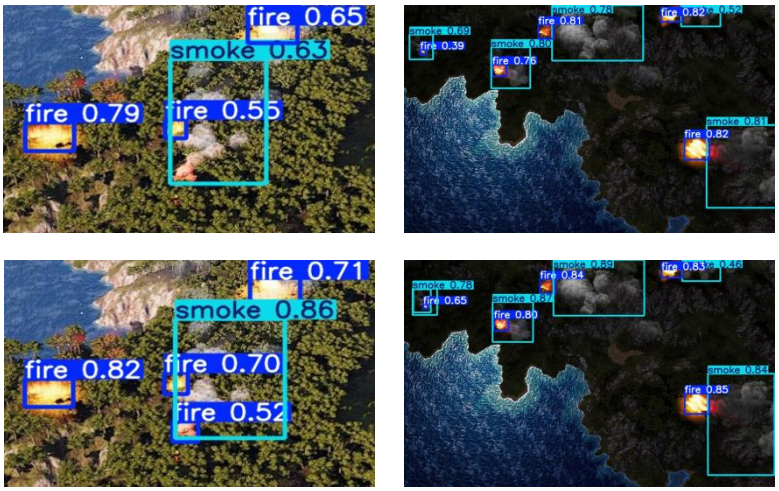

Fig. 10. Partial detection results (top two-YOLO11n, bottom two-DRC-YOLO)

4.3 Ablation Experiments

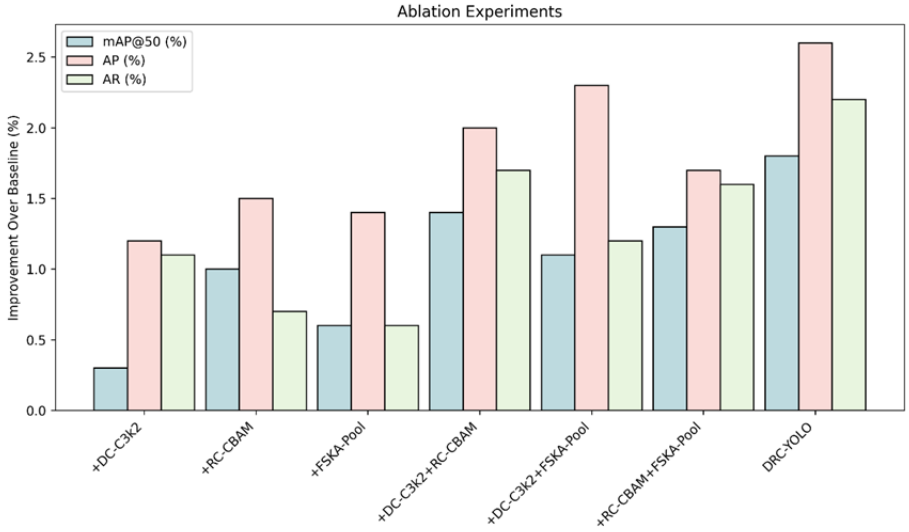

Fig. 11. Histogram for ablation experiments

Table 3. Ablation Experiments

	mAP@50(%)	AP(%)	AR(%)
YOLO11n(Baseline)	86.8	82.3	80.8
+DC-C3k2	87.1(+0.3)	83.5(+1.2)	81.9(+1.1)
+RC-CBAM	87.8(+1.0)	83.8(+1.5)	81.5(+0.7)
+FSKA-Pool	87.4(+0.6)	83.7(+1.4)	81.4(+0.6)
+DC-C3k2+RC-CBAM	88.2(+1.4)	84.3(+2.0)	82.5(+1.7)
+DC-C3k2+FSKA-Pool	87.9(+1.1)	84.6(+2.3)	82.0(+1.2)
+RC-CBAM+FSKA-Pool	88.1(+1.3)	84.0(+1.7)	82.4(+1.6)
DRC-YOLO(AllModules)	88.6(+1.8)	84.9(+2.6)	83.0(+2.2)

To investigate the effectiveness of the proposed enhancement modules, we performed progressive performance tests on each enhancement point of DRC-YOLO. Table 3 shows the experimental results of the ablation experiments, and it can be seen that the results of adding each module improved compared to the baseline model. After the introduction of DC-C3k2, mAP@50, AP and AR increased by 0.3%, 1.2% and 1.1%, respectively. After the introduction of RC-CBAM, mAP@50, AP and AR increased by 1.0%, 1.5% and 0.7%, respectively. With the addition of the FSKA-Pool module, mAP@50, AP, and AR increased by 0.6%, 1.4%, and 0.6%, respectively. Notably, the combination of RC-CBAM and FSKA-Pool delivers the highest mAP@50 improvement (+1.4%), demonstrating that the synergy between attention mechanisms and

multi-scale feature fusion is critical for location precision enhancement. Conversely, the DC-C3k2 and FSKA-Pool pairing achieves the greatest AR boost (+2.3%), validating that dynamic convolutions coupled with long-range dependency modeling effectively mitigate missed detections. DRC-YOLO, which integrates all the above modules, achieved significant performance improvements, with mAP@50 increasing by 1.8%, AP by 2.6%, and AR by 2.2%. For a more intuitive comparison, the results of the ablation experiments are presented as histogram in **Fig. 11.** Histogram for ablation experiments.

5 Conclusion

Aiming at a series of problems in the fire detection task, this paper proposes an improved model DRC-YOLO based on YOLO11. Firstly, due to the problem of fire and smoke target irregularity in the fire detection task, this paper introduces the Dynamic Convolutionally-improved C3k2 Module (DC-C3k2). Secondly, the Residual Connection-improved CBAM (RC-CBAM) is introduced to address the problems of complex background, light and weather interference in fire detection. Furthermore, the SPPF module improved with LSKA (FSKA-Pool) enhances the multi-scale feature extraction capability, effectively capturing critical features in fire detection. The experimental results on the M4SFWD dataset show that the improved DRC-YOLO has a significant improvement in mAP@50, AP and AR compared to the state-of-the-art target detection model, and have higher precision and robustness in fire detection tasks. Although the selected YOLO11n is a lightweight model among the YOLO models, while our model has achieved improved accuracy in fire detection, its lightweight design remains a potential area for further optimization. Due to the requirements of the fire detection task for the deployment of edge devices, future work will focus on further work on the lighter model while ensuring that it does not have a significant decrease in accuracy in the fire detection task.

References

1. Secilmis, N., Aksu, F.A., Dael, I., Shayea, A., El-Saleh, A.A.: Machine Learning Based Fire Detection: A Comprehensive Review and Evaluation of Classification Models. JOIV: Int. J. Inform. Visualization 7(3-2), 1982–1988 (2023)
2. Barmpoutis, P., Papaioannou, P., Dimitropoulos, K., Grammalidis, N.: A Review on Early Forest Fire Detection Systems Using Optical Remote Sensing. Sensors 20, 6442 (2020)
3. Kodur, V., Kumar, P., Rafi, M.M.: Fire Hazard in Buildings: Review, Assessment and Strategies for Improving Fire Safety. PSU Res. Rev. 4, 1–23 (2020)
4. Ivanov, M.L., Chow, W.K.: Fire Safety in Modern Indoor and Built Environment. Indoor Built Environ. 32, 3–8 (2023)
5. Gopinath, M.A., Palmur, V.V., Buddhe, M.V.: A Review Paper on Automated Fire Detection. Int. J. Innov. Eng. Res. Technol. 11(1), 1–xx (2024)
6. Hurley, M.J., Gottuk, D.T., Hall Jr., J.R., Harada, K., Kuligowski, E.D., Puchovsky, M., Torero, J.L., Watts Jr., J.M., Wieczorek, C.J.: SFPE Handbook of Fire Protection Engineering, 5th edn. Springer, New York (2016)

7. Khan, F., Xu, Z., Sun, J., Khan, F.M., Ahmed, A., Zhao, Y.: Recent Advances in Sensors for Fire Detection. Sensors 22, 3310 (2022)
8. El-afifi, M.I., Team, S.S.A.F.R., Elkelany, M.M.: Development of Fire Detection Technologies. Nile J. Commun. Comput. Sci. 7, 58–66 (2024)
9. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv:1506.01497 [cs.CV] (2016)
10. Hussain, M.: YOLOv1 to v8: Unveiling Each Variant – A Comprehensive Review of YOLO. IEEE Access, vol. 12, pp. 42816–42833 (2024)
11. Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Chen, J.: DETRs Beat YOLOs on Real-time Object Detection. arXiv:2304.08069 [cs.CV] (2024)
12. Bustos, N., Mashhadi, M., Lai-Yuen, S.K., Sarkar, S., Das, T.K.: A Systematic Literature Review on Object Detection Using Near Infrared and Thermal Images. Neurocomputing 560, 126804 (2023)
13. Bordbar, H., Alinejad, F., Conley, K., Ala-Nissila, T., Hostikka, S.: Flame Detection by Heat from the Infrared Spectrum: Optimization and Sensitivity Analysis. Fire Saf. J. 133, 103673 (2022)
14. Courbat, J., Pascu, M., Gutmacher, D., Briand, D., Wöllenstein, J., Hoefer, U., Severin, K., de Rooij, N.F.: A Colorimetric CO Sensor for Fire Detection. Procedia Eng. 25, 1329–1332 (2011)
15. Qiu, X., Wei, Y., Li, N., Guo, A., Zhang, E., Li, C., Peng, Y., Wei, J., Zang, Z.: Development of an Early Warning Fire Detection System Based on a Laser Spectroscopic Carbon Monoxide Sensor Using a 32-bit System-on-Chip. Infrared Phys. Technol. 96, 44–51 (2019)
16. Chen, G., Yuan, B., Zhan, Y., Dai, H., He, S., Chen, X.: Functionalized Graphene Paper with the Function of Fuse and Its Flame Triggered Self-Cutting Performance for Fire-Alarm Sensor Application. Mater. Chem. Phys. 252, 123292 (2020)
17. Luan, T., Zhou, S., Liu, L., Pan, W.: Tiny-Object Detection Based on Optimized YOLO-CSQ for Accurate Drone Detection in Wildfire Scenarios. Drones 8, 454 (2024)
18. Çelik, T., Demirel, H.: Fire Detection in Video Sequences Using a Generic Color Model. Fire Saf. J. 44, 147–158 (2009)
19. Yang, X., Hua, Z., Zhang, L., Fan, X., Zhang, F., Ye, Q., Fu, L.: Preferred Vector Machine for Forest Fire Detection. Pattern Recogn. 143, 109722 (2023)
20. Krishnaveni, S., Subramani, K., Sharmila, L., Sathiya, V., Maheswari, M., Priyaadarshan, B.: Enhancing Human Sight Perceptions to Optimize Machine Vision: Untangling Object Recognition Using Deep Learning Techniques. Meas. Sens. 28, 100853 (2023)
21. Zhan, J., Hu, Y., Zhou, G., Wang, Y., Cai, W., Li, L.: A High-Precision Forest Fire Smoke Detection Approach Based on ARGNet. Comput. Electron. Agric. 196, 106874 (2022)
22. Huang, P., Chen, M., Chen, K., Zhang, H., Yu, L., Liu, C.: A Combined Real-Time Intelligent Fire Detection and Forecasting Approach Through Cameras Based on Computer Vision Method. Process Saf. Environ. Prot. 164, 629–638 (2022)
23. Wu, Z., Xue, R., Li, H.: Real-Time Video Fire Detection via Modified YOLOv5 Network Model. Fire Technol. 58, 2377–2403 (2022)
24. Park, G., Lee, Y.: Wildfire Smoke Detection Enhanced by Image Augmentation with StyleGAN2-ADA for YOLOv8 and RT-DETR Models. Fire 7, 369 (2024)
25. Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., Liu, Z.: Dynamic Convolution: Attention over Convolution Kernels. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR) (2020)
26. Woo, S., Park, J., Lee, J.-Y., Kweon, I.S.: CBAM: Convolutional Block Attention Module. In: Proc. Eur. Conf. Comput. Vis. (ECCV), pp. 3–19 (2018)

27. Jocher, G., et al.: Ultralytics/Yolov5: v6.0-yolov5n ‘Nano’ Models, Roboflow Integration, Tensorflow Export, OpenCV DNN Support. Zenodo (2021)
28. Lau, K.W., et al.: Large Separable Kernel Attention: Rethinking the Large Kernel Attention Design in CNN. Expert Syst. Appl. 236, 121352 (2023)
29. Wang, G.: Multiple Scenarios, Multiple Weather Conditions, Multiple Lighting Conditions and Multiple Wildfire Objects Synthetic Forest Wildfire Dataset (M4SFWD). IEEE Dataport (2024). DOI: https://dx.doi.org/10.21227/m9kz-bw61
30. Ouyang, D., He, S., Zhan, J., Guo, H., Huang, Z., Luo, M., Zhang, G.: Efficient Multi-Scale Attention Module with Cross-Spatial Learning. Aerosp. Sci. & Ind. Shenzhen (Group) Co., Shenzhen (2020)
31. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 11534–11542 (2020)