ISSN: 2710-0588

Lightweight Forest Flame Smoke Detection Algorithm Based on Yolov5

Ziyi Yang[1, a]

[1]Southwest University, Chongqing, 400715, China

[a]ziyiy404@gmail.com

Abstract

Forest smoke and flame detection is dominant in ensuring forest safety. To extinguish fire sources promptly and prevent the spread of wildfires, this paper proposes a lightweight improved YOLOv5 algorithm that is more accessible to embedded devices. The proposed algorithm is based on two pivotal ideas: (1) Introducing a normalization-based NAM attention mechanism into the neck network, which suppresses insignificant weights through weight sparsity penalties to enhance key feature extraction. (2) Incorporating a Slim-neck structure, which leverages GSConv and VoVGSCSP to construct a lightweight neck network. Research findings indicate that the optimized network outperforms the baseline YOLOv5s architecture, with a 3.34% gain in precision (P), while reducing FLOPs by 8.8%. This ensures a more lightweight model while maintaining detection accuracy.

Keywords

Forest fire; YOLOv5; lightweight; NAM; Slim-neck.

1. Introduction

Forest fires are devastating natural disasters that pose serious risks to human lives and result in significant property damage. Therefore, achieving timely and accurate forest fire detection is highly significant.

Over the past few years, as the filed of object detection grows by leaps and bounds, object recognition algorithms have been increasingly applied to forest fire detection. Currently, the two mainstream approaches in object detection are two-stage detection algorithms, exemplified by the R-CNN series, and single-stage detection algorithms, exemplified by the YOLO series.

Among them, YOLO-based algorithms have demonstrated advantages in speed and accuracy for forest fire detection and are widely used in forest fire detection.

Pi Jun et al. proposed a modification to the YOLOv5s architecture by substituting its original backbone with the more lightweight ShuffleNetv2 network while introducing an attention module, further Minimizing its model size, without sacrificing accuracy [1]. Yang Wu et al. developed an improved wildfire recognition approach on the basis of a re-parameterized YOLOv5s model. They designed the RSBlock to replace the original backbone network by integrating RepVGG, DBB, and lightweight network concepts and introduced a lightweight feature fusion block to replace conventional convolutional layers [2]. Qian Chengshan et al. enhanced the YOLOv5 model by incorporating a Transformer model into the convolutional neural network while optimizing the Transformer for lightweight deployment, effectively enhancing detection precision while achieving a marked improvement in the number of parameters [3]. Li Jianwei et al. incorporated the bottleneck architecture from MobileNetv3 and substituted standard convolutions with depthwise separable convolutions to improve detection accuracy [4].

ISSN: 2710-0588

In summary, to meet the requirements of forest fire detection in terms of speed and accuracy, as well as practical deployment constraints, this paper proposes an improved YOLOv5 algorithm. By optimizing the neck network, introducing the NAM attention mechanism, and integrating lightweight convolutional structures to construct a Slim-neck network, the model achieves higher detection speed while maintaining accuracy.

2. Principle of the YOLOv5 Algorithm

The YOLOv5 network was introduced in 2020, offering significantly faster training speed compared to YOLOv4. Additionally, the model size of YOLOv5 was compressed by 90% compared to YOLOv4, which lowered deployment costs and improved deployment efficiency.

YOLOv5 is comprised of three principal components: backbone, Feature Fusion Structure, and detection head. The backbone network obtains crucial information the image processed at the input end. through the use of a cross-stage network structure. Feature Fusion Structure focuses on aggregating and refining the extracted features to ensure that the final feature map retains both rich semantic information and spatial details. The detection head employs GIOU_Loss as the loss function and precisely predicts the classification and coordinates of each goal instance, utilizing the attributes extracted by the backbone and neck networks. It produces three different scale outputs, corresponding to large, medium, and small object predictions.

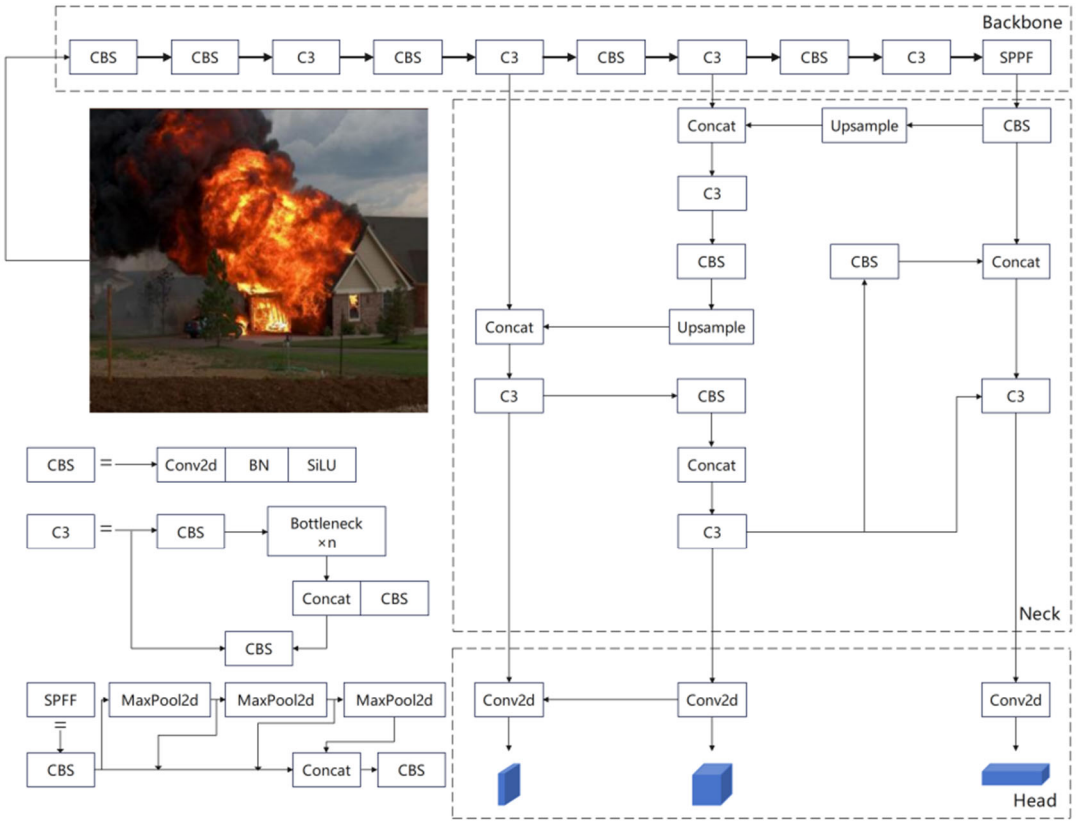

Figure 1. Original YOLOv5 model

3. Improved YOLOv5 Forest Fire Detection Algorithm

3.1. Overview of the Improved YOLOv5 Algorithm

To enhance the accuracy of forest fire detection while addressing the challenge of deploying the model on resource-constrained monitoring devices, this paper proposes an improved

ISSN: 2710-0588

lightweight YOLOv5-based forest fire detection algorithm. The network structure of the improved algorithm is shown in Figure 2.

This study introduces the following two key improvements to the YOLOv5 algorithm:(1) Incorporating the Normalization-based Attention module into the neck network to enhance key feature extraction in target regions, thereby improving detection accuracy.(2) Replacing the YOLOv5s neck network with a lightweight Slim-neck network to decrease the scale of network and inference cost while maintaining or even improving model capability. The framework of the enhanced network is presented in Figure 2.

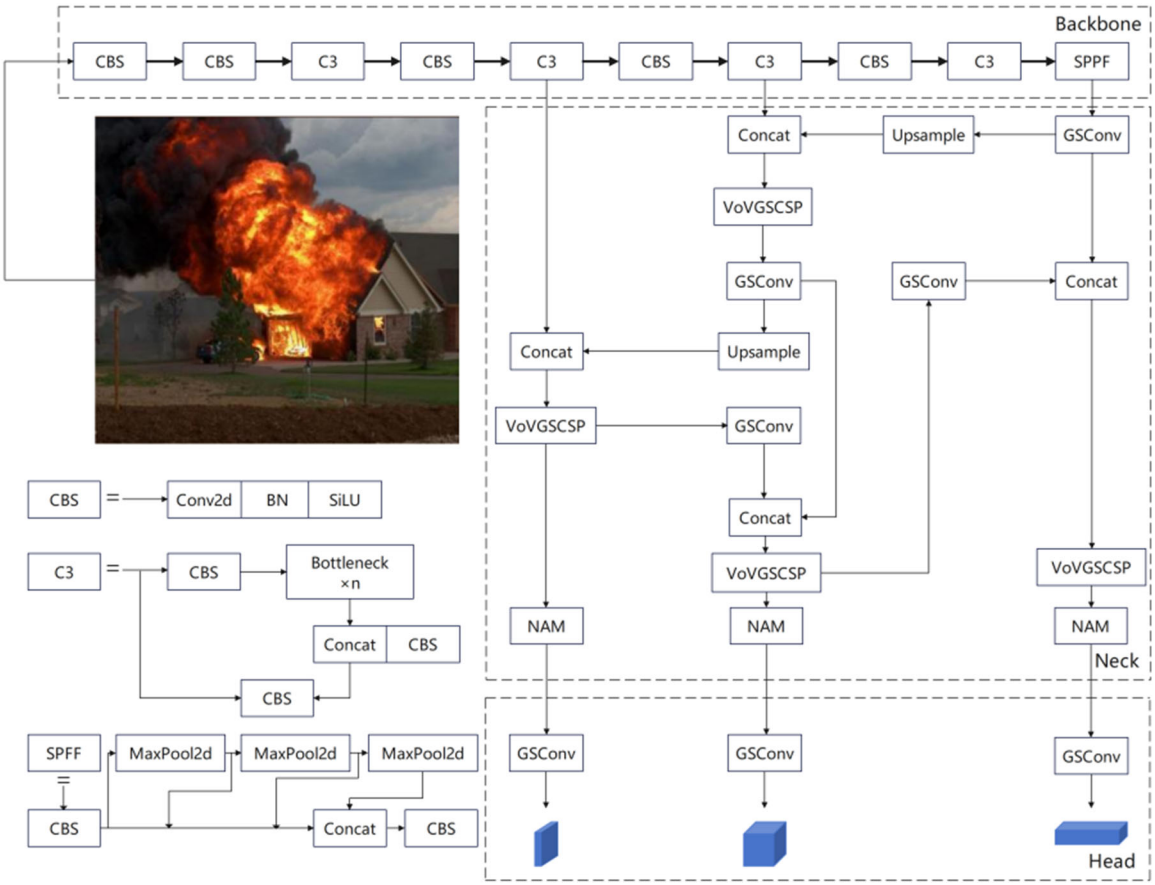

Figure 2. Improving the YOLOv5 model

3.2. Neck Network Improvement

To simplify the initial network while Sustaining precision, this paper introduces the Slim-neck structure. Slim-neck reconstructs the C3 module in the neck network based on GSConv, designs VoVGSCSP, and then replaces the original neck modules in YOLOv5s with GSConv and VoVGSCSP, thereby making the neck network more lightweight.

3.2.1. GSConv Module

To ensure that the output of depthwise separable convolution closely approximates that of standard convolution, this paper introduces grouped-channel shuffled convolution, which integrates the concepts of group convolution and channel shuffle [5]. GSConv decomposes standard convolution into two steps: performing grouped convolution, where the input feature map is partitioned into several groups, and each group undergoes convolution separately, which helps reduce the processing load and network size for each convolution operation. In the channel shuffle process, the outputs of each group convolution are rearranged to increase cross-channel information interaction.

ISSN: 2710-0588

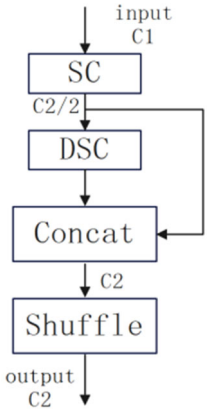

Figure 3. Structure of GSConv

3.2.2. VoVGSCSP Module

Building upon GSConv, VoVGSCSP introduces the GS bottleneck and employs a one-shot aggregation strategy for the construction of the Cross Stage Partial block. This module increases the network depth and enhances feature extraction capability while reducing computational complexity, maintaining good accuracy.

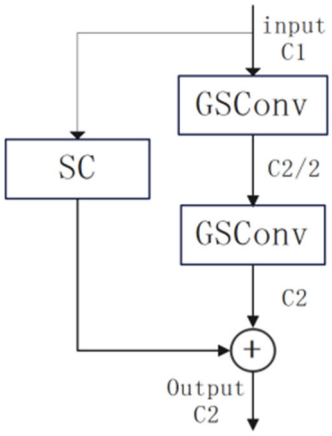

Figure 4. GSbottleneck structure

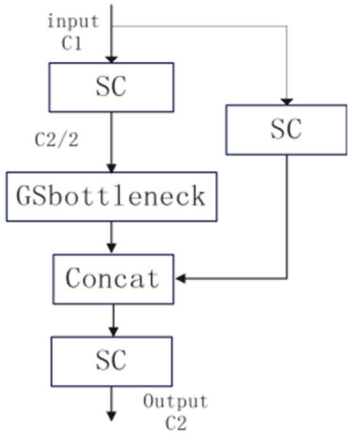

Figure 5. VoVGSCSP structure

ISSN: 2710-0588

3.3. NAM Attention

Based on the framework of CBAM, the NAM attention mechanism sequentially redesigns the channel and spatial attention modules to achieve improved feature refinement, utilizing weight contribution factors to improve attention performance [6]. To avoid the additional fully connected layers and convolutional layers found in SE and CBAM mechanisms. NAM attention employs the scaling factor in batch normalization (BN) to quantify the significance of weights, as shown in Equation 1, to avoid the additional fully connected layers and convolutional layers found in SE and CBAM mechanisms [7].

$$B_{\text{out}} = BN(B_{in}) = \gamma \frac{B_{in} - \mu_\beta}{\sqrt{\sigma_\beta^2 + \varepsilon}} + \beta \tag{1}$$

Among them, μ_B and σ_B represent the mean and standard deviation of the mini-batch B, respectively, while$\gamma$and $\beta$are trainable affine transformation parameters (scale and shift). $\varepsilon$is an infinitesimal constant introduced to prevent the denominator from being zero.

The channel attention mechanism is shown in Figure 6, and the weight computation method and the definition of the channel attention output are given in Equations (2) and (3), respectively.

$$w_\gamma = \gamma_i / \sum_{j=0} \gamma_j \tag{2}$$

$$M_c = Sigmoid(W_\gamma(BN(F_1))) \tag{3}$$

The same method used in the channel attention module is applied along the spatial dimension, where batch normalization is utilized in the pixels in the spatial dimension, that is pixel normalization. The structure of the spatial attention mechanism is depicted in Figure 6. Its weight computation method and output definition are given in Equations (4) and (5), respectively.

$$w_\lambda = \lambda_i / \sum_{j=0} \lambda_j \tag{4}$$

$$M_s = Sigmoid(W_\lambda(BN(F_2))) \tag{5}$$

In addition, to suppress insignificant weights, NAM adds a regularization component to the loss function, termed:

$$Loss = \sum_{(x,y)} l(f(x, W), y) + p \sum g(\gamma) + p \sum g(\lambda) \tag{6}$$

Among them, x and y are the feature input and feature output respectively, w is the network weight,$l(\cdot)$and$g(\cdot)$represents the loss function and the norm penalty term l_1, and p is used to balance$g(\gamma)$and$g(\lambda)$respectively.

ISSN: 2710-0588

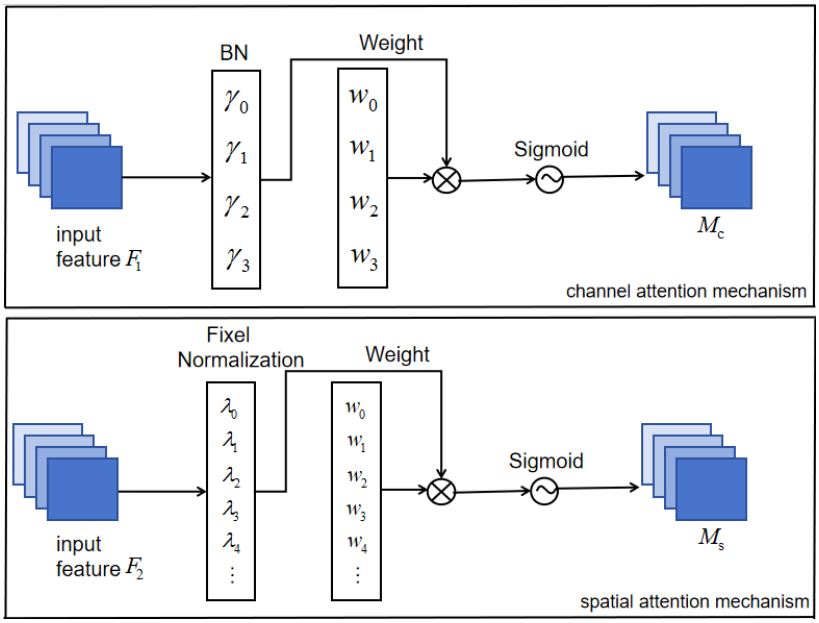

Figure 6. Structure of NAM attention mechanism

4. Experiment and Results

4.1. Dataset Construction

Since there is currently no unified public standard database for forest fires, the experimental dataset in this paper is mainly obtained from Baidu Images. The dataset contains a total of 6000 images, including 926 forest fire images. In order to enhance the generalization of image detection, 5074 images of various types of fire scenes are introduced to assist in model training. Considering that a large amount of smoke is produced during a fire, smoke is introduced into the training as an important condition to assist in determining the occurrence of a fire. Figure 7 displays representative images extracted from the dataset. The dataset is annotated in YOLO format, and data enhancement strategies including flipping, rotation, and cropping are implemented on the original dataset to enhance model's adaptation capability. Labelimg is used to annotate the dataset, and the collected data is segmented into a training subset containing 4,200 images and a test subset with 1,800 images, adhering to a 70/30 split ratio, as detailed in Table 1.

Figure 7. Part of the fire data set

ISSN: 2710-0588

Table 1. Division of the Forest Fire Detection Dataset

parameters	images
Total dataset size	6000
Training set	4200
Test set	1800

4.2. Experimental Configuration and Network Training

The Trial was carried out on the Windows 11 platform, using the PyTorch architecture, and computations are performed with an RTX 4090 GPU. The detailed experimental setup is depicted in Table 2.

Table 2. Experimental Configuration

parameters	Configuration
CPU	Intel(R) Core(TM) i5-1155G7 @ 2.50GHz
GPU	RTX4090
System Environment	Windows11
Language	Python3.10.14
Acceleration Environment	CUDA11.8

In the training stage of the network model, the batch size was 16, with the decay rate set to 0.0005 and and the overall iteration cycle is 300. The starting learning coefficient was 0.01. The figure 8 shows the training process of the enhanced YOLOv5 network model, with model convergence occurring after approximately 150 iterations. The model's performance was evaluated using evaluation criteria including precision, recall, mean average precision (mAP), and the loss function, as shown in Figure 8.

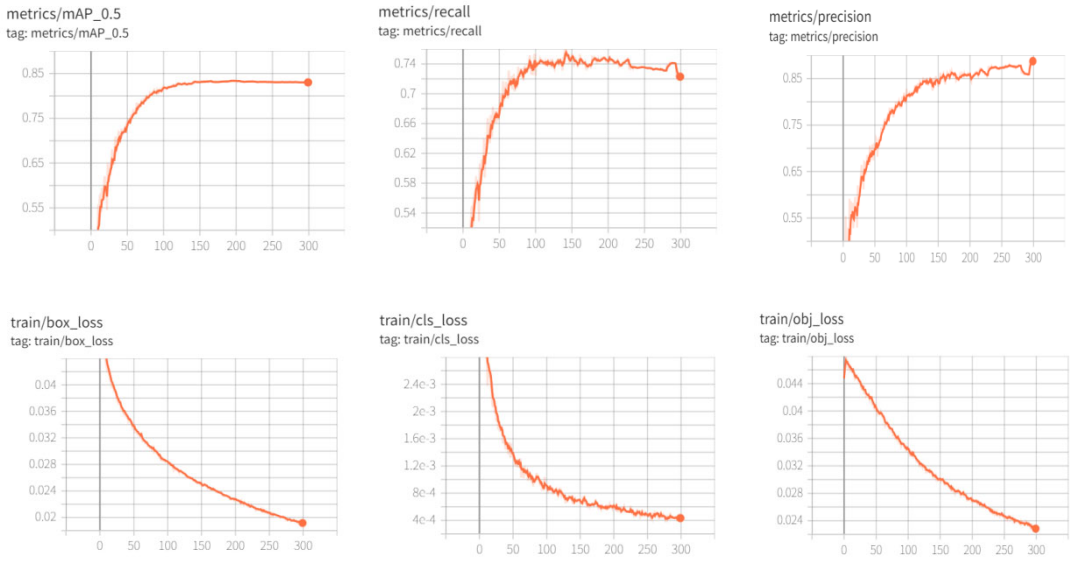

Figure 8. Model Performance Evaluation

The mean average precision (mAP) in Figure 8, depicts the mean value of the average precisions of all target classes. When the model is trained to around 150 iterations, the value approaches

ISSN: 2710-0588

0.83. Recall refers to the probability that the correct class in the instances is correctly predicted, where TT represents the quantity of accurately detected instances of the correct class, and TF represents the quantity of erroneously detected instances of the correct class. At around the 100-th iteration, the recall value approaches 0.74.

$$Recall = \frac{TT}{TT+TF} \tag{7}$$

Precision is interpreted as the percentage of accurately predicted instances to the overall quantity of instances detected as positive, where FT indicates the quantity of erroneously detected samples classified as correct. At around the 150-th iteration, the precision value exceeds 0.85.

$$Precision = \frac{TT}{TT+FT} \tag{8}$$

This study adopts the Generalized Intersection over Union (GIoU) loss function, whose mathematical expression is as follows. From the performance evaluation results of the model during training, it can be seen that the optimized YOLOv5 obtained ideal outcomes during the training phase.

$$giou = iou - \frac{A^c - U}{A^c} \tag{9}$$

4.3. Ablation Experiment and Result Analysis

This paper conducts ablation experiments by setting up different network structure combinations, such as YoloV5+NAM, YoloV5+Slim-neck, and NAM-Slimneck_YoloV5, to compare the contribution of different improvements to the performance enhancement of the algorithm. The trail outcomes are shown in Table 3.

Table 3. Ablation experiment

model	Precision/%	Recall/%	mAP@0.5/%	mAP@0.5:0.95/%	FLOPs
YOLOv5	0.865	0.74	0.83	0.622	16.0
YOLOv5+Slimneck	0.878	0.729	0.828	0.631	14.6
YOLOv5+NAM	0.882	0.736	0.831	0.622	16.0
Enhanced algorithm	0.889	0.722	0.83	0.628	14.6

As shown in Table 4, integrating Slim-neck to Substitute the neck network in YOLOv5s results in a 1.3% increase in precision, and a 8.8% reduction in FLOPs. Introducing the NAM attention mechanism into the YOLOv5 neck structure leads to an increase of 1.7% in precision, and a 0.01% increase in mAP@0.5. The improved YOLOv5 network model (NAM-Slimneck_YOLOv5) shows a 3.34 % increase in precision, with a 8.8 % reduction in FLOPs.

This paper introduces the NAM mechanism and the Slim-neck lightweight network into the initial YOLOv5s architecture. Through ablation experiments, the results indicate that the enhanced NAM-Slimneck_YOLOv5 model significantly improves precision compared to the original model, while the network size is optimized, enhancing its suitability for real-time accurate recognition of forest fires and deployment on hardware.

ISSN: 2710-0588

5. Conclusion

This study optimizes the neck architecture of the YOLOv5s model and proposes a lightweight YOLOv5s-based forest fire detection algorithm with an improved neck design. The proposed algorithm integrates the NAM attention mechanism into the original YOLOv5s neck and adopts a lightweight Slim-neck architecture. Experimental findings demonstrate that the optimized YOLOv5s network achieves 88.9% in precision, 72.2% in recall, and 83% in mAP on the evaluation dataset. The GPIO processing time is 14.6 seconds, demonstrating a significant improvement in detection performance while reducing the model size.

References

[1] Pi, J., Liu, Y.H., Li, J.H. (2023) Research on lightweight forest fire detection algorithm based on YOLOv5s. Journal of Graphics, 44(01): 26-32.

[2] Yang, W., Yu, H.Y., Zhao, X.Y., et al. (2024) Forest fire detection algorithm based on reparameterized YOLOv5s. Radio Engineerinsg, 54(02): 284-293.

[3] Qian, C.S., Shen, Y.W., Sun, N., et al. (2023) Research on mountain fire detection method based on transformer improved YOLOv5. Electronic Measurement Technology, 46(16): 46-56.

[4] Li, J.W., Tang, H., Li, X.D., et al. (2024) LEF-YOLO: A lightweight method for intelligent detection of four extreme wildfires based on the YOLO framework. International Journal of Wildland Fire, 33(01).

[5] Li, H.L., Li, J., Wei, H.B., et al. (2022) Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. Journal of Real-Time Image Processing, 21.

[6] L, Y.C., Shao, Z.R., Teng, Y.Y., et al. (2021) NAM: Normalization-based Attention Module. arXiv preprint arXiv:2111.12419.

[7] Redmon, J., Farhadi, A. (2018) YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767.