remote sensing

MDPI

Article

FireNet: A Lightweight and Efficient Multi-Scenario Fire Object Detector

Yonghuan He [1,2], Age Sahma [1], Xu He [1], Rong Wu [2] and Rui Zhang [1,*]

[1] Faculty of Geosciences and Engineering, Southwest Jiaotong University, Chengdu 611756, China; 12221833@stu.lzjtu.edu.cn (Y.H.); shamaage@my.swjtu.edu.cn (A.S.); 2023300608@my.swjtu.edu.cn (X.H.)
[2] Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730000, China; wurong@lzjtu.edu.cn
* Correspondence: zhangrui@swjtu.edu.cn

Abstract: Fire and smoke detection technologies face challenges in complex and dynamic environments. Traditional detectors are vulnerable to background noise, lighting changes, and similar objects (e.g., clouds, steam, dust), leading to high false alarm rates. Additionally, they struggle with detecting small objects, limiting their effectiveness in early fire warnings and rapid responses. As real-time monitoring demands grow, traditional methods often fall short in smart city and drone applications. To address these issues, we propose FireNet, integrating a simplified Vision Transformer (RepViT) to enhance global feature learning while reducing computational overhead. Dynamic snake convolution (DSConv) captures fine boundary details of flames and smoke, especially in complex curved edges. A lightweight decoupled detection head optimizes classification and localization, ideal for high inter-class similarity and small targets. FireNet outperforms YOLOv8 on the Fire Scene dataset (FSD) with a mAP@0.5 of 80.2%, recall of 78.4%, and precision of 82.6%, with an inference time of 26.7 ms. It also excels on the FSD dataset, addressing current fire detection challenges.

Keywords: fire and smoke detection; RepVit; DSConv; decoupled detection head; YOLO

check for updates

Citation: He, Y.; Sahma, A.; He, X.; Wu, R.; Zhang, R. FireNet: A Lightweight and Efficient Multi-Scenario Fire Object Detector. *Remote Sens.* **2024**, *16*, 4112. https://doi.org/10.3390/rs16214112

Academic Editor: Melanie Vanderhoof

Received: 12 September 2024
Revised: 21 October 2024
Accepted: 23 October 2024
Published: 4 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Fire and smoke detection has always been a hot research topic in multiple fields. However, there are still some issues threatening human life and property safety. Common fire scenarios include buildings [1], transportation [2], forests [3], etc., which are major hazards to human life safety and the natural environment. They also cause immeasurable economic losses and environmental damage to society [4]. Currently, the focus on reducing fire risks should not only be on prevention management but also on timely detection, prevention, and early warning. Generally, fires in the early stages are not large and hard to detect, but a large amount of smoke is produced in the initial phase. Therefore, the method of detecting fire smoke images has become a key direction in fire monitoring research [5]. At present, most monitoring methods still rely on manual observation or fire alarms. However, manual observation lacks timeliness and coverage, and traditional point smoke detectors can only detect fire signals when smoke particles enter the sensing chamber. Signal detection takes time, especially in larger spaces and outdoor scenes, where detection takes even longer. Therefore, monitoring and preventing the occurrence of fires has always been a focal point of attention [6].

In recent years, UAV (Unmanned Aerial Vehicle) power inspection has attracted attention due to its compact size and cost-effectiveness [7]. It has performed well in disaster detection and post-disaster search and rescue. Compared to traditional manual search methods, UAV approaches are not restricted by post-disaster ground traffic and communications, allowing for rapid, large-area searches and assessments, with good energy efficiency [8,9]. However, the operational characteristics of UAVs, such as high-altitude cruising and variable flight heights, often result in captured images featuring drastic

changes in target scale, complex backgrounds, and the presence of numerous small, dense targets [10,11]. Specifically, the perspective of UAV images differs from that of traditional cameras, leading to diversity in objects' angles, scales, and shapes in the images. These changes increase the complexity of the target detection task. Moreover, UAV-captured images have a wider field of view, especially under outdoor conditions with complex background elevations, making the target detection task even more complicated. Lastly, due to the limitations of the device itself and the height of operation, UAV-captured images may have low resolution, resulting in the loss of detailed features and information, which leads to a decrease in detection accuracy. Therefore, target detection in UAV images remains a challenging task.

Currently, the more mainstream methods of fire monitoring can be primarily categorized into three types: (1) Manual monitoring, which includes setting up fire watch towers in forest areas and patrols by rangers. (2) Traditional methods, mainly identifying smoke and flames in collected digital images based on color [12], edge contours [13], and texture [14]. However, as the color, contours, and textures of flames and smoke dynamically change with the development of the fire, this increases the difficulty of recognition. (3) Methods based on Convolutional Neural Networks (CNNs) in deep learning, which include two-stage detection algorithms that first obtain many candidate boxes from sample processing, then extract features from samples, complete classification tasks through convolutional neural networks, and finally make target frames precise through post-processing operations (R-CNNs [15], Faster R-CNN [16]) and one-stage detection algorithms (YOLO series [17], SSD [18]).

In recent years, deep learning methods (such as CNNs) have gradually matured in target detection technology combined with UAVs. CNNs, with their characteristics of local perception and weight sharing, are commonly used to extract image features and are widely applied in fire smoke detection tasks. Muhammad et al. [19] introduced an early fire detection framework based on CNNs and IoMT, using AlexNet as the baseline architecture. This method focuses on adjusting the fully connected layer to adapt to specific datasets, aiming to reduce false positives. Obviously, this approach lacks generalizability, performs well only in specific datasets, and still faces challenges in false positives and multi-scenario application capabilities. Barmpoutis et al. [20] utilized the Faster R-CNN model to identify candidate fire areas and adopted multi-dimensional texture analysis to improve detection accuracy. This method, however, increases computational and parameter requirements, which is not conducive to real-time tasks. Zhao et al. [21] proposed a real-time video fire detection method based on the YOLOv5 as the baseline model. Due to the efficiency of the YOLO series, the detection speed was resolved, and the accuracy of fire detection was further improved. Li et al. [22] proposed a Bidirectional Transposed FPN (BCMNet) for cross-layer extraction structure and multi-scale downsampling network with YOLOv3 as the baseline model to improve the speed and accuracy of wildfire smoke detection. It can bidirectionally fuse shallow visual features and deep semantic features at corresponding scales, emphasizing the feature information flow between smoke feature maps of different resolutions. It is evident that the YOLO series usually performs well in such real-time monitoring tasks. Although CNNs have initially realized real-time monitoring of fires, we find that when facing remote sensing images under UAVs, they encounter complex background interference, issues with targets being too small, and accompany varying target shapes, significantly affecting detection accuracy. As a deep learning model, transformers have made revolutionary progress in Natural Language Processing (NLP) tasks. The core of the transformer model is the self-attention mechanism, which allows the model to focus on different parts of the sequence data, thus capturing complex dependencies within the data. Dosovitskiy et al. [23] first proposed the Vision Transformer (ViT), a deep learning model that directly applies the transformer architecture to image recognition tasks to leverage the powerful features of transformers and achieve long-distance capture of dependencies. Sun et al. [24] addressed the issue of slow iterative convergence caused by the Hungarian loss and transformer cross-attention mechanism by combining CNNs with ViTs, further

enhancing the performance of transformer models in object detection tasks. Integrating the concept of ViT into the YOLO model can further improve the insufficient feature extraction capability of YOLO. Dai et al. [25] added Transformer blocks to the backbone of YOLOv5, addressing the problem of insufficient global information capture. However, transformers usually sacrifice computational complexity at the cost of improving feature extraction capability, which is fatal in real-time monitoring tasks. Existing research continues to overcome this shortcoming, for instance, Li et al. [26] effectively minimized the network layers and parameters by using the development of customized convolutional blocks, thereby achieving smaller model sizes and higher classification speeds. Huang et al. [27] replaced the backbone network part of YOLOX-L with GhostNet to reduce the overall network parameters. However, it is evident that there is still a lack of balance between detection accuracy and model complexity.

The current fire detection systems based on CNNs have achieved significant success. However, due to the uniqueness of fire detection tasks, the following challenges still exist:

1. In complex environments, shape and color are significant features, which are also the most important factors in feature fusion in traditional detection methods [28]. For example, in the perspective of drone-collected images, the features of smoke and clouds have a high degree of similarity.
2. Flames and smoke, due to their strong shape variability, have irregular shapes, leading to the loss of edges in detection. The edges of smoke from a drone's perspective can easily blend with the air, resulting in incomplete recognition without global correlation.
3. In this paper's pursuit of high accuracy, the network depth increases, which may lead to reduced detection speed.
4. In existing samples, there is an imbalance of positive and negative samples. Since the ratio of flame and smoke pixels in the whole sample image is too small compared to the background, the background (negative samples) usually far exceeds the foreground targets (positive samples). This causes the model to be overly sensitive to frequently occurring categories, concentrating too much attention on the background.

To address these challenges, this study proposes a FireNet network model, a real-time detection model for aerial fire in multi-scene drones. Moreover, we have built a multi-scene (buildings, indoors, cigarette butts, forests, traffic, etc.) fire drone dataset. The main contributions of this paper are as follows:

1. Traditional ViT structures can globally access information, helping the model to capture long-range dependencies of fire targets and edges of smoke. However, they usually have a large number of parameters. We created a lightweight feature extraction backbone, which independently completes the feature extraction work in the backbone through Revisiting Mobile CNN From ViT Perspective (RepViT), showing superior performance and lower latency compared to lightweight CNNs. This improvement is generally attributed to the multi-head self-attention module, which enables the model to learn global representations.
2. In response to the irregular boundary features of flames and smoke and their variable contours, Tao et al. [29] addressed the fragile and variable boundaries of smoke by breaking the fixed geometric structure of standard convolutions, considering both representative features and local weak features for feature complementarity. However, the proposed variable convolution allows the network to freely learn geometric variations, leading to perceptual region drift, especially on some thinner smoke structures. We have designed a dynamic snake-like convolutional structure for the neck of the network, considering the features of partial fine structures and supplementing the free learning process with constraints, specifically enhancing the perception of fine structures based on the existing variable convolution.
3. To address challenges such as the high similarity of small targets under and background noise interference, we designed a lightweight detection head by integrating channel and spatial attention mechanisms. It adaptively selects the most representa-

tive image features and concentrates them, thereby improving the network's classification accuracy. At the same time, a multi-scale feature extraction strategy effectively extracts features of different scales and levels, helping the detector achieve stronger positioning and classification performance while avoiding an increase in model computational complexity.

4. This paper collected flame and smoke images from multiple scenarios to construct the Fire and Smoke (FAS) dataset. In addition to common fire-prone scenarios, the dataset also includes non-fire scenes, such as burning cigarettes and open fires during picnics. Notably, this paper incorporated a large number of drone-collected images into the dataset to test small target fire scenes. Furthermore, this paper compared FireNet with state-of-the-art (SOTA) algorithms on the FAS dataset and additionally conducted comparisons on the public fire smoke dataset. The results demonstrate that the proposed model performs exceptionally well in multi-scenario fire detection. (We will upload this paper's source code1) 1 [Online]. Available: https://github.com/DC9874/FireNet, accessed on 22 October 2024.

2. Related Work

In recent years, with the development of the YOLO series, many versions have been produced, and the models have shown good application prospects in disaster monitoring. The overall idea of the YOLO series divides the network into three parts: the backbone, the neck, and the detection head, each consisting of different modules. YOLOv8 is the latest SOTA model in the YOLO series. Compared to the main component c3 of YOLOv5 [30], YOLOv8's main component c2f provides a richer gradient flow than the C3 structure, reduces computational redundancy through optimized computation paths, and decreases the model's computational requirements. As a result, the model maintains high accuracy while running faster, making it more suitable for use on resource-limited devices. We have drawn inspiration from the "trisected" structural concept of the YOLO series and designed a variant of the c2f from YOLOv8 in the neck. The FireNet structure is shown in Figure 1. The following sections will specifically introduce this paper's current related work.

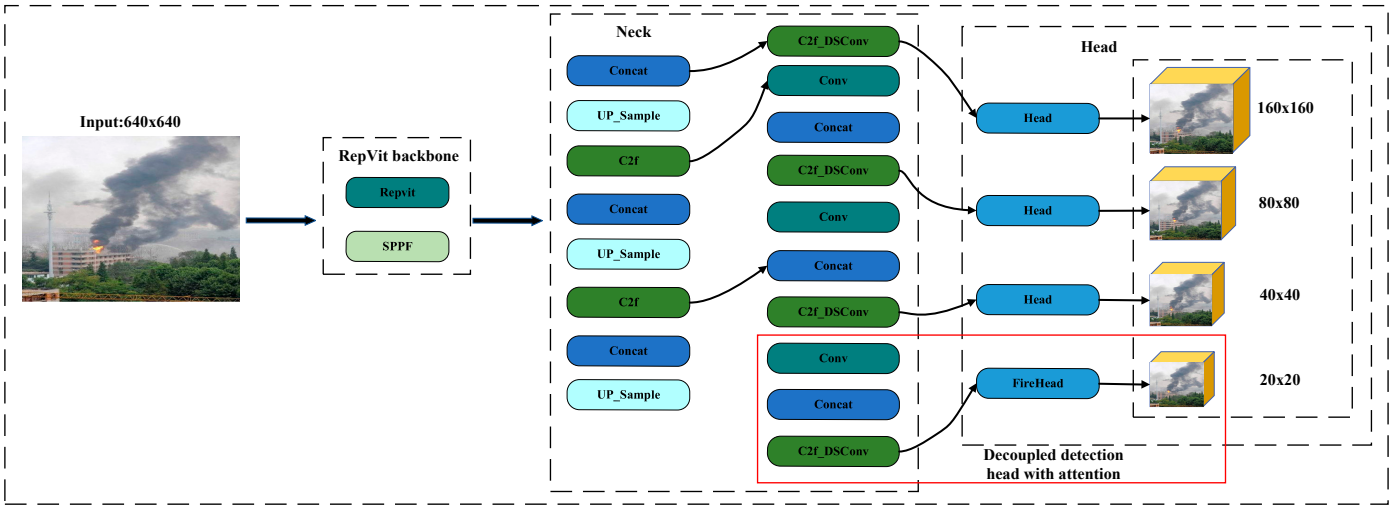


Figure 1. "Trisected" FireNet structure.

2.1. Dataset

The effective evaluation of fire detection model performance depends on a high-quality dataset [28]. We obtained images of some uncommon fire scenarios (such as forest fires, building fires, etc.) through web crawlers and multiple sources. Additionally, we collected some data in northwestern China using drones and handheld shooting equipment to create the FAS dataset. The specific parameters of the dataset are presented in Table 1. We divided this paper's FAS dataset into training, validation, and test sets with a distribution of 8:1:1,

which includes images of multiple scenarios where fires occur as well as common fire-prone locations as background images. This dataset will serve as an important dataset for validating different network models in this paper.

Table 1. Data set composition.

Dataset	Train	Valid	Test	Sum
Quantity	6028	753	753	7534

The FAS dataset we constructed (as shown in Figure 2) encompasses various scenes of flames and smoke, including urban fires, forest fires, traffic fires, building fires, campfires, industrial smoke, and cigarette smoke. The experiments in this paper not only utilized fire samples but also included some civilian scenarios. From a geographical perspective, these scenes are divided into urban and natural areas. In urban areas, some sources of smoke and flames, such as cigarette butts and small-scale ground fires caused by campfires or flammable materials, typically burn at lower temperatures and cover small areas, leading to fewer available feature pixels, which poses challenges for feature learning. Situations like urban traffic fires and building fires present complex backgrounds, and the high similarity between the target features and the environment also increases detection difficulty. Additionally, we included instances of fires caused by grasslands, forests, and shrubs. These fires are usually larger in scale and are accompanied by thick smoke. In such recognition tasks, timeliness is of utmost importance, and there is generally a higher demand for the model's recognition speed. These diverse scenarios aim to challenge and validate the robustness and efficiency of fire detection models under various conditions.

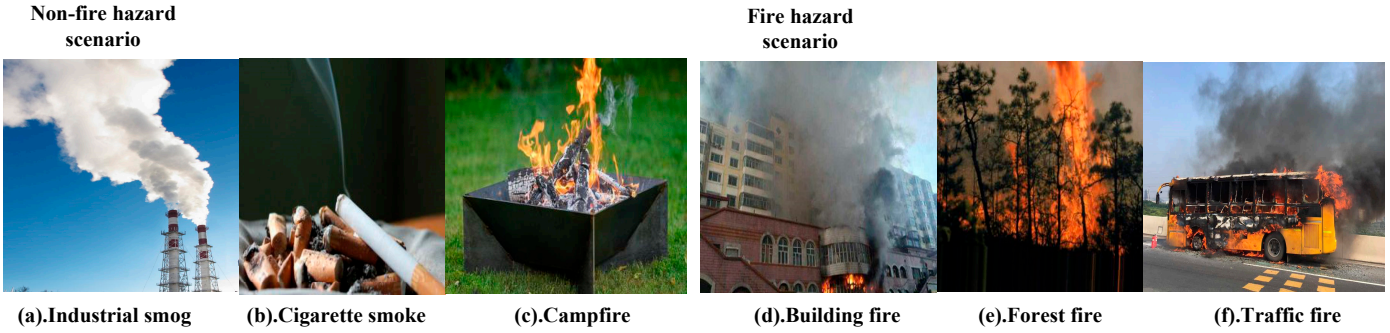


Figure 2. Multi-scenario sample images.

To test the generalizability and superiority of the model, we utilized the public dataset, Fire Smoke Dataset, which contains 23,730 images. The Fire Smoke Dataset includes a richer variety of fire scenarios and a large number of feature-similar non-fire scenes. This provides a good performance benchmark for assessing the model's ability to generalize and accurately extract features across diverse and challenging environments. The inclusion of numerous scenarios that mimic or resemble fire-related environments without actual fires tests the model's capacity to distinguish between genuine fire signals and misleading conditions, thereby evaluating its precision and reliability in real-world applications.

2.2. Reparameterization Vision Transformer Block (RepViTBlock)

In recent research, methods combining Natural Language Processing(NLP) with Computer Vision(CV) have been widely applied in disaster strategy directions. Liu and others [31] proposed a model using the Swin Transformer as the backbone network for feature extraction. In fire detection, the ViT can globally capture information in the image, including the overall edge shapes and distribution of flames and smoke. Through the self-attention mechanism, ViT can directly capture global information in the input sequence without being limited by the size of the convolution kernel, enabling it to better understand

the edge features of flames and smoke. It is essential to consider the actual fire detection scenarios. In urgent fire scenarios, efficiently processing a large amount of fire photo information is indispensable, and high detection accuracy often comes from deeper network architectures. The fact that drones are used for fire detection presents a significant challenge to the ViT structure due to its generally large number of parameters, especially when processing high-resolution images, leading to high computational costs. This means that more computing resources and longer inference time are needed in practical applications, thus limiting its application in real-time or efficient fire monitoring systems. Traditional lightweight models like MobileNet [32] and AlexNet [33] use techniques such as depth-wise separable convolutions, reducing the size of kernels, decreasing network depth, global average pooling, and batch normalization to reduce computational complexity and parameter count. However, compared to ViT, these networks might not provide sufficient performance in handling complex tasks such as object detection and semantic segmentation.

In this study, we adopted an innovative feature extraction framework named RepViT, which combines the advantages of efficient lightweight Vision Transformers and traditional lightweight CNNs. In this paper's model structure design, we took strategic measures by moving deep convolution layers to higher levels of the network for early processing of deep convolution features. Furthermore, we optimized the layout of the Squeeze-and-Excitation (SE) attention mechanism from MobileNetV3 (Figure 3a) by placing it before deep filters to enhance the model's capability to facilitate spatial information exchange. This layout adjustment optimizes the efficiency of the SE mechanism in assessing the importance of each channel in the feature maps, thereby improving the representation and recognition performance of spatial features in forest fire detection scenarios. Moreover, to enhance model performance, this study introduced a structural reparameterization technique to realize the multi-branch topology of deep filters during training. The application of this technique allowed the token mixer and channel mixer to be effectively separated, thereby constructing the RepViTBlock (Figure 3b) module. In the model inference phase, the multi-branch topology of the token mixer is simplified into a single deep convolution operation as shown in Figure 3c, significantly reducing computational and memory requirements and reducing latency, thereby accelerating the model's inference speed. This optimization makes the RepViT framework particularly suited for applications on mobile devices such as drones, providing an efficient and practical solution for fire detection.

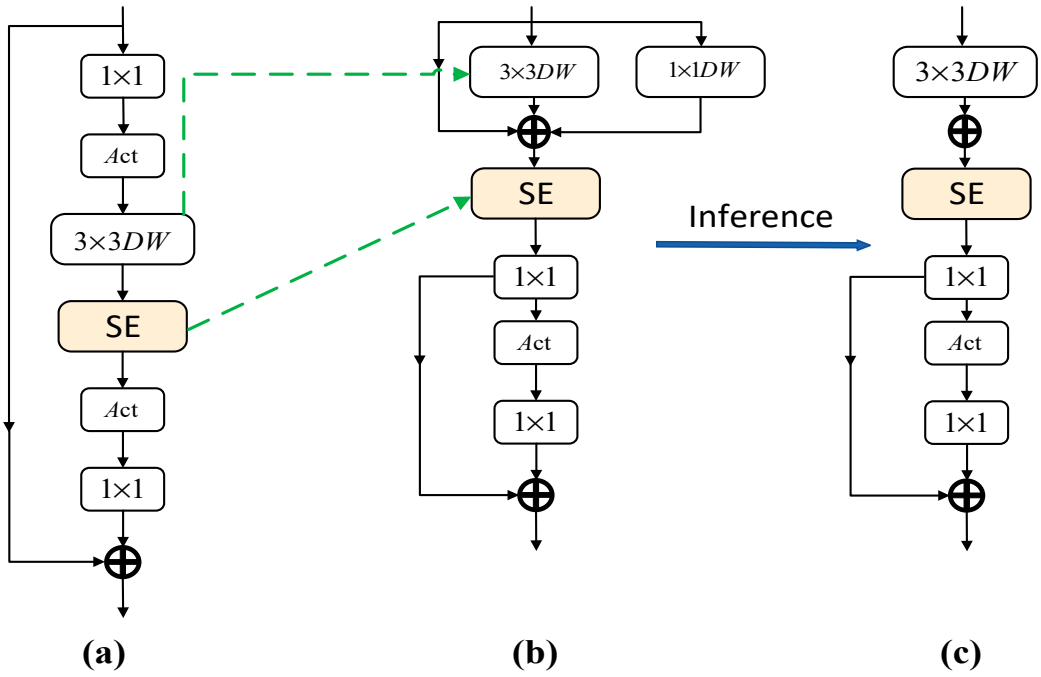


Figure 3. (**a**) MobileNetV3 block. (**b**) RepViTSEBlock. (**c**) RepViTSEBlock inference phase structure.

To enhance the performance of the model and its adaptability to mobile devices, we optimized the following four stages:

1. ViTs segment the input image into non-overlapping small patches, which is equivalent to using non-overlapping convolution operations with large kernel sizes and large strides. However, research in [34] has found that this approach might cause ViTs to fall short in terms of optimization performance and sensitivity to training strategies. The study suggests using a small number of stacked 3×3 convolution layers with a stride of 2 as the stem, a method known as "early convolutions". This approach has also been widely used in lightweight models subsequently.
2. In traditional ViTs, spatial downsampling is typically achieved through a separate merging layer (patch merging layer) using convolutions with a kernel size of 4 and a stride of 2. This design, while mitigating information loss due to reduced resolution, also increases network depth. We, on the other hand, form a feedforward network (FFN) by connecting the input and output of two 1×1 convolutions through residual connections. We also added the RepViTBlock module to further deepen the downsampling layers, mitigating information loss in the spatial dimension, not only improving the model's accuracy but also maintaining lower latency, adapting to this paper's proposed resource-limited drone detection system.
3. We employed a global average pooling layer and a linear layer as the classifier. By using a simple classifier and adjusting the stage ratio and depth of the network, we can effectively balance the performance and latency of lightweight ViTs, making them more suitable for applications on mobile devices while also enhancing the model's accuracy.
4. Overall stage ratio, we found that different numbers of blocks in the four stages of the model have varying impacts on the model's performance. Hou and others [35] indicate that more aggressive stage ratios and deeper layouts perform better for smaller models. The Conv2Former's Conv2Former-T and Conv2Former-S adopted series ratios of 1:1:4:1 and 1:1:8:1, respectively. Whereas [35] shows that using an optimized stage ratio of 1:1:7:1 for the network achieves a deeper layout. We have illustrated the framework of RepViT's four structural components in Figure 4.

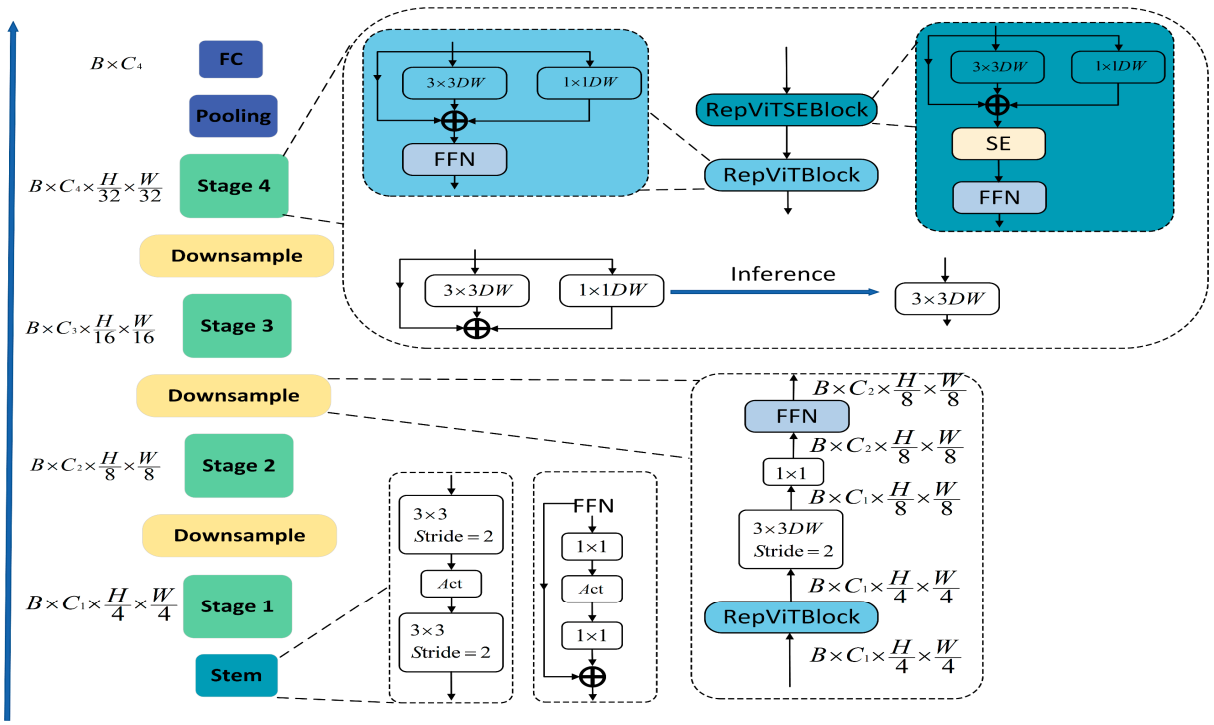


Figure 4. RepViT's four-phase structure.

2.3. Introduction to the Neck Module of Dynamic Snake Convolution (DSConv)

In the previous section, we mentioned that in fire scenarios, flames and smoke make up only a small part of the entire input image, resulting in limited pixels available for feature extraction. This requires the model to have a strong capability to extract subtle feature variations, as flames and smoke typically do not have fixed presence features. These problems demonstrate that a crucial aspect of fire detection tasks lies in the extraction of key features. Dai et al. [36] introduced a deformable convolution that, compared to standard convolutions, can adjust the shape of the convolution kernels during the feature learning process to focus on the fragile and variable shapes of flames and smoke. Considering the small proportion of core structural features in postures and the risk of convolution kernels deviating from posture structures, the model inevitably loses awareness of the posture structure. To address this issue, we introduced a dynamic snake convolution module [35], which effectively extracts key features in fire detection under certain constraints while preserving the target structure.

Our objective is to extract the local characteristics of tubular structures. Initially, we assume a coordinate system for the two-dimensional convolution, denoted as K, with the central coordinate being the most critical, represented as $K_i = (x_i, y_i)$ Within this framework, we employ a 3×3 convolution kernel with a dilation factor of 1. K is represented as (1) [33].

$$K = \begin{Bmatrix} (x-1, y-1), & (x, y-1), & (x+1, y-1) \\ (x-1, y), & (x, y), & (x+1, y) \\ (x-1, y+1), & (x, y+1), & (x+1, y+1) \end{Bmatrix} \tag{1}$$

To make the convolution kernels more adaptable to complex geometric features, especially for the ever-changing postures of flames and smoke, this paper proposes the introduction of deformable offsets Δ. This method allows the model to have greater flexibility when dealing with these dynamic geometric changes. However, one challenge faced by this approach is that if the learned deformable offsets are random, the perceptive region may deviate from the target. This is particularly problematic when dealing with scenes that have complex edge features, as it could result in the model failing to accurately focus on the target. To overcome this challenge, this paper introduces an iterative strategy. As shown in Figure 5, this strategy sequentially matches each target to an observable position, ensuring through this method that the model continuously focuses on the features without excessively expanding the perceptive area due to too large deformable offsets. Through this iterative matching strategy, the model can more precisely locate and track targets, even in the presence of complex geometric features and dynamic changes. The key to this method is that controlling the variation of the offsets, it ensures the proper adjustment of the perceptive region, thereby enhancing the model's adaptability and accuracy to complex geometric features.

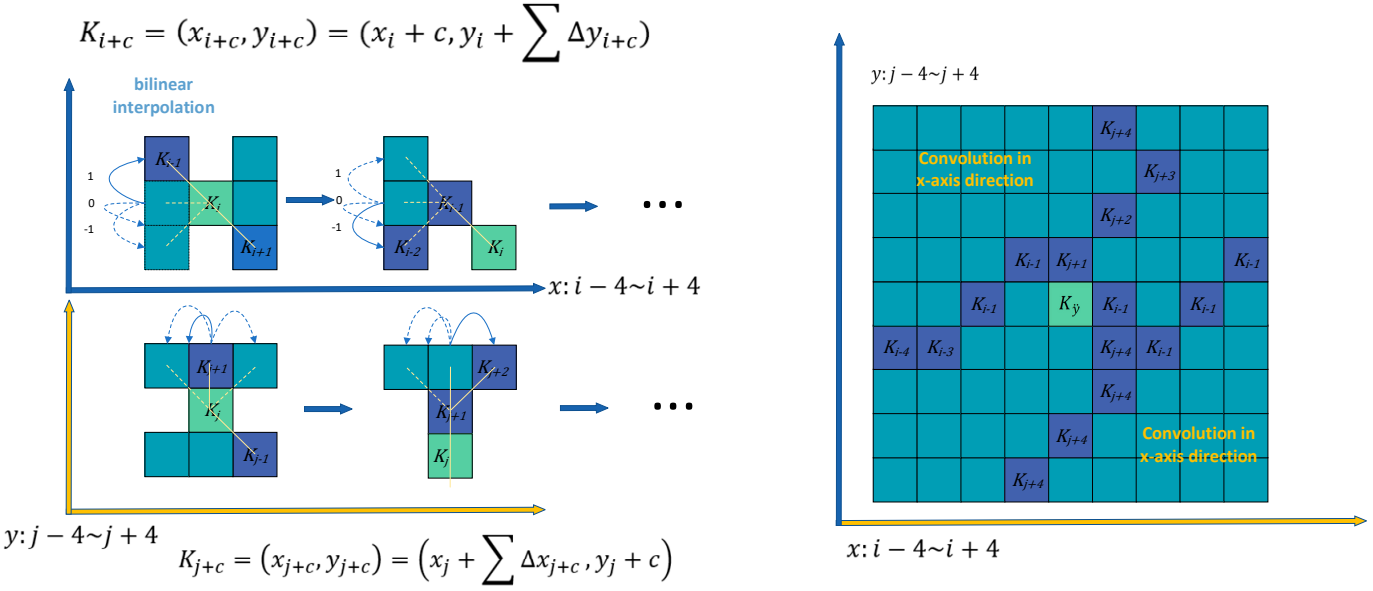
The standard convolution kernel is linearized along both the x-axis and y-axis. The kernel size is set to 9, and the position of each grid point in K is expressed as $K_{(i\pm c)} = (x_{(i\pm c)}, y_{(i\pm c)})$ [37]. Here, $c = \{0, 1, 2, 3, 4\}$ represents the horizontal distance of the grid points from the central grid. The selection of grid positions in K is a cumulative process, starting from the central position K_i. Each adjacent grid position is determined relative to the previous one, with an offset $\Delta = \{\delta | \delta \in [-1, 1]\}$. The total sum of these offsets is denoted by Σ. Equation (2) is modified in the y-axis direction as follows: in summary, these parameters and notations are used to describe and control the grid positions of the convolution kernel and their relative movements. As depicted in Figure 5a is the graphical representation of DSConv coordinate calculation, and Figure 5b is the acceptance field of DSConv.

$$K_{itc} = \begin{Bmatrix} (x_{itc}, y_{itc}) = \left(x_i + c, y_i + \sum_i^{i+c} \Delta y\right) \\ (x_{i-c}, y_{i-c}) = \left(x_i - c, y_i - \sum_{i-c}^{i} \Delta y\right) \end{Bmatrix} \tag{2}$$

In the context of A, the process of bilinear interpolation is undertaken. Within this framework, the symbol K is utilized to indicate positions that are fractional, where as K' signifies those positions that are quantified in integer space. Moreover, B is designated as the kernel that facilitates the bilinear interpolation. It is important to note that the formulation for K is meticulously outlined in what is referred to as Equation (3).

$$K = \sum_{K'} B(K', K) \cdot K' \tag{3}$$

We previously mentioned that the main component c2f of YOLOv8 provides richer gradient flow than the C3 structure used in models like YOLOv5, playing a crucial role in utilizing features of various scales and integrating contextual information to enhance detection accuracy. We selected c2f as the baseline component for the neck structure, but c2f does not perform well when dealing with the indeterminate target structures and fragile boundaries mentioned earlier. Moreover, the fixed nature of the standard convolution kernels used in c2f limits the network's receptive field to only capturing local object information. Therefore, common issues include missing detections when dealing with multi-scale objects, multiple objects, or occluded objects. Additionally, the presence of noise or interference increases the likelihood of false positives or missed detections, especially for small objects and variable structures.



(a).Visualization of dynamic snake convolution coordinate calculation.

(b).Receptive field of dynamic snake convolution.

Figure 5. Working process of dynamic snake convolution.

To address this problem, the DSConv structure replaces the original standard convolution structure in the C2f module. As can be seen in Figure 6, the overall structure of DSConv in (b) presents a thin-wall shape, which is crucial for focusing on the thin-wall tubular structures of flames and smoke edges. In (a), traditional standard convolutions operate on a fixed grid R, where each sampling point is weighted by the convolution kernel, which does not fit the unique structures in this paper's detection targets.

Our designed C2f_DSConv module (Figure 7) adjusts the number of channels in the input feature map with a 1×1 convolution kernel and adopts a technique different from 1×1 convolution—feature segmentation—to further process the feature map. This strategy, by repeatedly applying the DSConv module, not only expands the model's perceptual range but also enhances the diversity of information flow by adding skip connections.

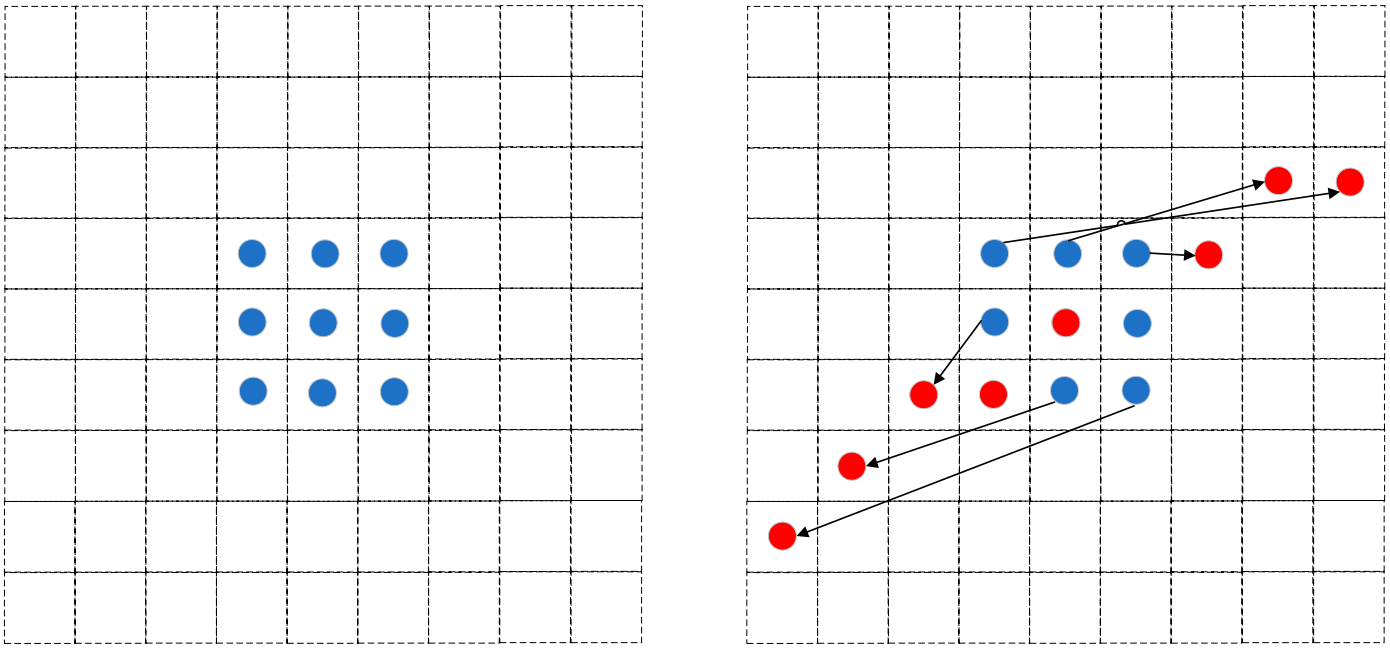


Figure 6. Sampling comparison between standard convolution and Dynamic Snake Convolution.

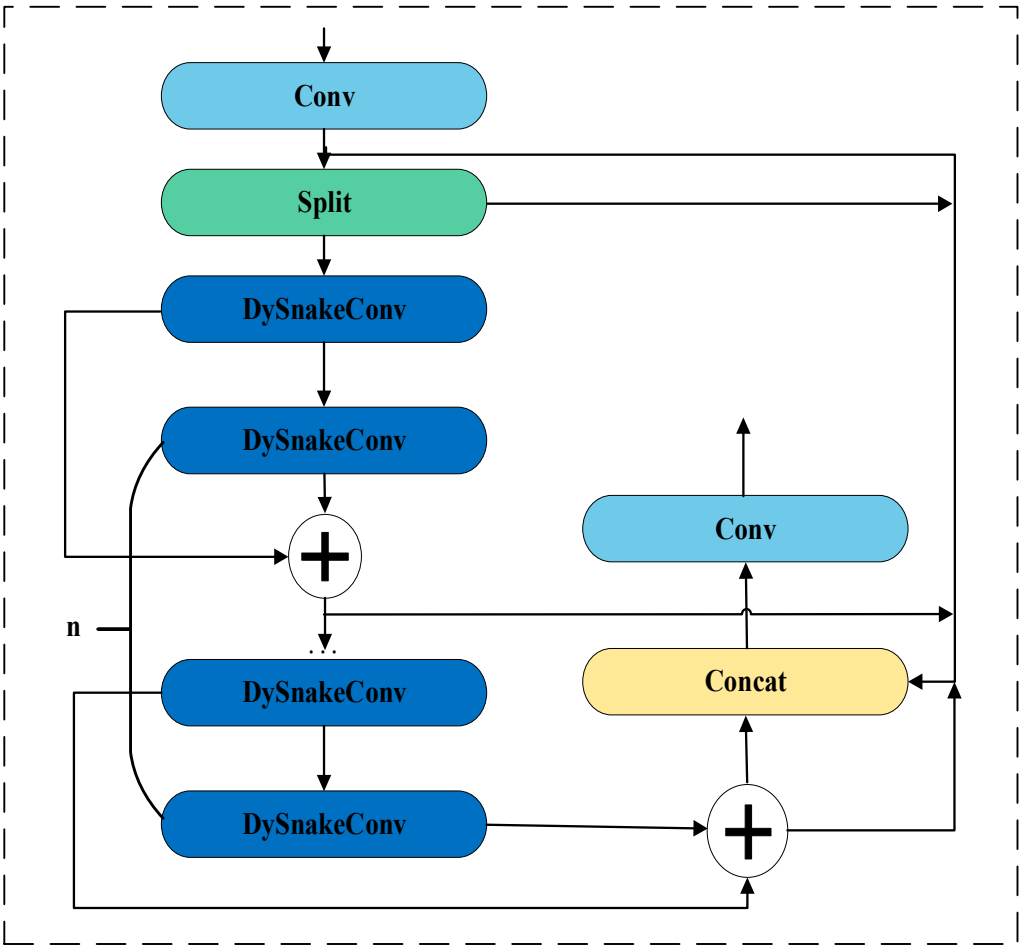


Figure 7. C2f_Dsconv structure.

Such architectural optimization is particularly effective in enhancing the model's recognition of edge details and complex textures because it can produce a more detailed gradient distribution, thereby capturing broader and more layered information during the feature extraction process. This design allows the network to exhibit greater flexibility and accuracy when dealing with objects with complex boundaries and textures, helping the model to have greater flexibility when dealing with dynamically changing geometric features of smoke and fire. Dang and others in [38] mentioned that generally, using such deformable convolutions can aid in feature extraction, but it should be noted that excessive use might not linearly improve performance and might even reduce speed.

2.4. Decoupled Detection Head with Attention

In this study, we found that in samples captured by drones, some targets present a significant challenge for recognition due to their relatively small pixel proportion (small targets) within complex backgrounds occupying a larger proportion. To address this challenge, we designed a small-scale detection head, Firehead, equipped with an attention mechanism.

2.4.1. CBAM Attention Mechanism

The essence of the attention mechanism is to assign more weight to the features of interest points, thereby filtering out the background and noise, enabling the model to better focus on the pixels in the feature areas. We introduce a Convolutional Block Attention Module (CBAM) to counteract interfering factors. CBAM is a simple and effective feed-forward attention module. The CBAM structure consists of a Channel Attention Module (CAM) and a Spatial Attention Module (SAM) [39]. The output of a convolutional layer passes through the CAM and SAM modules. First, an intermediate feature map $F \in R(C \times H \times W)$ is produced. Then, 1D mapping $Mc \in R(C \times 1 \times 1)$ and 2D mapping $Ms \in R(1 \times H \times W)$ are created by CAM and SAM, respectively. The entire process is summarized in the following formula, where $\otimes$ denotes element-wise multiplication, CA$(\cdot)$ represents the operation of the channel attention module, and SA$(\cdot)$ represents the operation of the spatial attention module. F'' is the final output.

$$F' = M_c(F) \otimes F \tag{4}$$

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

The structure of the CAM is shown in Figure 8. It generates a channel attention map by utilizing the inter-channel relationships of features. Before operations, CAM compresses the feature map in the spatial dimension, generating a one-dimensional vector [40]. Initially, the input feature map undergoes global max pooling and global average pooling based on width and height [41]. Two spatial context descriptors are then created: the average pooling descriptor F^c_{Avg} and the max pooling descriptor F^c_{Max}. These two descriptors represent the features processed through average pooling and max pooling, respectively, resulting in two feature maps of dimensions $C \times 1 \times 1$.

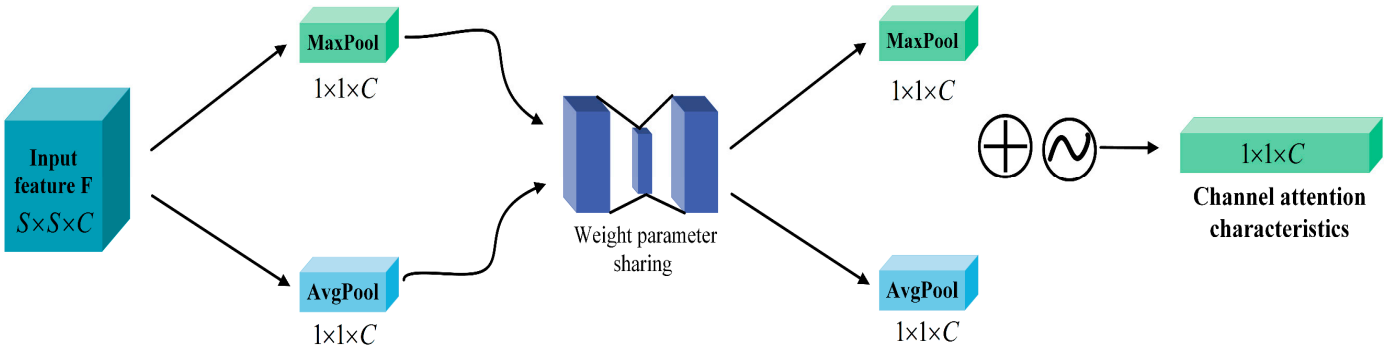


Figure 8. Channel Attention Module.

Subsequently, these two feature maps are each fed into a dual-layer neural network for processing. The parameters of this neural network are shared across the two processing steps. Through this process, the neural network's output features are element-wise summed. Next, these summed features are processed through a Sigmoid activation function to produce the final channel attention feature map, which we refer to as *Mc*. Lastly, to construct the input features required for the spatial attention module, an element-wise multiplication is performed between *Mc* [42] and the original input feature map F. In this manner, the channel attention computed thus guides the model to focus more on important feature channels.

$$\begin{aligned} Mc(F) &= \sigma(MLP(AvgPool(F)) + MLP(AvgPool(F))) \\ &\sigma(W_1(W_0(F^c_{Avg})) + W_1(W_0(F^c_{Max})) \end{aligned} \tag{6}$$

The symbol σ represents the sigmoid function $W_0 \in RC/r \times C$, $W_1 \in RC \times C/r$.

SAM is depicted in Figure 9. Particularly for small targets, spatial attention helps to gather spatial information. Spatial attention utilizes the connections between various spatial positions to learn a 2D space weight map. The obtained spatial positions are then multiplied to acquire more representative features. The input feature map for this module, F', is generated using the previous CAM module. Initially, we consider channel-wise global maximum pooling and global average pooling. It obtains two $H \times W \times 1$ feature maps, F^s_{Avg} and F^s_{Max}. These two feature maps are channel-wise concatenated. The dimensionality is reduced to one channel through a 7×7 convolution. The spatial attention features *Ms* [43] are generated by the sigmoid function. The resulting final features are calculated by multiplying the module's features by its input features. Here σ represents the sigmoid function and $f^{(7\times7)}$ represents the convolution operation with a size of 7×7. The calculation of spatial attention is as follows:

$$\begin{aligned} Ms(F) &= \sigma(f^{(7\times7)}([AvgPool(F); MaxPool(F)])) \\ &= \sigma(f^{(7\times7)}([F^s_{Avg}; F^s_{Max}])) \end{aligned} \tag{7}$$

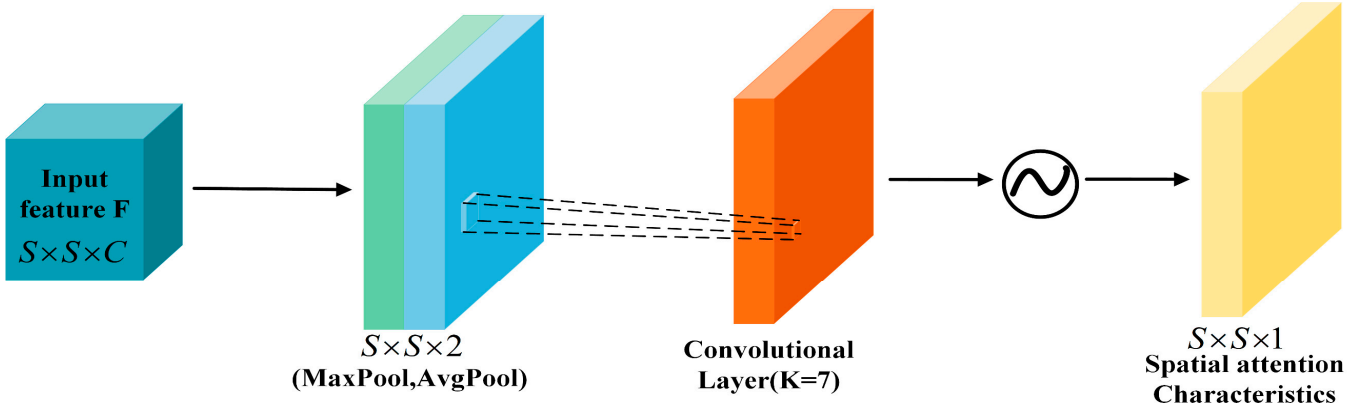


Figure 9. Spatial Attention Module.

2.4.2. Firehead

The framework of the decoupled detection head with CBAM attention that we proposed is illustrated in Figure 10b. We compared the original detection head of YOLOv8 with this paper's proposed Firehead. As shown in Figure 10a, the original detection head outputs predictions through only three convolutions. In contrast, we enhance the calculation of classification and regression subtasks (Loss_cls and Loss_box) by incorporating a CBAM module into each of the two parallel branches. This allows each branch to better learn the characteristics needed for its task based on the loss functions. Each subtask branch is processed by CA and SA modules, guiding the detection head to focus on the effective

information of feature maps according to task attributes. In the subsequent experimental section, it is evident that this paper's designed Firehead, compared to the original detection head, enables each of this paper's sub-branch modules to more effectively focus on features of interest.

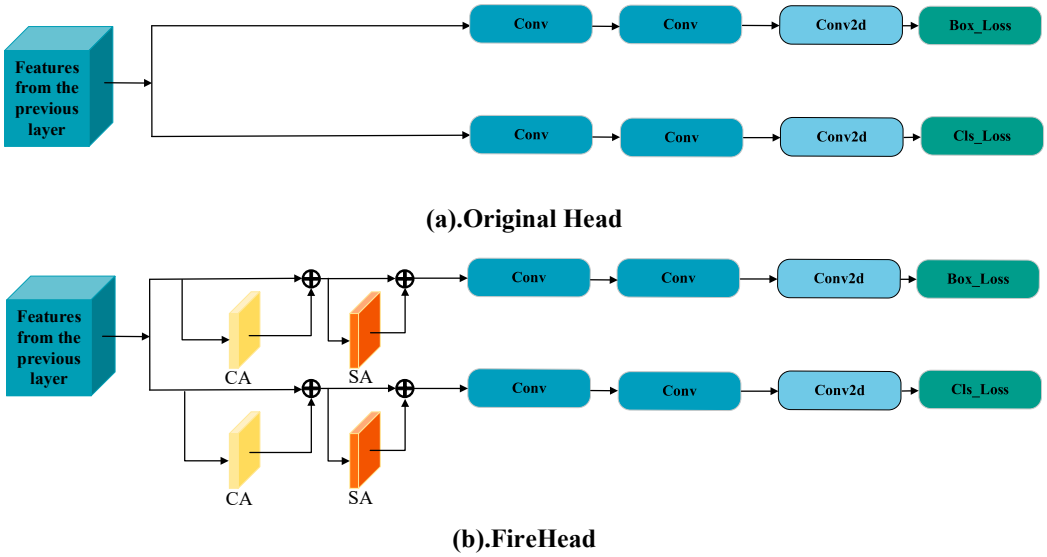


Figure 10. Structure of the original decoupled head and the improved decoupled head.

Additionally, with a minimal amount of parameters and computational cost, it relatively effectively solves the problems of false detections and missed detections of small targets under the interference of complex backgrounds.

3. Experiments and Results

3.1. Experimental Configuration

To ensure fairness and authenticity in the experiments, we conducted tests under the same hardware and software conditions to evaluate the performance of this paper's designed FireNet. Specific parameters are listed in Table 2.

Table 2. Experimental environment settings.

Hardware environment	CPU	Inte(R) Xeon(R) Silver 4210R CPU @ 2.40 GHz
	GPU	NVIDIA GeForce RTX 3080
	RAM	64 G
Software Environment	OS	Windows 10
	CUDA Toolkit	12.2
	Python	3.8.18
Training information	Optimizer	SGD
	Epoch	300
	Batch size	8
	Learning range	0.01

3.2. Evaluation Criteria

To objectively evaluate the performance of the model, we used a set of commonly utilized evaluation metrics. Accuracy (*Acc*) Equation (8): Accuracy is the proportion of correct predictions among all predictions. Precision (*P*) Equation (9): Precision refers to the probability of correct detection among all detected targets. Recall (*R*) Equation (10): Recall is the probability of correctly identifying all positive samples. mAP0.5 Equation (11):

Set the IoU threshold to 0.5, calculate the Average Precision (*AP*) for each class across all images, and then take the average across all classes [43].

$$Acc = \frac{TP + TN}{TP + FN + FP + FN} \tag{8}$$

$$Acc = \frac{TP + TN}{TP + FN + FP + FN} \tag{9}$$

$$\text{Re}call = \frac{TP}{TP + FN} \tag{10}$$

$$mAP = \frac{\sum_{j=1}^{M} AP_j}{M} \tag{11}$$

where *TP* represents true positive samples, *FP* represents false positive samples, *TN* represents true negative samples, and *FN* represents false negative samples, *M* represents the total number of classes to be classified.

3.3. Performance Evaluation and Study of FireNet Modelt

3.3.1. Global Experiment

From Table 3, it can be observed that setting the learning rate to 0.01 allows the model to learn effectively without causing excessive fluctuations or overly slow learning. Ultimately, the network structure improved by 5.9% in mAP@0.5, with only a 1.4 ms sacrifice in inference speed. Additionally, the parameters and Floating Point Operations Per Second (FLOPs) increased by just 0.15 M and 0.3 G, respectively, making this approach economically feasible. This indicates that the strategy enhances the model's feature fusion capability. The RepViT backbone significantly improved the model's contextual connectivity, contributing to a 2.3% increase in mAP@0.5. Similarly, the Dynamic Snake Convolution used as the main component of the neck effectively addressed issues of missed and false detections at edges, with an inference speed loss of only 0.2 ms and FLOPs remaining nearly consistent with the baseline model. Furthermore, the proposed Firehead module excelled in handling background and interference factors, resulting in a 2.6% improvement in mAP@0.5, while also maintaining FLOPs consistent with the baseline model. This aligns well with the lightweight and efficient detection head design philosophy of this paper. These compelling results highlight the effectiveness of these modules in achieving real-time fire detection without significantly impacting detection accuracy.

Table 3. Global experiment.

RepVit	C2f_DSConv	FireHead	mAP@0.5%	mParam (M)	GFLOPs	Time (ms)
-	-	-	0.743	5.18	10.3	25.3
√	-	-	0.766	5.20	10.3	25.8
-	√	-	0.753	5.19	10.3	25.5
-	-	√	0.769	5.21	10.3	25.9
√	√	-	0.767	5.21	10.4	26.3
√	-	√	0.788	5.25	10.5	26.4
-	√	√	0.796	5.24	10.4	26.2
√	√	√	0.802	5.33	10.6	26.7

3.3.2. ViT Structure Experiment (Backbone)

In the previous sections, we mentioned that compared to cellular neural networks, (ViT exhibit superior global feature extraction capabilities, which is crucial for the backbone's main components. We compared several mainstream ViT models (Swin Transformer [43], EfficientViT [44], MobileViT [45]). We conducted ablation replacements on the backbone part, RepViT, addressing concerns mentioned earlier about the impact of ViT modules on detection efficiency while enhancing feature extraction ability and recognition accuracy.

From the perspective of practical detection applications, we simultaneously compared mAP@0.5 and inference time. As shown in Figure 11, we used a scatter plot to intuitively present the performance of several ViT models as backbone structures, where points closer to the direction indicated by the arrow represent better overall performance. It can be observed that this paper's RepViT backbone performs the best overall, offering the highest detection accuracy while only being slightly slower than the EfficientViT backbone. When considered a lightweight ViT network, RepViT stands out in this paper's fire detection tasks. Meanwhile, the commonly used Swin Transformer network, although second in detection accuracy, has a significantly higher detection speed compared to other lightweight models, which corroborates This paper's concerns about the speed issues of ViT models.

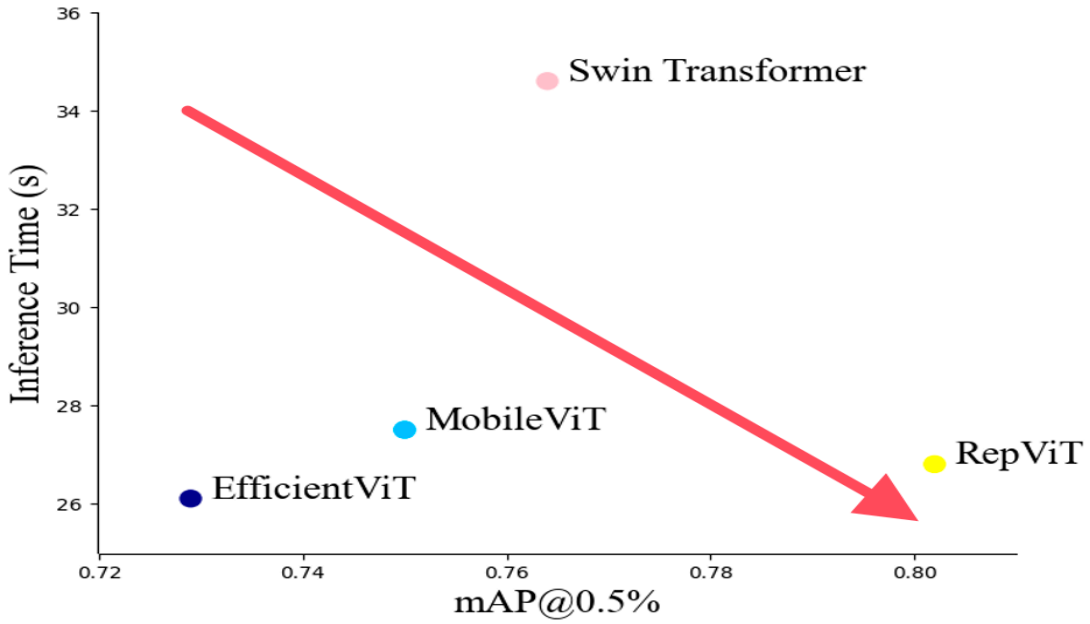


Figure 11. Experiment on the Backbone Part.

3.3.3. Neck 'Number and Position' Study

In the previous text, we designed a neck structure based on the DSConv (Dynamic Snake Convolution) as the core idea, to address the fragile and variable boundaries of flames and smoke. We also mentioned that the increase in the number of DSConvs does not linearly enhance performance and might even add unnecessary computational burdens to the model. Yao et al. [46] mentioned that besides the number of modules affecting the model's performance, the position of the modules also has a significant impact. As shown in Figure 12, we set the upper limit of the number of C2f_DSConv modules to four, and we marked the four positions where the C2f_DSConv exists as A, B, C, D.

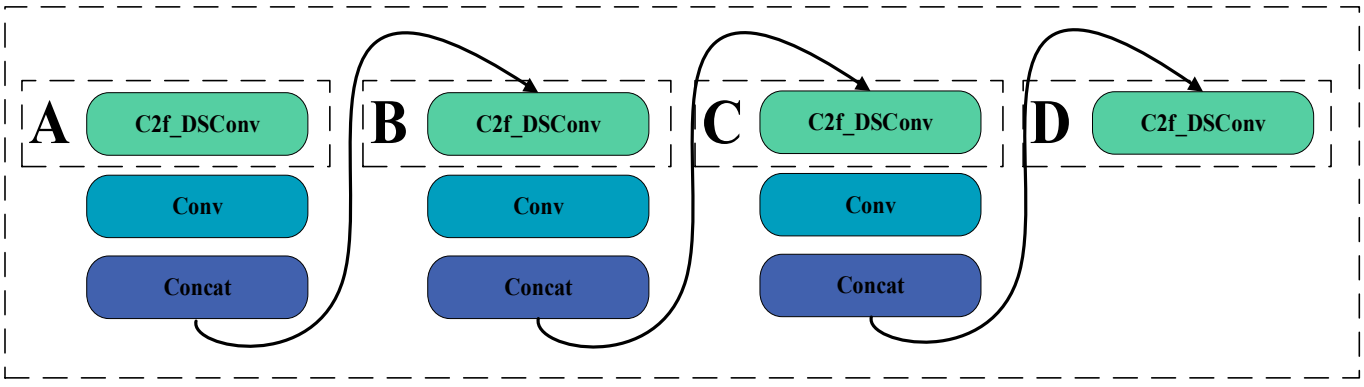


Figure 12. C2f_DSConv Position and Quantity Distribution.

As shown in Table 4, in the variety of modules, adjusting the position does not necessarily affect the model's FLOPs. We observed that although the combination of ABCD achieved the second highest mAP@0.5 at 80.11%, it also brought a significant increase in computational and parameter volume, with inference speed reaching 30.1 ms. Clearly, this method of simply stacking effective modules is not economical. On the other hand, the BC combination is the most economical method. Compared to the best single module, C, BC increased the mAP@0.5 by 1.17% and reduced the inference speed by 1.6 ms. Compared to the best three-module combination, ACD, BC increased the mAP@0.5 by 0.99% and reduced the inference speed by 0.7 ms. Taking everything into account, we chose the BC combination for its highest economy.

Table 4. The impact of the position and quantity of C2f_DSConv on the model.

	mAP@0.5/%	Params/M	GFLOPs	Time (ms)
A	78.24	5.34	10.5	28.3
B	78.59	5.29	10.5	26.4
C	79.11	5.34	10.5	28.3
D	77.02	5.53	10.5	28.0
AB	78.65	5.33	10.6	27.8
AC	78.47	5.41	10.6	27.1
AD	79.44	5.60	10.6	29.2
BC	80.28	5.33	10.6	26.7
BD	78.98	5.33	10.6	26.8
CD	76.33	5.60	10.6	29.3
ABC	79.22	5.42	10.7	27.1
ABD	78.57	5.45	10.7	27.3
ACD	79.29	5.49	10.7	27.4
BCD	78.64	5.48	10.7	27.4
ABCD	80.11	5.69	10.8	30.1

3.3.4. Experiment with Decoupled Detection Heads Carrying Attention Mechanisms

We designed a decoupled detection head with attention mechanisms, integrating different types of attention mechanisms into various task branches. To evaluate the effectiveness of the model, we conducted ablation replacement experiments on our self-built FAS dataset, testing various types of attention mechanisms, including SE, EMA, and GAM. SE dynamically adjusts channel weights to enhance feature extraction capabilities, EMA reduces computational costs by optimizing feature distribution, and GAM improves detection accuracy through a global attention mechanism. It is important to note that the ablation studies were conducted based on Firehead.

As shown in Table 5, the results show that compared to the original YOLOv8 detection head, the decoupled detection heads equipped with an attention mechanism indeed caused a change in model performance. However, introducing such detection heads does indeed reduce model speed, but this reduction is minimal and can be considered negligible. Even the slowest GAM only reduced the speed by 3.7 ms, a loss that is acceptable compared to the precision benefits it brings. We selected three contrastive attention mechanisms: SE dynamically adjusts the weights of different channels, thereby enhancing focus on important channels and effectively improving detection accuracy. Compared to the baseline model, the detection head with the SE attention mechanism increased mAP@0.5 by only 0.2%, which does not perform very well for this specific detection task. EMA aims to retain information on each channel and reduce computational costs, reshaping some channels into the batch dimension and grouping the channel dimensions into multiple sub-features, allowing spatial semantic features to be evenly distributed in each feature group. It is observable that its mAP@0.5 reaches up to 80.1%, and it does not bring excessive computational burdens or vast parameters to the model. However, compared to CBMA, CBMA achieved a higher mAP@0.5 while reducing 0.77 M in parameter size and 0.8 GFLOPs. GAM is a global-based attention mechanism structure, it is evident that it

increased the mAP@0.5 by 2.7% compared to the baseline structure, but it introduced more than 2.15 M additional parameters and 9.5 GFLOPs compared to the baseline model, which is not economical from a cost perspective.

Table 5. Comparison of different attention mechanisms.

	mAP@0.5%	Recall%	Precision%	Param (M)	GFLOPs	Time (ms)
None	76.7	74.2	77.3	5.21	10.4	25.9
SE [47]	76.8	75.5	78.0	5.33	10.7	27.1
EMA [44]	80.1	79.8	81.1	6.10	11.4	26.9
GAM [45]	79.4	74.4	79.9	7.46	19.9	29.6
CBAM	80.2	78.4	82.6	5.33	10.6	26.7

Therefore, choosing the CBMA module to construct this paper's detection head effectively improves this paper's model's feature extraction ability and effectively solves the dual objectives of lightweight deployment and high-accuracy detection within the fire detection framework. This well-considered decision aligns with the stringent requirements for computational efficiency while optimizing fire event recognition and mitigation accuracy in actual field scenarios.

3.4. Comparison Experiments and Visualization

In a controlled experimental setting, we compared the FireNet model with the most representative algorithms currently available, including one-stage algorithm representatives from the YOLOv7 [48] series, YOLOv8 series, the two-stage algorithm representative Faster-RCNN [49], the popular remote sensing algorithm RetinaNet [50], BF_MB-YOLOv5 [51], as well as the latest YOLOv11. Additionally, on the public FSD dataset, we conducted comparisons and evaluations of seven models based on six parameters: mAP@0.5, recall, precision, parameters, GFLOPs, and inference time, as shown in Table 6. To clearly display the comparison results, we used a tabular format, where larger values in the second to fourth columns indicate better performance, while smaller values in the fifth to seventh columns indicate better performance.

Table 6. Comparison with other detection models.

	mAP@0.5/%	Recall/%	Precision/%	Param (M)	GFLOPs	Time (ms)
Fast R-CNN	69.4	72	75.7	5.18	10.2	25.2
RetinaNet	70.4	69.6	68.7	6.91	16.2	31.8
YOLOv7	73.6	72.2	74.2	5.53	10.8	27.2
YOLOv7x	74.1	73.1	75.9	6.11	12.8	28.8
YOLOv8	74.3	72.1	76.8	5.18	10.3	25.3
BF_MB-YOLOv5	79.3	78.2	76.8	5.46	11.5	26.5
YOLOv11	80.3	78.5	82.7	5.30	11.2	27.0
(Ours)	80.2	78.4	82.6	5.33	10.6	26.7

It can be observed that in the three key metrics of object detection accuracy—mAP@0.5, recall, and precision—our method performs second only to YOLOv11. Compared to BF_MB-YOLOv5, which ranks third in object detection accuracy, This paper method surpasses it by 0.9%, 0.2%, and 5.8% respectively, while the inference speed between the two remains almost identical. In terms of parameters and GFLOPs, This paper method is smaller, with a reduction of 1.3 M parameters and 0.9 GFLOPs compared to BF_MB-YOLOv5, and 0.6 M parameters and 0.3 GFLOPs compared to YOLOv11. Overall, This paper's FireNet model enhances detection capabilities while maintaining a smaller parameter count, which is a significant advantage. Compared to the lightweight Fast R-CNN model, This paper model is only 1.5 ms slower in inference speed but achieves a 10.8% higher mAP@0.5. In terms of

comprehensive performance assessment, This paper model continues to hold an advantage in fire detection. Overall, this comparison reaffirms the success of This paper's model.

Finally, to visually represent the performance of FireNet, we also compared FireNet, YOLOv8, and Fast R-CNN on the FSD public dataset in Figure 13 where we showcased representative samples from four types of detection problems. In (a) Fast R-CNN shows unnecessary clutter detection issues, while YOLOv8 encounters background false detections. In (b) small target structures, Fast R-CNN experiences omissions while YOLOv8 mistakenly detects smoke-like trees nearby. In (c) within the mixed samples of clouds and smoke, both YOLOv8 and Fast R-CNN are observed to have varying degrees of misidentification. In (d) it can be seen that both YOLOv8 and Fast R-CNN struggle with unclear boundary targets, unable to fully recognize them, with significant missing detections. However, This paper FireNet compensates for this series of issues. This further indicates that This paper model possesses strong performance across various fire-related scenarios.

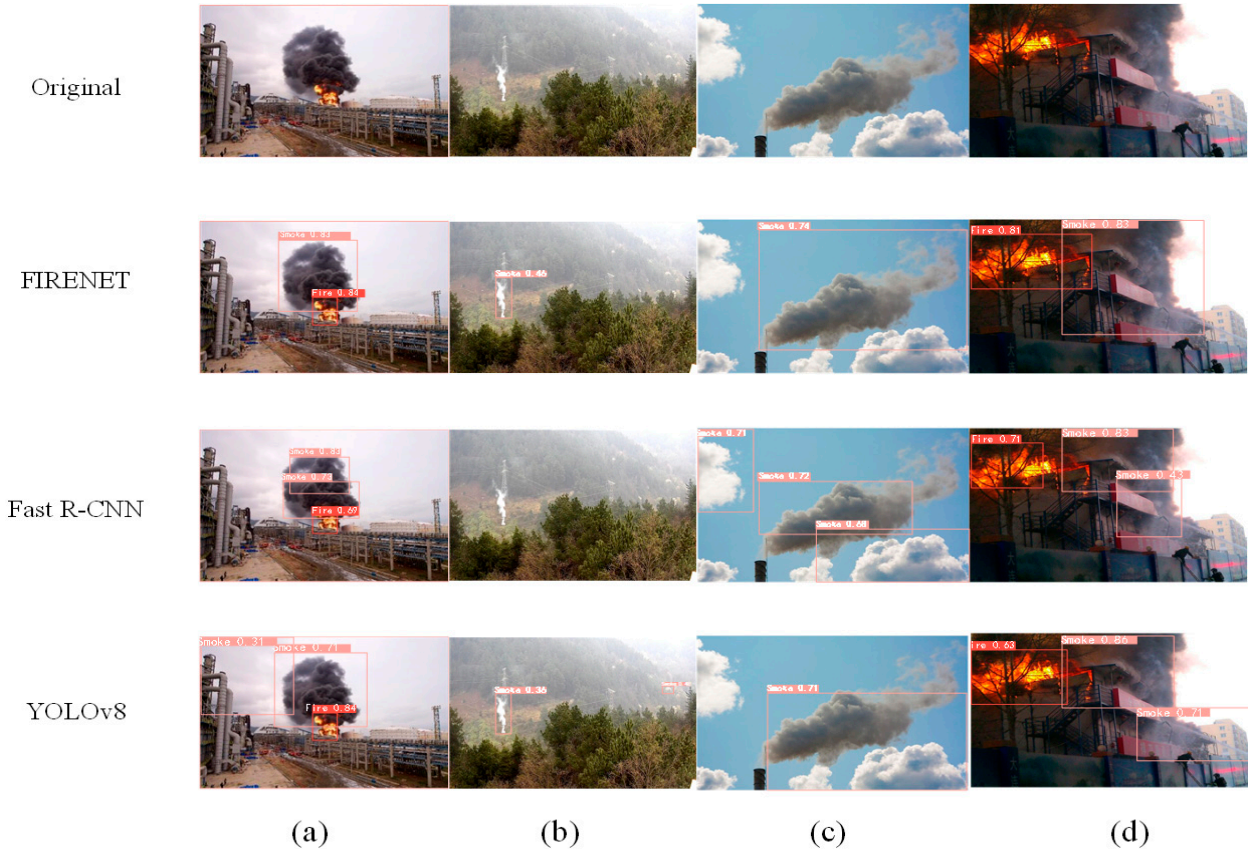


Figure 13. Detection results on FSD datasets. (**a**) Background interference. (**b**) Small target. (**c**) Similar targets. (**d**) Fragile and fluctuating edges.

4. Discussion

This study aims to develop a real-time detection model, FireNet, that is lightweight and highly accurate for multi-scenario fire detection. The focus is on enhancing the feature extraction capability and detection accuracy while keeping the parameters and computational complexity within an economical range, thus maximally satisfying the needs of multi-scenario fire detection.

Inspired by the YOLO series and the trisected structure, we designed a network structure composed of a backbone, neck, and head. For the backbone part, we built on the integration of NLP and CV in the ViT structure and used RepViT as the backbone structure, which is an improvement over the standard lightweight MobileNetV3-L block structure. By moving the depthwise separable convolutions upward, we successfully separated the token

mixer and channel mixer. Additionally, we introduced structural reparameterization to effectively reduce computation and memory costs. Furthermore, to minimize the model's parameters further, we unified the kernel size of convolutions in MobileNetV3-L to 3×3. This change enabled us to achieve a lightweight model, speed up inference while ensuring accuracy, and significantly reduce the model's dependence on hardware environments. Compared to the Vit models applied in fire detection like Swin Transformer, This paper RepViT showed superior performance with less decrease in detection speed, which is not economical for real-time detection as evidenced in Figure 11.

To tackle the problem of missing and false detections caused by the uncertain and variable boundaries of flames and smoke, we designed a C2f_DSConv structure, using dynamic snake convolutions instead of standard convolutions. DSConv was initially developed to adapt to tubular structures with minor feature changes, similar to the edge features of flames and smoke. The improvement allows this paper's network to pay more attention to the fine-grained features of the targets while enhancing feature extraction capabilities. To validate the effect of the variable convolution's number and position proposed in [38], we experimented with the number and location of C2f_DSConv, and the data in Table 4 show that the impact of DSConv on the model mostly relates to the number, with less effect from the position. Based on this theory, we designed the best-performing BC arrangement.

Finally, we designed a decoupled detection head carrying an attention mechanism. This decoupled detection head can provide independent branches for localization and classification, avoiding mutual interference. When facing complex backgrounds and interference factors in fire detection, compared to the dilated convolution strategy used for the decoupled head in [52,53], this paper's attention mechanism guides the detector to focus more on effective features and ignore the interference noise caused by complex backgrounds. We tested various attention mechanism heads in Table 5, and this paper's designed detection improved mAP@0.5 by 3.5% with only a 0.8 ms sacrifice in inference speed, which is negligible.

We selected several current cutting-edge algorithms for comparison on the multi-scenario fire dataset FAS (Figure 13), which consists of 40% urban fire scenarios, 35% forest fire scenarios, and 25% residential fire scenarios. Experiments prove that the method proposed in this paper outperforms others in terms of detection accuracy and efficiency, including Fast R-CNN, YOLOv7, YOLOv8, RetinaNet, and BF_MB-YOLOv5, making the FireNet proposed in this paper the most outstanding and economical solution. It is worth noting that these algorithms have been widely used in the field of real-time fire detection (YOLOv7 [54], YOLOv8 [55], Fast R-CNN [56], RetinaNet [57], YOLOv5 [42]) and have significant reference value. Despite the current good performance of the FireNet model, there are still several areas worth paying attention to in the future.

Although the FireNet model performs well in terms of accuracy, its detection speed and parameter size are slightly inferior to YOLOv8 and Fast R-CNN, requiring further lightweight optimization in the future. The impact of module quantity and position on performance is not linear and needs further investigation. Additionally, the detection of complex boundaries of flames and smoke remains challenging, especially in high-altitude images captured by drones, where the edges can easily blend with the background, increasing the difficulty of recognition.

Future Directions:

- More lightweight

In Figure 13, we observe that although This paper FireNet performs well in terms of the object detection accuracy metrics mAP@0.5, recall, and precision, it is slightly inferior to YOLOv8 and Fast R-CNN in terms of detection speed and the number of parameters. However, this slight loss in speed is economical under the substantial improvement in accuracy. In subsequent network optimization, we plan to test some lightweight modules, such as GhostNet, which combines a small number of convolutional kernels with cheaper linear transformation operations. We also plan to test the SimAM module, which derives

3D attention weights for feature maps without the need for additional parameters. It is important to note that we need to optimize detection speed while ensuring detection progress, in order to accommodate the model in portable devices.

- Module position study

We proposed 15 schemes for this paper's designed neck structure based on current research on module layout and quantity. However, experimental results showed differences from those proposed by Yao et al. [46], suggesting that module position does not significantly affect model speed, parameter size, and computational complexity under certain module numbers. We suspect this may be due to different datasets and many other factors. In the future, to validate this theory, we need to continue discussing the impact of position on the model under consistent conditions.

5. Conclusions

A model for real-time multi-scenario fire detection, FireNet, was proposed. We constructed the network using a "trisected" model architecture, incorporated a lightweight ViT module, RepViT, as the backbone, and replaced standard convolutions with a DSConv structure tailored for fragile and variable tubular walls to form the C2f_DSConv neck. Lastly, we designed a decoupled detection head equipped with an attention mechanism, which is intended to guide the neural network to extract more effective features from complex backgrounds while minimizing additional computational costs. Ultimately, this paper's designed FireNet achieved 80.2% (an increase of 5.9%) in mAP@0.5, a recall rate of 78.4% (an increase of 6.3%), and an accuracy rate of 82.6% (an increase of 5.8%), with an inference speed reaching 26.7 ms.

In this paper's future research, this paper's goal is to focus more on the real-time detection capabilities of the model, ensuring that it performs well in various complex background detections while exploring issues of detection speed through various lightweight modules. Given FireNet's strong performance in detecting under complex backgrounds and high precision, we aim to continue researching its broader applications beyond multi-scenario fire detection [58].

6. Impact Statement

We developed a novel trifurcated network tailored for various fire detection scenarios, introducing a pioneering blend of c2f with dynamic serpentine convolutions. This innovative approach exploits the unique convolutional framework of serpentine convolutions to adeptly navigate the complex and shifting tubular structures characteristic of fires, offering marked improvements over traditional convolution methods, especially in processing fragile and intricate feature structures. Additionally, This paper design integrates a bifurcated detection head enhanced with an attention mechanism, showcasing the practicality and effectiveness of incorporating attention mechanisms directly into the network's head. This adaptation significantly advances the network's capability in identifying small and highly similar objects, establishing a fresh paradigm for the application of attention mechanisms within detection systems. This novel architecture not only broadens the scope of fire detection capabilities but also sets a new standard for convolutional network design in dynamic and challenging environments.

Author Contributions: Conceptualization, Y.H. and R.Z.; methodology, Y.H., A.S., X.H. and R.Z.; Validation, X.H.; investigation, R.Z; data curation, R.W.; writing—original draft preparation, Y.H. and A.S.; writing—review and editing, Y.H., X.H., R.W. and R.Z.; funding acquisition, R.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was jointly funded by the National Key Research and Development Program of China (Grant No. 2023YFB2604001), Tibet Autonomous Region Key Research and Development Program (XZ202401ZY0057), the National Natural Science Foundation of China (Grant No. 42371460) and the Major Science and Technology Special Project of Sichuan Province (Grant No. 2023ZDZX0030).

Data Availability Statement: This research data, are available on https://github.com/DC9874/FireNet, accessed on 22 October 2024.

Acknowledgments: We are grateful to the European Space Agency for providing the Sentinel-1 data freely. We are also thankful to NASA for providing the SRTM DEM data. In addition, we sincerely thank the editors and all anonymous reviewers for their constructive and excellent reviews of This paper work.

Conflicts of Interest: The authors declare that they have no financial or other conflicts of interest in the course of this study.

References

1. Jain, A.; Srivastava, A. Privacy-Preserving Efficient Fire Detection System for Indoor Surveillance. *IEEE Trans. Ind. Inform.* **2022**, *18*, 3043–3054. [CrossRef]
2. Yang, X.; Zhang, R.; Li, Y.; Pan, F. Passenger Evacuation Path Planning in Subway Station Under Multiple Fires Based on Multiobjective Robust Optimization. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21915–21931. [CrossRef]
3. John, J.; Harikumar, K.; Senthilnath, J.; Sundaram, S. An Efficient Approach with Dynamic Multiswarm of UAVs for Forest Firefighting. *IEEE Trans. Syst. Man Cybern. Syst.* **2024**, *54*, 2860–2871. [CrossRef]
4. Çelik, T.; Özkaramanlı, H.; Demirel, H. Fire and smoke detection without sensors: Image processing based approach. In Proceedings of the 2007 15th European Signal Processing Conference, Poznan, Poland, 3–7 September 2007; pp. 1794–1798.
5. Almeida, J.S.; Huang, C.; Nogueira, F.G.; Bhatia, S.; de Albuquerque, V.H.C. EdgeFireSmoke: A Novel Lightweight CNN Model for Real-Time Video Fire–Smoke Detection. *IEEE Trans. Ind. Inform.* **2022**, *18*, 7889–7898. [CrossRef]
6. Xie, J.; Zhao, H. Forest Fire Ob-ject Detection Analysis Based on Knowledge Distillation. *Fire* **2023**, *6*, 446. [CrossRef]
7. Wang, Z.; Gao, Q.; Xu, J.; Li, D. A Review of UAV Power Line Inspection. Advances in Guidance. *Navig. Control. Lect. Notes Electr. Eng.* **2022**, *644*, 3147–3159.
8. Chiu, Y.-Y.; Omura, H.; Chen, H.-E.; Chen, S.C. Indicators for post-disaster search and rescue efficiency developed using progressive deathtolls. *Sustainability* **2020**, *12*, 8262. [CrossRef]
9. Ye, T.; Qin, W.; Li, Y.; Wang, S.; Zhang, J.; Zhao, Z. Dense and small object detection in UA V-vision based on a global-local feature enhanced network. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–13.
10. Jayathunga, S.; Pearse, G.D.; Watt, M.S. Unsupervised Methodology for Large-Scale Tree Seedling Mapping in Di-verse Forestry Settings Using UAV-Based RGB Imagery. *Remote Sens.* **2023**, *15*, 5276. [CrossRef]
11. Dong, Y.; Xie, X.; An, Z.; Qu, Z.; Miao, L.; Zhou, Z. NMS Free Oriented Object Detection Based on Channel Expansion and Dynamic Label Assignment in UAV Aerial Images. *Remote Sens.* **2023**, *15*, 5079. [CrossRef]
12. Chen, X.; An, Q.; Yu, K.; Ban, Y. A Novel Fire Identification Algorithm Based on Improved Color Segmentation and Enhanced Feature Data. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–15. [CrossRef]
13. Qiu, T.; Yan, Y.; Lu, G. An Autoadaptive Edge-Detection Algorithm for Flame and Fire Image Processing. *IEEE Trans. Instrum. Meas.* **2012**, *61*, 1486–1493. [CrossRef]
14. Xie, Y.; Zhu, J.; Cao, Y.; Zhang, Y.; Feng, D.; Zhang, Y.; Chen, M. Efficient Video Fire Detection Exploiting Motion-Flicker-Based Dynamic Features and Deep Static Features. *IEEE Access* **2020**, *8*, 81904–81917. [CrossRef]
15. Xi, D.; Qin, Y.; Luo, J.; Pu, H.; Wang, Z. Multipath Fusion Mask R-CNN with Double Attention and Its Application Into Gear Pitting Detection. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–11. [CrossRef]
16. Fang, F.; Li, L.; Zhu, H.; Lim, J.-H. Combining Faster R-CNN and Model-Driven Clustering for Elongated Object Detection. *IEEE Trans. Image Process.* **2020**, *29*, 2052–2065. [CrossRef]
17. Hnewa, M.; Radha, H. Integrated Multiscale Domain Adaptive YOLO. *IEEE Trans. Image Process.* **2023**, *32*, 1857–1867. [CrossRef]
18. Zhang, H.; Tian, Y.; Wang, K.; Zhang, W.; Wang, F.-Y. Mask SSD: An Effective Single-Stage Approach to Object Instance Segmentation. *IEEE Trans. Image Process.* **2020**, *29*, 2078–2093. [CrossRef]
19. Muhammad, K.; Ahmad, J.; Baik, S.W. Early fire detection using convolutional neural networks during surveillance for effective disaster management. *Neurocomputing* **2018**, *288*, 30–42. [CrossRef]
20. Barmpoutis, P.; Dimitropoulos, K.; Kaza, K.; Grammalidis, N. Fire Detection from Images Using Faster R-CNN and Multidimensional Texture Analysis. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 8301–8305.
21. Wu, Z.; Xue, R.; Li, H. Real-Time Video Fire Detection via Modified YOLOv5 Network Model. *Fire Technol.* **2022**, *58*, 2377–2403. [CrossRef]
22. Li, J.; Zhou, G.; Chen, A.; Lu, C.; Li, L. BCMNet: Cross-Layer Extraction Structure and Multiscale Downsampling Network with Bidirectional Transpose FPN for Fast Detection of Wildfire Smoke. *IEEE Syst. J.* **2023**, *17*, 1235–1246. [CrossRef]
23. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A. An Image is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
24. Sun, Z.; Cao, S.; Yang, Y.; Kitani, K.M. Rethinking transformer-based set prediction for object detection. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 3611–3620.

25. Dai, Y.; Liu, W.; Wang, H.; Xie, W.; Long, K. YOLO-Former: Marrying YOLO and Transformer for Foreign Object Detection. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–14. [CrossRef]
26. Li, W.; Zhang, L.; Wu, C.; Cui, Z.; Niu, C. A new lightweight deep neural network for surface scratch detection. *Int. J. Adv. Manuf. Technol.* **2022**, *123*, 1999–2015. [CrossRef] [PubMed]
27. Huang, J.; He, Z.; Guan, Y.; Zhang, H. Real-Time Forest Fire Detection by Ensemble Lightweight YOLOX-L and Defogging Method. *Sensors* **2023**, *23*, 1894. [CrossRef]
28. Liu, L.; Song, X.; Lyu, X. FCFR-Net: Feature fusion based coarse-to-fine residual learning for depth completion. *arXiv* **2020**, arXiv:2012.08270. [CrossRef]
29. Tao, H. A label-relevance multi-direction interaction network with enhanced deformable convolution for forest smoke recognition. *Expert. Syst. Appl.* **2024**, *236*, 121383. [CrossRef]
30. Jocher, G.; Stoken, A.; Borovec, J.; Changyu, L.; Hogan, A.; Diaconu, L.; Rai, P. Ultralytics/YOLOv5: Initial Release; Zenodo: 2020. Available online: https://zenodo.org/record/3983579 (accessed on 22 October 2024).
31. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 9992–10002.
32. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Adam, H. Efficient convolutional neural networks for mobile vision applications. *arXiv* **2020**, arXiv:1704.04861.
33. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
34. Xiao, T.; Singh, M.; Mintun, E.; Darrell, T.; Dollár, P.; Girshick, R. Early convolutions help transformers see better. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–14 December 2021; pp. 30392–30400.
35. Qi, Y.; He, Y.; Qi, X.; Zhang, Y.; Yang, G. Dynamic Snake Convolution based on Topological Geometric Constraints for Tubular Structure Segmentation. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 6047–6056.
36. Dai, J. Deformable Convolutional Networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 764–773.
37. Karen, S.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
38. Dang, C.; Wang, Z.X. RCYOLO: An Efficient Small Target Detector for Crack Detection in Tubular Topological Road Structures Based on Unmanned Aerial Vehicles. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 12731–12744. [CrossRef]
39. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
40. Hu, J.; Shen, L.; Albanie, S.; Sun, G. Squeeze-and-excitation networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [CrossRef]
41. Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv* **2013**, arXiv:1312.4400.
42. Yazdi, A.; Qin, H.; Jordan, C.B.; Yang, L.; Yan, F. Nemo: An Open-Source Transformer-Supercharged Benchmark for Fine-Grained Wildfire Smoke Detection. *Remote Sens.* **2022**, *14*, 3979. [CrossRef]
43. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
44. Zheng, Y.; Wang, M.; Chang, Q. Real-Time Helmetless Detection System for Lift Truck Operators Based on Improved YOLOv5s. *IEEE Access* **2024**, *12*, 4354–4369. [CrossRef]
45. Li, X.; Chen, S.; Zhang, S.; Hou, L.; Zhu, Y.; Xiao, Z. Human Activity Recognition Using IR-UWB Radar: A Lightweight Transformer Approach. *IEEE Geosci. Remote Sens. Lett.* **2023**, *20*, 1–5. [CrossRef]
46. Zhang, X.; Li, J.; Hua, Z. MRSE-Net: Multiscale Residuals and SE-Attention Network for Water Body Segmentation From Satellite Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 5049–5064. [CrossRef]
47. Sudakow, I.; Asari, V.K.; Liu, R.; Demchev, D. MeltPondNet: A Swin Transformer U-Net for Detection of Melt Ponds on Arctic Sea Ice. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 8776–8784. [CrossRef]
48. Li, G.; Shi, G.; Zhu, C. Dynamic Serpentine Convolution with Attention Mechanism Enhancement for Beef Cattle Behavior Recognition. *Animals* **2024**, *14*, 466. [CrossRef]
49. Wang, C.; Zhang, B.; Cao, Y.; Sun, M.; He, K.; Cao, Z.; Wang, M. Mask Detection Method Based on YOLO-GBC Network. *Electronics* **2023**, *12*, 408. [CrossRef]
50. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef]
51. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K. Focal Loss for Dense Object Detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2999–3007.
52. Dang, C.; Wang, Z.; He, Y.; Wang, L.; Cai, Y.; Shi, H.; Jiang, J. The Accelerated Inference of a Novel Optimized YOLOv5-LITE on Low-Power Devices for Railway Track Damage Detection. *IEEE Access* **2023**, *11*, 134846–134865. [CrossRef]
53. Baek, J.-W.; Chung, K. Swin Transformer-Based Object Detection Model Using Explainable Meta-Learning Mining. *Appl. Sci.* **2023**, *13*, 3213. [CrossRef]

54. Zhang, W.; Liu, Z.; Zhou, S.; Qi, W.; Wu, X.; Zhang, T.; Han, L. LS-YOLO: A Novel Model for Detecting Multiscale Landslides with Remote Sensing Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 4952–4965. [CrossRef]
55. Cao, X.; Su, Y.; Geng, X.; Wang, Y. YOLO-SF: YOLO for Fire Segmentation Detection. *IEEE Access* **2023**, *11*, 111079–111092. [CrossRef]
56. Guo, X.; Cao, Y.; Hu, T. An Efficient and Lightweight Detection Model for Forest Smoke Recognition. *Forests* **2024**, *15*, 210. [CrossRef]
57. Zheng, X.; Chen, F.; Lou, L.; Cheng, P.; Huang, Y. Real-Time Detection of Full-Scale Forest Fire Smoke Based on Deep Convolution Neural Network. *Remote Sens.* **2022**, *14*, 536. [CrossRef]
58. Kundu, S.; Maulik, U.; Sheshanarayana, R.; Ghosh, S. Vehicle Smoke Synthesis and Attention-Based Deep Approach for Vehicle Smoke Detection. *IEEE Trans. Ind. Appl.* **2023**, *59*, 2581–2589. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.