FCLGYOLO: Feature Constraint and Local Guided Global Feature for Fire Detection in Unmanned Aerial Vehicle Imagery

Dong Ren, Yang Zhang, Lu Wang, Hang Sun, Shun Ren, and Jian Gu

***Abstract*—Recently, the use of unmanned aerial vehicle (UAV) imagery for object detection in forest fire detection has gained significant attention and has shown remarkable performance. However, most existing object detection models have neglected the exploration of relationships between positive sample features, which is crucial for learning more representative and color-robust features. In addition, small objects in UAV images poses challenges in capturing sufficient object information and hinders accurate object detection. To address these issues, we propose FCLGYOLO that aims to constrain positive sample features and enrich the object information in the feature maps. Specifically, a feature invariance and covariance constraint structure proposed to maintain feature invariance among positive samples and remove internal correlations. Furthermore, a local guided global module proposed to enrich object positioning and semantic information in the feature map, which leverages local features that focus on spatial information to facilitate the learning of global features that focus on frequency information. It is interesting to show that FCLGYOLO performs well even in the presence of heavy smoke or tree occlusions. Compared with multiple state-of-the-art object detection models on a forest fire dataset, experimental results demonstrate the superiority of FCLGYOLO.**

***Index Terms*—Feature constraint, Fourier transform, global feature, local feature, object detection.**

I. Introduction

FOREST fires pose a significant threat to forest ecosystems, property and personal safety. In recent years, The combination of unmanned aerial vehicle (UAV) imagery and object detection algorithms has emerged as a promising approach in this field, benefiting from the advancements in UAV technology [1], [2], [3], [4] and computer vision [5], [6], [7], [8]. There has been a growing interest and research focus on UAV and computer vision techniques to detect forest fires promptly and accurately [9], [10], [11]. Object detection is a fundamental task in computer vision, involving the classification and localization of objects in images [12], [13], [14]. While object detection algorithms have achieved considerable success in natural images, the task becomes highly challenging when dealing with images observed from UAVs. Detecting fires in such scenarios is particularly difficult due to factors, such as smoke or obstruction from trees, as well as the small scale at which fires are often captured by UAVs. These factors pose limitations on the performance of fire detection algorithms.

To meet the requirements of deplorability on UAVs and real-time detection, current mainstream forest fire detectors are lightweight and high-speed single-stage object detectors. The training process of a single-stage object detector [15] is depicted in Fig. 1(a), relying solely on detection labels to train the entire network. Some detectors incorporate auxiliary branches during training to facilitate network learning. These auxiliary branches are used for training but are discarded during inference, not increasing the inference cost. Fig. 1(b) illustrates how certain detectors utilize features from the backbone network or fused features from the neck [16] to perform auxiliary tasks, such as semantic segmentation [17] and super-resolution [18]. These detectors utilize both detection labels and supplementary labels from auxiliary tasks to enhance detection performance. However, they mainly focus on label-dependent information and overlook the relationships between features. Fig. 1(c) shows self-supervised learning models [19], [20], [21], which employ two different augmented views of the same image to train an encoder that can extract features for downstream tasks, by assuming that the feature representations from different views of the same image are similar. Motivated by self-supervised learning, We propose the feature invariance and covariance constraint (FICC) structure, as illustrated in Fig. 1(d). FICC structure leverages feature relationships as additional supervisory signals to assist the detector's learning. Unlike self-supervised learning model, object detector utilize labels to train networks for detecting and classifying objects. Furthermore, while self-supervised learning models mainly focus on capturing global features containing information from the entire image, object detector prioritize local features of individual objects. Considering these differences, the FICC structure explicitly constrains positive sample features to reduce the impact of fire feature variations caused by occlusions.

Manuscript received 15 November 2023; revised 28 December 2023; accepted 22 January 2024. Date of publication 25 January 2024; date of current version 8 March 2024. This work was supported by the Natural Science Foundation of Hubei Province of China under Grant 2021CFB004. *(Dong Ren and Hang Sun are co-first authors.) (Corresponding author: Lu Wang.)*

Dong Ren, Yang Zhang, Lu Wang, Hang Sun, and Shun Ren are with the Hubei Key Laboratory of Intelligent Vision Based Monitoring for Hydroelectric Engineering, China Three Gorges University, Yichang 443002, China (e-mail: rendong5227@163.com; zhang_yang@ctgu.edu.cn; wanglul@ctgu.edu.cn; sunhang0418@whu.edu.cn; renshun@ctgu.edu.cn).

Jian Gu is with Yichang City Forestry Comprehensive Law Enforcement Detachment, Yichang 443299, China (e-mail: 316168318@qq.com).

Code is available online at https://www.github.com/zhangshao249/FCLGYOLO.

Digital Object Identifier 10.1109/JSTARS.2024.3358544

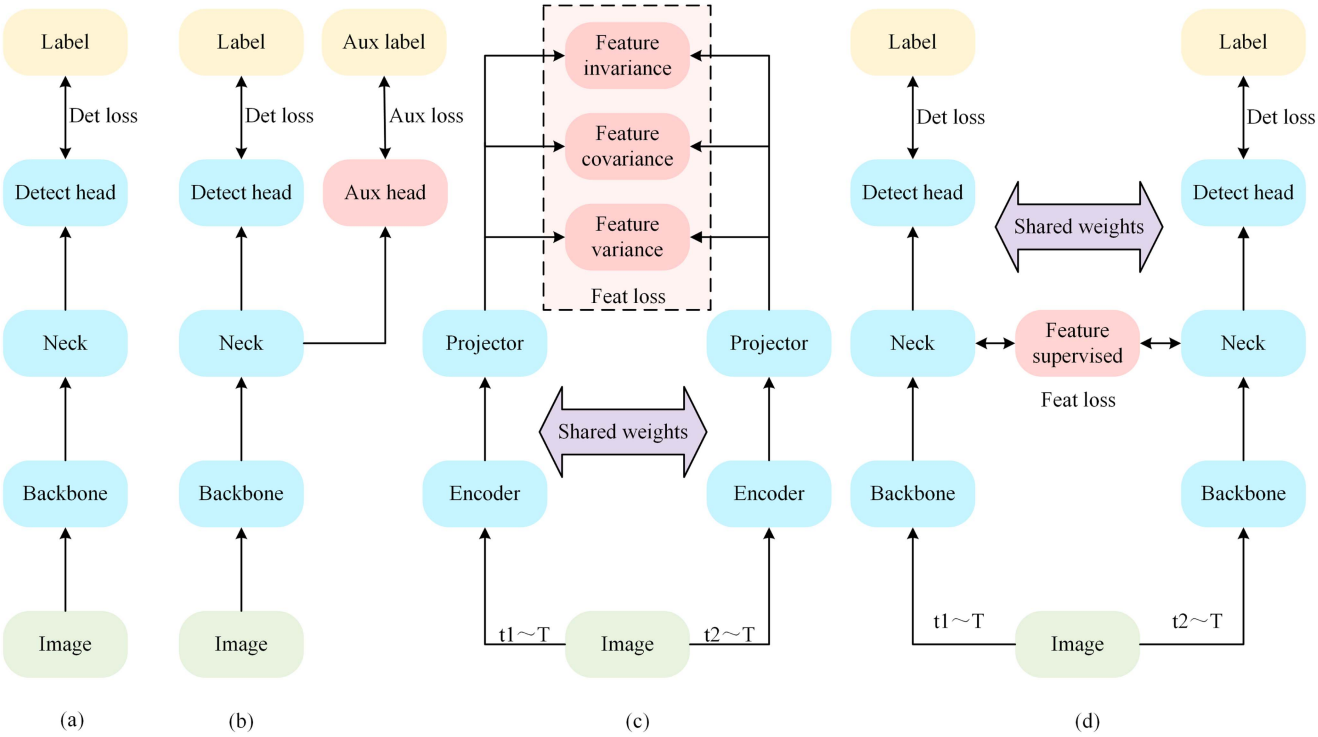

Fig. 1. Single-stage detector, detector with auxiliary task, self-supervised learning model and our model. (a) General structure of a single-stage object detector, where the entire model is supervised by detection labels. (b) Detector with auxiliary task, where the entire model is supervised by both detection labels and auxiliary task labels. (c) Self-supervised learning model, where the entire network is supervised by the relationships between features. (d) Our method, which simultaneously learns from detection labels and feature relationships.

Due to the small scale of fires, the positive sample corresponding to the center point of the object contains rich object information in the feature map. In contrast, the other two positive sample features may contain less object information or not contain any object information at all. Although higher-level feature maps are fused in the neck to expand the receptive field of the current layer's feature map [22], [23], [24], [25], [26], [27], these higher-level feature maps mainly focus on object semantic information and may not adequately address the positional information requirements of the object localization task. To compensate for the lack of object positional information in the feature maps, local guided global module (LGGM) has been proposed to complement the missing object positional information in the feature maps. In LGGM, the frequency module expands the receptive field to the entire image through Fourier transform [28], [29] and extracts global features that focus on frequency information. The spatial module extracts local features that emphasize spatial information. Then, the local features extracted by the spatial module guide the learning of global features extracted by the frequency module to get rich positional information. In summary, our main contributions are as follows.

1) A well-designed FICC structure explicitly incorporates constraints on object features to ensure invariance of the same object features and covariance within features. This design enhances the model's robustness, especially in the presence of smoke influences.
2) A well-designed LGGM enhances the discriminability of positive sample features by enhancing the semantic and positional information of objects in the feature map, resulting in significant improvement in the model's resistance to occlusion caused by trees.
3) A novel model has been proposed that effectively integrates FICC structure and LGGM for forest fire detection. This integration enhances the discriminability and saliency of the extracted object features, resulting in our model achieving the best performance on forest fire datasets.

II. Related Work

A. Fire Detection

Visual-based fire detection methods can be categorized into two groups: traditional machine learning methods and deep learning methods. Thepade et al. [30] utilized the LUV color space has shown promising results in reducing false alarm rates in fire detection. Chowdhury et al. [31] combined visual sensors with smoke sensors can provide more effective fire detection. Traditional machine learning methods heavily rely on handcrafted features and support vector machines (SVMs) for fire detection [32]. However, these methods are susceptible to false alarms caused by objects with similar colors to fire. With the development of computer vision, a deep classification network has been employed to classify the presence or absence

of fire in images [33]. Mseddi et al. [34] combined YOLOv5 with U-Net to accomplish fire detection and segmentation tasks. While fire image classification is a coarse-grained method that cannot provide accurate flame localization, object detection methods can determine both the presence of fire and the precise location of flames in an image. Furthermore, fire segmentation aims at classifying each pixel in the image as either fire or non-fire [35]. However, detection tasks align more closely with the practical requirements of fire detection, as the primary methods widely used are still based on object detection methods.

Object detector can be broadly categorized into two groups: CNN-based detector and transformer-based [36] detector. CNN-based detectors encompass popular models, such as R-CNN [37], RetinaNet [13], Faster R-CNN [14], and YOLOv5 [15]. R-CNN utilizes deep neural networks to extract features, which are then used by SVMs and bounding box regressors for classification and bounding box regression tasks, respectively. Single-stage detectors [13], [15] streamline the workflow of two-stage detectors [14] by eliminating the need for a region proposal network. Instead, they directly generate prediction boxes and predicted class on the feature map. Transformer-based detectors include DETR [38] and its variants [39], [40], [41]. They have shown significant improvements in detection accuracy and convergence speed on the COCO dataset [42]. However, CNN-based detectors have unique advantages, such as compact size, requiring less training data, and fast inference speed. CNN-based detectors still dominate in practical applications. Although CNN-based detectors are mature, they still have limitations in specific tasks, such as forest fire, detection in UAVs.

B. Self-Supervised Learning

Self-supervised learning [43], [44] is a process of training a generic feature extractor using a large amount of unlabeled data, which can be used for various downstream tasks. Self-supervised learning originated from the field of natural language processing [45] and was initially introduced to the computer vision domain through the definition of various pretext tasks. Examples of these pretext tasks include: converting RGB images to grayscale and then training a network to reconstruct the original RGB images using the grayscale [46]; dividing images into patches, randomly shuffling the patches, and training a network to predict the relative positions of each patch [47]; training a generator to generate high-resolution images from low-resolution input images, while also training a discriminator to distinguish between real high-resolution images and high-resolution images generated by the generator [48]. With the development of self-supervised learning, the predefined tasks have now been unified into two types: image inpainting or reconstruction of missing or distorted parts, and contrastive learning of different augmented views of the same image. Image inpainting or reconstruction is mainly used to train models related to transformer, while contrastive learning of different augmented views is mainly used for encoder models related to CNNs. The two different augmented view of the same image are considered as a positive sample pair, while any pair of different images' augmented views is considered as a negative sample pair. Such as, SimCLR [19], MoCo [20], and InstDisc [49], aim to bring the feature representations of positive sample pairs closer and push away the feature representations of negative sample pairs to learn meaningful visual feature representations. These methods have slight difference. BYOL [50] and SimSiam [51] only require bringing the feature representations of positive sample pairs closer to learn meaningful visual feature representations. The VICReg [21] algorithm is based on the covariance matrix of the features from two augmented views. The variance along each feature representation dimension is regularized to prevent feature collapse. The invariance term ensures that the features from different augmented views are encoded into similar representations, while the covariance term encourages the encoding of different feature information across different dimensions. The FICC structure is inspired by self-supervised learning and uses feature constraint to facilitate the learning of detection networks.

C. Fourier-Based Networks

Recently, Fourier transform has been introduced into deep learning for analyzing the optimization and generalization abilities of DNNs [52]. In addition, Yang et al. [53] applied Fourier transform to both source and target domains, exchanging their low-frequency signals to reduce the distribution discrepancy between the two domains, thus achieving domain adaptation. Xu et al. [54] suggested that phase information contains high-level semantic information and is less susceptible to domain shift. Based on this, they design amplitude-mixing data augmentation to strengthen the learning of domain-invariant information, thereby improving the model's generalization ability. Rao et al. [28] transformed feature maps into the frequency domain and multiplied them with learnable convolutional kernels to obtain a global receptive field. Fuoli et al. [55] designed a frequency-domain supervised loss to supervise a super-resolution network to learn high-frequency information that is easily lost during the super-resolution process. Jiang et al. [56] introduced a frequency-domain focal loss by reducing the weight of simple components in the frequency domain, encouraging the reconstruction network to focus on learning difficult frequency-domain components. Yu et al. [57] designed an amplitude-guided phase module in the frequency domain and a global-guided local module in the spatial domain to jointly accomplish the task of image dehazing. Huang et al. [58] fully utilized frequency-domain information to restore brightness and structural details in overexposed images for image restoration. Wang et al. [29] proposed a novel network architecture that extracts frequency and spatial information separately using frequency branches and spatial branches. It then uses multi-head attention mechanisms to fuse the frequency and spatial information, reconstructs the high-resolution image using the fused features, and uses the frequency features to separately reconstruct the phase and amplitude parts of the high-resolution image.

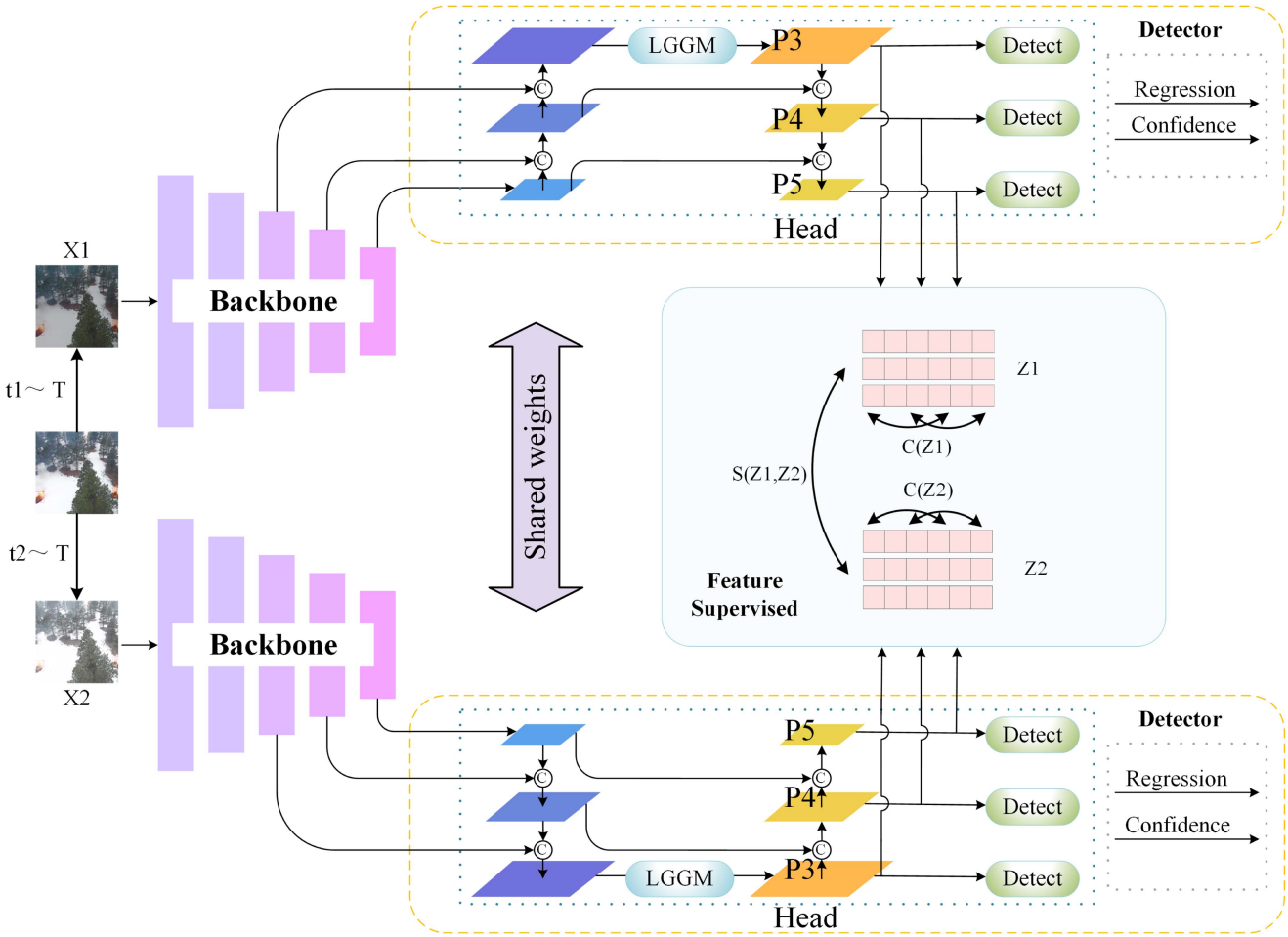

Fig. 2. FICC structure, which comprises two identical networks sharing the same architecture and parameters, along with a feature supervision module. Each network receives two distinct image augmentation views of the same image and is supervised by its respective detection labels. The fused features from the neck of the two networks undergo additional constraint through the feature supervision module, facilitating the overall network learning process. It is important to note that the figure does not depict the auxiliary head, and a detailed explanation of the LGGM depicted in the figure will be presented in Section III-B.

III. Methods

The FICC structure encompasses two identical detection networks along with a feature supervision module, which will be thoroughly elucidated in Section III-A. During the inference phase, solely one detection network is utilized to predict, while another detection network and the feature supervision module are disregarded. On the other hand, the LGGM consists of three essential blocks: frequency block, spatial block, and fuse block, which will be comprehensively expounded upon in Section III-B. During the inference stage, only the frequency block remains active, whereas the spatial block and fuse block are not employed.

A. FICC Structure

The FICC structure as illustrated in Fig. 2, each detection network consists of three components: a backbone network for extracting multiscale features, denoted as $F = \{F_l \in R^{C_l \times H_l \times W_l}\}$, where $l \in \{3, 4, 5\}$ denotes the feature pyramid levels, from the input image $X \in R^{C \times H \times W}$, where C represents the channel dimension of the image and (H, W) represent the spatial dimensions of the image; a neck for fusing the multiscale features F; and a detection head for generating prediction results. The dimensions (C_l, H_l, W_l) are generally equal to $(C_0 \times 2^l, \lfloor H/2^l \rfloor, \lfloor W/2^l \rfloor)$. Image X is partitioned into grids of uniform size, with grid corresponding to a size of $2^l \times 2^l$ at level l. Consequently, after downsampling by the backbone, each grid is represented as a single point in the feature map. Any point $P \in R^{C_l}$ in the feature map F_l is decoded by the detection head into prediction boxes. In the training stage, all points in the feature map F_l are matched with objects via the label assignment strategy, and will be categorized into two groups: positive samples and negative samples. Positive samples correspond to points that are matched with at least one object, while negative samples do not have any matches. Consequently, positive samples are responsible for predicting the matched objects, while negative samples do not need contribute to the object predict.

Image X has corresponding labels $= \{A_1, A_2, \ldots, A_k\}$, where k is the number of objects in image X, $A_i = (\text{cls}, \text{box})$, $\text{cls} \in \{0, 1, 2, \ldots, n-1\}$ represents the class of the object, $\text{box} \in R^4$ represents the bounding box, and n is the number of object classes to be detected. In this case, we detect only

one class, which is fire. Here, $n = 1$. Assuming that the positive sample corresponding to the center point of object A on the feature map F_l is denoted as a_1, and the other two positive samples in the feature map F_l are denoted as a_2 and a_3, where a_1, a_2, and $a_3 \in R^{C_l}$. During the training process, these positive samples a_1, a_2, and a_3 is decoded into prediction boxes offsets (t_x, t_y, t_w, t_h) and class probabilities by the detection head. To obtain the predicted box (b_x, b_y, b_w, b_h), the following formula can be utilized:

$$\begin{cases} b_x = c_x + \sigma(t_x) \times 2 - 0.5 \\ b_y = c_y + \sigma(t_y) \times 2 - 0.5 \\ b_w = p_w \times (2 \times \sigma(t_w))^2 \\ b_h = p_h \times (2 \times \sigma(t_h))^2 \\ \sigma(x) = \frac{1}{1+e^{-x}} \end{cases} \tag{1}$$

where c_x and c_y denote the coordinates of the top-left position of the grid that contains the object feature, while p_w and p_h represent the width and height of a predefined anchor box. These prediction boxes will then be supervised by the ground truth bounding box and class label of object A, which is a common in all object detector.

Image augmentation distribution is denoted as T, and two random image augmentation operations t_1 and t_2 are sampled from T. The image X is then transformed by t_1 and t_2, resulting in two augmented views of X, $X_1 = t_1(X)$ and $X_2 = t_2(X)$. Similarly, we obtain the feature maps $F_{l1} = \text{neck}(\text{backbone}(X_1))$ and $F_{l2} = \text{neck}(\text{backbone}(X_2))$ for each level l. Taking F_{31} and F_{32} as illustrative examples, let us consider an object $A_i = (\text{cls}_i, \text{box}_i)$, with the positive samples corresponding to the center point of A_i in F_{31} and F_{32} denoted as $a_{(3i)1}$ and $a_{(3i)2}$, respectively. The other two positive samples are represented by $(a_{(3i+1)1}, a_{(3i+2)1})$ and $(a_{(3i+1)2}, a_{(3i+2)2})$. In F_{31}, there is a unique possibility where $a_{(3i)1}$, $a_{(3i+1)1}$, and $a_{(3i+2)1}$ are decoded by the detection head as the prediction boxes for object A_i. Hence, we can regard $a_{(3i)1}$, $a_{(3i+1)1}$, and $a_{(3i+2)1}$ as the three feature representations of object A_i in F_{31}. In a similar manner, we can consider $a_{(3i)2}$, $a_{(3i+1)2}$, and $a_{(3i+2)2}$ as the three feature representations of object A_i in F_{32}. As the three feature representations of object A_i, $a_{(3i)1}$, $a_{(3i+1)1}$, and $a_{(3i+2)1}$ occupy distinct grids, the position offsets (t_x, t_y) required by the detection head for each feature representation must also differ. Consequently, if we desire the prediction boxes corresponding to these feature representations to approximate the ground truth box, there need to be discernible variations among the three feature representations $a_{(3i)1}$, $a_{(3i+1)1}$, and $a_{(3i+2)1}$.The feature representations of object A_i in different views, $(a_{(3i)1}, a_{(3i)2})$, $(a_{(3i+1)1}, a_{(3i+1)2})$, and $(a_{(3i+2)1}, a_{(3i+2)2})$ are located in the same grid. If the same feature representation is decoded by the detection head, it will inevitably result in the same box offsets (t_x, t_y). Therefore, these three pairs of feature representations should be identity.

The object feature representations in feature map F_{31} are arranged into an object feature representation matrix denoted as $Y_1 = [a_{01}, a_{11}, a_{21}, \ldots, a_{(3n-1)1}]$. Similarly, the object feature representation matrix in feature map F_{32} is denoted as $Y_2 = [a_{02}, a_{12}, a_{22}, \ldots, a_{(3n-1)2}]$. Here, Y_1 and Y_2 are matrices in $R^{3n \times C_3}$, where n represents the number of objects and C_3 denotes the channel dimension of feature maps F_{31} and F_{32}. After applying the projector, Y_1 and Y_2 undergo projection processes resulting in their corresponding embedding matrices denoted as $Z_1 = [z_{01}, z_{11}, z_{21}, \ldots, z_{(3n-1)1}]$ and $Z_2 = [z_{02}, z_{12}, z_{22}, \ldots, z_{(3n-1)2}]$, where $Z_1, Z_2 \in R^{3n \times C_3}$. We impose a invariance constraint on the embedding matrices Z_1 and Z_2

$$S(Z_1, Z_2) = \frac{1}{3n} \sum_{i=0}^{3n-1} (z_{i1} - z_{i2})^2. \tag{2}$$

Subsequently, we impose a covariance constraint, which aims to decorrelate the different dimensions of the features and prevent them from encoding similar information, on the embedding matrices Z_1 and Z_2

$$C(Z) = \frac{1}{3n-1} \sum_{i \neq j} [Cm(Z)]_{i,j}^2 \tag{3}$$

where Cm is to get the covariance matrix defined by

$$Cm(Z) = \frac{1}{3n-1} \sum_{i=0}^{3n-1} (z_i - \bar{z})^T (z_i - \bar{z}) \tag{4}$$

$$\text{where} \quad \bar{z} = \frac{1}{3n-1} \sum_{i=0}^{3n-1} z_i. \tag{5}$$

The final feature supervised loss is

$$\text{Loss}_{\text{feat}} = \alpha \times S(Z_1, Z_2) + \beta \times [C(Z_1) + C(Z_2)] \tag{6}$$

where α and β are the weights of the two terms $\alpha = 1$ and $\beta = 25$ following Bardes et al. [21]. This component draws inspiration from self-supervised learning and serves as an additional feature supervision mechanism.

B. Local Guided Global Module

The LGGM is introduced in the neck to supplement the object information to positive samples in the feature maps. Benefiting from the Fourier transform, amplitude component and phase component can capture the image-size receptive field. Notably, the phase component offers a wealth of structural information that significantly aids in object discrimination. Table III tabulates that the forest fire dataset contains a considerable number of small objects that is primarily detected in the P3. Consequently, the LGGM is exclusively inserted before the P3 to enhance the model's ability to detect small objects.

To optimize the performance of the frequency block while minimizing the associated increment in parameters and computations in the inference stage, we formulated the LGGM, as illustrated in Fig. 3(a). During the training stage, the spatial block and fuse block modules play a pivotal role in guiding the learning process of the frequency block. The spatial block extracts local features enriched with spatial information, while the frequency block captures global features enriched with frequency information. Furthermore, the fuse block integrates the features extracted from both the spatial block and frequency block, and in conjunction with the auxiliary head, facilitates the learning process of the frequency block. Serving as the

TABLE I
INFORMATION OF DATASET SOURCE

Source	Resolution	Aircraft	Camera
FLAME [60]	3840×2160	Phantom 3 Professional	Phantom 3 camera
FLAME2 [61]	1920×1080	Mavic 2 Enterprise Advanced	M2EA Visual Camera

TABLE II
TECHNICAL SPECIFICATION OF CAMERAS

Camera	FOV	35 mm format equivalent	Effective Pixels
Phantom 3 camera	94°	20 mm	12.4 M
M2EA Visual Camera	84°	24 mm	48 M

TABLE III
DATASET INFORMATION

	Image	Fire	Large	Middle	Small	Occlusion
train	3565	15 252	1436 (9.4%)	5037 (33.0%)	8779 (57.6%)	3213 (21.1%)
val	1018	4350	379 (8.7%)	1431 (32.9%)	2540 (58.4%)	966 (22.2%)
test	509	2174	210 (9.7%)	687 (31.6%)	1277 (58.7%)	501 (23.0%)

TABLE IV
COMPARISON WITH OTHER DETECTION MODELS

Method	Params	GFLOPs	mAP50
Faster R-CNN [14]	41.3M	203.0G	61.7%
ATSS [62]	32.1M	203.0G	72.1%
AutoAssign [63]	36.2M	199.0G	78.4%
FCOS [64]	32.1M	198.0G	68.9%
RetinaNet [13]	36.3M	201.0G	61.8%
FreeAnchor [65]	36.3M	206.0G	77.9%
DINO [66]	47.5M	274.5G	78.4%
VarifocalNet [67]	32.7M	199.0G	71.0%
FCLGYOLO(Ours)	**7.0M**	**16.4G**	**81.1%**

cornerstone of the LGGM, the frequency block solely operates during the inference stage.

In the following sections, we will provide a comprehensive explanation of the frequency block, spatial block, and fuse block, which are integral components of the LGGM:

$$F_{\text{freIn}} = \text{FreBlock}(F) \tag{7}$$

$$F_{\text{spaIn}} = \text{SpaBlock}(F) \tag{8}$$

$$F_{\text{out}} = \text{FuseBlock}(F_{\text{freIn}}, F_{\text{spaIn}}) \tag{9}$$

where $F \in R^{C_l \times H \times W}$ represents the feature map.

The frequency block is implemented as outlined below. First, the feature map F is converted to the frequency domain by applying the 2-D Fourier transform, expressed as $\text{fre} = \text{fft2d}(F)$. Subsequently, the phase and amplitude components are acquired individually

$$\text{pha} = \text{abs}(\text{fre}), \text{amp} = \text{angle}(\text{fre}). \tag{10}$$

The phase and amplitude components undergo separate enhancement via two convolutional layers with a kernel size of 3×3

$$\text{pha}_{\text{ref}} = \text{Conv2d}(\text{ReLU}(\text{Conv2d}(\text{pha}))) \tag{11}$$

$$\text{amp}_{\text{ref}} = \text{Conv2d}(\text{ReLU}(\text{Conv2d}(\text{amp}))). \tag{12}$$

The enhanced feature representation in the frequency domain is obtained by combining the enhanced phase and amplitude

$$\begin{cases} \text{real} = \text{amp}_{\text{ref}} \times \cos(\text{pha}_{\text{ref}}) \\ \text{imag} = \text{amp}_{\text{ref}} \times \sin(\text{pha}_{\text{ref}}) \end{cases} \tag{13}$$

$$\text{fre}_{\text{ref}} = \text{complex}(\text{real}, \text{imag}). \tag{14}$$

Finally, the feature map with attention to frequency information and global receptive field is obtained by inverse 2-D Fourier transform

$$F_{\text{freIn}} = \text{abs}(\text{ifft2d}(\text{fre}_{\text{ref}})) + F. \tag{15}$$

The implementation of the spatial block is as follows:

$$F_{\text{spaIn}} = \text{Conv2d}(\text{Res}(\text{ReLU}(\text{Conv2d}(F)))) \tag{16}$$

where Res is the residual block defined by

$$\text{Res}(F) = \text{Conv2d}(\text{ReLU}(\text{Conv2d}(F))) + F. \tag{17}$$

The spatial block consists of two convolutional layers and a residual block is designed to extract feature with attention to local spatial information.

The implementation of the fuse block is as follows:

$$F_{\text{fre}} = \text{ReLU}(\text{Conv2d}(F_{\text{freIn}})) \tag{18}$$

$$F_{\text{spa}} = \text{ReLU}(\text{Conv2d}(F_{\text{spaIn}})) \tag{19}$$

$$F_{\text{cat}} = \text{Concat}(F_{\text{fre}}, F_{\text{spa}}) \tag{20}$$

$$F_{\text{fuse}} = \text{TokenMixer}(F_{\text{cat}}) \tag{21}$$

$$F_{\text{out}} = \text{Dropout}(\text{ReLU}(\text{Conv2d}(F_{\text{fuse}}))) \tag{22}$$

where TokenMixer is defined by

$$\text{TokenMixer}(F) = \Re(\text{fft}(F)) \tag{23}$$

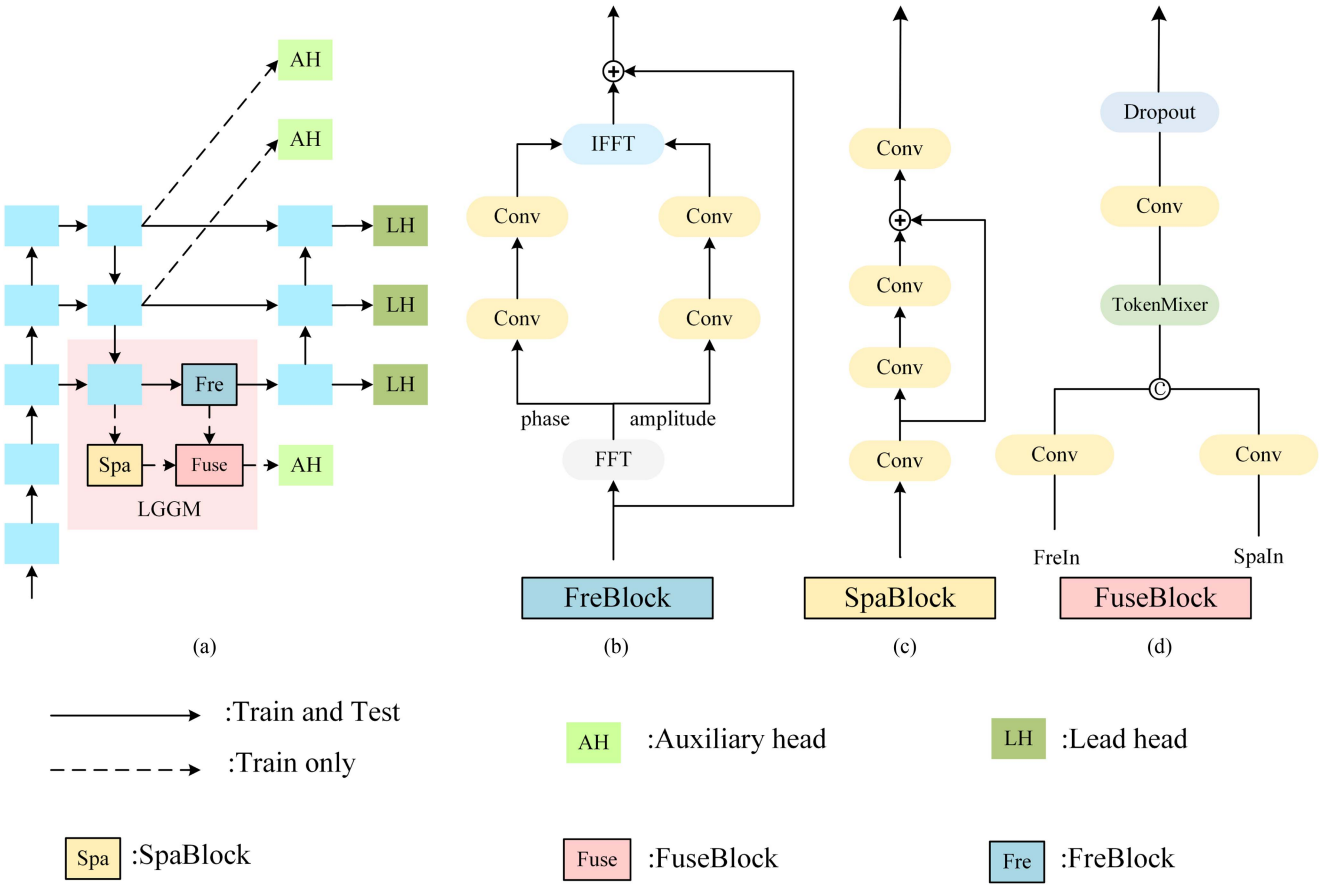

Fig. 3. (a) Illustrates detector with LGGM. (b) Detailed structure of the frequency block. (c) Detailed structure of the spatial block. (d) detailed structure of the fuse block. In the figure, "Fre" and "Freblock" both represent the frequency block, while "Spa" and "SpaBlock" refer to the spatial block.

where the function $\Re(x)$ denotes extracting the real part of the result following the Fourier transform. The fuse block simultaneously takes the feature maps from the spatial block and the frequency block. The two feature maps are individually passed through separate convolutional layers, then concatenated along the channel dimension. The concatenated feature map undergoes reshaping along the H and W dimensions are flattened into a single dimension HW, subsequently being passed into the token mixer layer. Within the token mixer layer, a complete integration of the spatial and frequency information in the feature map is achieved. The token mixer layer plays a crucial role in integrating frequency and spatial information. This is accomplished by extracting the real part of the results obtained through Fourier transform. This operation serves as a viable substitute for multihead attention [59], offering a similar effect in terms of information integration but without any learnable parameters. As a result, it leads to a substantial reduction in the model's computational complexity and alleviates the training load on the network.

C. Loss

The complete loss consists of three parts: the loss of the main detection heads of two branches of the network, the loss of the auxiliary detection heads of two branches of the network, and the feature supervision loss. The expressions are as follows:

$$\text{Loss} = \text{Loss}_{\text{det}} + W_a \times \text{Loss}_{\text{aux}} + W_f \times \text{Loss}_{\text{feat}} \tag{24}$$

where W_a and W_f correspond to the weights assigned to the loss of the auxiliary detection head and the feature supervision loss, respectively. These weights are employed to achieve a balanced importance among the various losses. The loss of lead detection head and the auxiliary detection head is as follows:

$$\text{Loss}_{\text{det}} = \gamma \times \text{Loss}_{\text{pos}} + \eta \times \text{Loss}_{\text{obj}} \tag{25}$$

$$\text{Loss}_{\text{aux}} = \gamma \times \text{Loss}_{\text{apos}} + \eta \times \text{Loss}_{a\text{obj}} \tag{26}$$

where γ and η refer to the weights assigned to the prediction box loss and the target loss, Setting $\gamma = 0.05$ and $\eta = 1.0$. Since the forest fire dataset contains only a single class of target, there is no classification loss component in either the main detection head or the detection head assistance loss. The detailed expression of $\text{Loss}_{\text{feat}}$ can be found in (6).

IV. Experimental Results

A. Dataset

The dataset was obtained from FLAME [60] and FLAME2 [61], consisting of video recordings with resolutions of 3840 × 2160 and 1920 × 1080 pixels. The hardware to collect

the dataset are detailed in the Tables I and II. The correlation between consecutive frames can aid in the detection of forest fires. However, our primary focus is on identifying the presence of a fire in each individual frame, and considering the interframe relationship regarding fires is unnecessary and would require additional computational resources. As a result, we extracted the frames from the original videos and randomly selected 2533 and 2559 images containing fires, respectively. Hopkins et al. [61] provided timestamps and latitude–longitude information in the top-left corner of the videos. While retaining the original aspect ratio of 1920:1080, we aimed to preserve as much of the image content as possible when removing the timestamps and latitude–longitude information. This process resulted in an image resolution of 1611 $\times$ 907 pixels. Using the labelimg tool, a total of 5092 images were annotated and create a dataset in the YOLO label format. Subsequently, the dataset was divided into training, validation, and testing sets in a ratio of 7:2:1. Please refer to Table III for specific details regarding the dataset.

In Table III, "large," "medium," and "small" present large, medium, and small object, respectively. "Occlusion" column indicates the number of fires that are obscured by smoke. Aligns with the definition provided in [42]: objects smaller than 32 $\times$ 32 pixels are defined as small objects, objects larger than 96 $\times$ 96 pixels are considered large objects, and objects falling within this size range are defined as medium objects. The dataset predominantly consists of small targets, making up close to 60% of the dataset. In Table III, For all experiments conducted in this article, unless explicitly stated otherwise, the dataset used is this dataset. This dataset is available online.[1]

B. Implementation Details

The proposed framework was implemented using PyTorch and runs on an NVIDIA 3090 GPU. The effectiveness of the framework was assessed through ablation experiments and comparative experiments conducted on the Section IV-A dataset. The input image size was set to 640 $\times$ 640 pixels for both the training and inference stages. In the training stage, the input images underwent image augmentation, including transformations, such as hue saturation value adjustments, translation, left–right flips, up–down flips, and mosaic augmentation. Conversely, no image augmentation operations were applied during the inference stage. The optimizer employed was stochastic gradient descent with a momentum value of 0.937 and weight decay of 0.0005. The batch size was specified as 8, the initial learning rate was set to 0.01, and the training process was carried out for 300 epochs.

C. Accuracy Metrics

We employ mean Average Precision at 50% (mAP50) IoU as a metric to compare and evaluate the detection performance of our model. The mAP50 is calculated by considering precision and recall values. The calculations of precision, recall, and mAP50 metrics are defined as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{27}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{28}$$

$$\text{mAP} = \frac{\text{AP}}{N} = \frac{\int_0^1 p(r)dr}{N}. \tag{29}$$

Here, TP, FP, and FN represent the counts of true positive predictions, false positive predictions, and false negative predictions, respectively. In addition, p indicates precision, r indicates recall, and N denotes the number of categories. The precision and recall metrics are associated with the concepts of commission and omission errors, respectively. Furthermore, the mAP50 is a comprehensive indicator obtained by averaging AP values, utilizing an integral method to calculate the area enclosed by the Precision–Recall curve and the coordinate axis of all categories.

Furthermore, we utilize Giga Floating-point Operations (GFLOPs) to quantify the computational complexity of the model, and the parameter size to measure the model's size. Unless stated, otherwise, all tables presented hereafter include GFLOPs and Params, representing the computational complexity and the number of parameters used by the model during the testing stage, respectively.

D. Comparisons With Previous Methods

In order to validate the advantages of our proposed model, we compare FCLGYOLO with advanced detectors, such as Faster R-CNN, ATSS, AutoAssign, FCOS, RetinaNet, FreeAnchor, DINO, and VarifocalNet. As shown in Table IV, DINO and AutoAssign performed the best on fire dataset among these advanced detectors. Our proposed model improved the detection accuracy by 1.7% compared with DINO and AutoAssign. Notably, DINO had 6.8$\times$ the number of parameters and 16.7$\times$ the computational complexity of our proposed model, while AutoAssign had 5.2$\times$ the number of parameters and 12.1$\times$ the computational complexity. This demonstrates the significant advantages of our model in terms of model size, computational complexity, and detection performance. In particular, DINO, which is transformer-based detection method, benefits from the self-attention mechanism, which can achieve global receptive field. LGGM in our proposed FCLGYOLO, shares similarities in achieving global receptive field, yet it is noteworthy that our method exhibits significant advantages in terms of model size and computational complexity.

Furthermore, in Fig. 4, we visualize three detectors that performed well on the fire dataset, along with our proposed model. DINO displayed missed detections in Fig. 4(a), (b), (c), and (e) due to its emphasis on global features while neglecting the local features, leading to the overlooking of small objects. Conversely, when severe occlusion is present in Fig. 4(d), both AutoAssign and FreeAnchor produced multiple overlapping bounding boxes. This suggests an overemphasis on local object features by these models, resulting in incorrect identification of certain fire

[1][Online]. Available: https://www.github.com/zhangshao249/FCLGYOLO

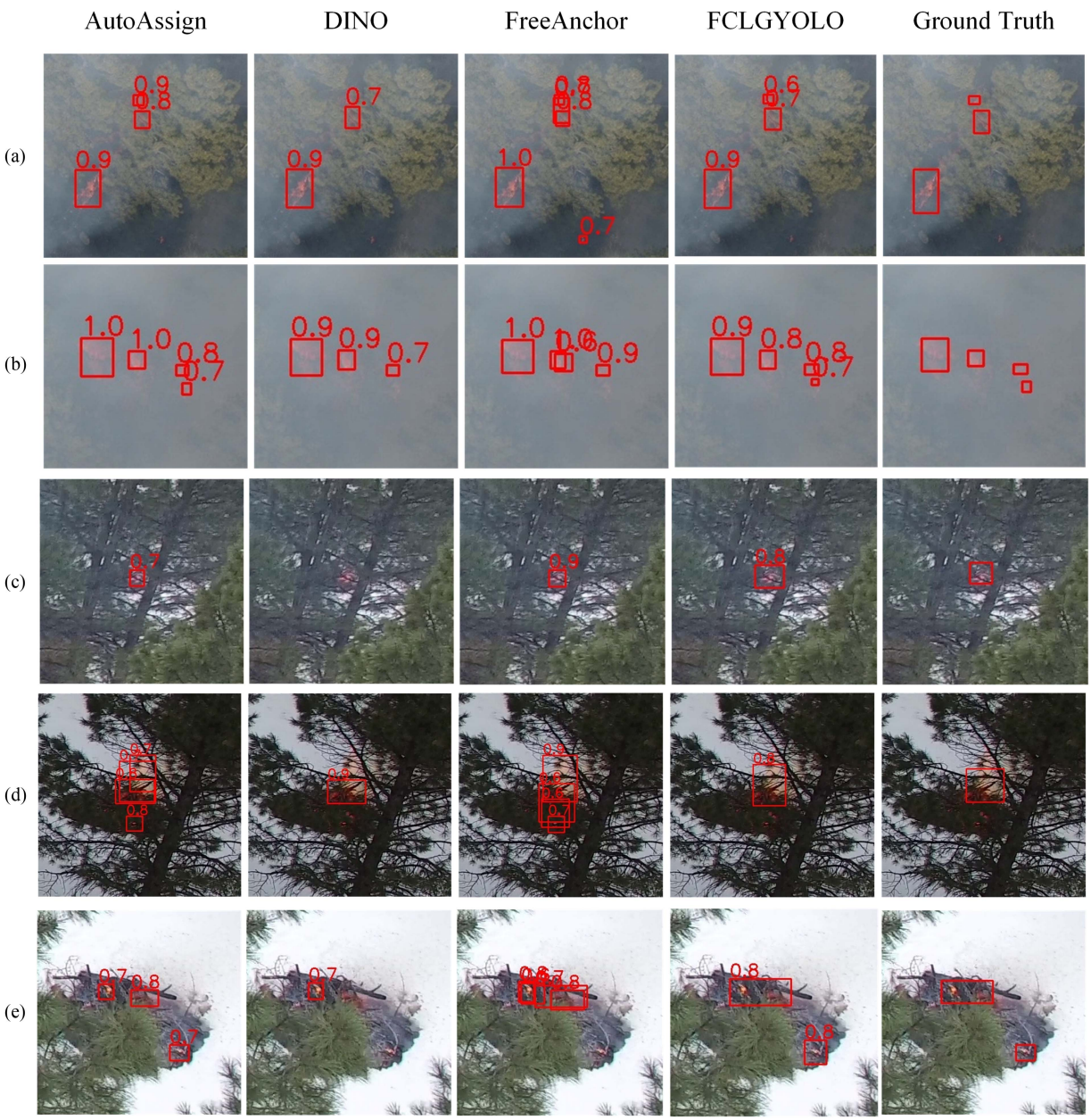

Fig. 4. Visualization results of multiple detectors, which include AutoAssign, DINO, FreeAnchor, and our proposed FCLGYOLO. To better showcase the detection performance of the various detectors, (a)–(e) image patchs were cropped from different images, considering the significant size difference between the images and the targets.

components as complete fires. In contrast, FCLGYOLO exhibited superior performance in such scenarios. The incorporation of the FICC structure improved the detection performance of FCLGYOLO by minimizing the feature disparity between fires exhibiting notable color variations under dense smoke occlusion in Fig. 4(b). Furthermore, in Fig. 4(d) and (e), FCLGYOLO effectively detected severely occluded fires as complete flames, benefiting from the extensive global receptive field of LGGM. This highlights the capability of FCLGYOLO in effectively detecting fires even in challenging scenarios with thick smoke and severe object occlusion.

E. Ablation Study

Table V presents the model size, computational complexity, and testing accuracy for various base models. YOLOv5$\times$ achieves the highest detection accuracy, but it requires $12.3\times$ the number of parameters and $12.8\times$ the computational complexity compared with YOLOv5s. Despite slightly lower accuracy compared with other YOLOv5 models, YOLOv5s offers a significant advantage in terms of fewer parameters and GFLOPs, making it ideal for deployment and real-time detection on UAVs. As a result, YOLOv5s is selected as the baseline model.

TABLE V
COMPARISON OF DETECTION MODELS IN YOLO SERIES

Method	Params	GFLOPs	mAP50
YOLOv4 [68]	9.1M	20.6G	75.0%
YOLOv7 [69]	6.0M	13.2G	63.8%
YOLOx [70]	8.9M	13.1G	78.5%
YOLOv5×	86.2M	204.6G	**80.8%**
YOLOv5l	46.1M	108.2G	80.5%
YOLOv5m	20.9M	48.2G	80.0%
YOLOv5s	**7.0M**	**15.9G**	79.1%

The bold entities highlight the best value.

TABLE VI
IMPACT OF FICC STRUCTURE AND LGGM

FICC	LGGM(AUX)	mAP50
		79.1%
✓		80.3%
	✓	80.2%
✓	✓	81.1%

TABLE VII
IMPACT OF AUXILIARY HEAD IN LGGM

LGGM	AUX	mAP50
		79.1%
✓		79.6%
	✓	79.4%
✓	✓	**80.2%**

The bold entities highlight the best value.

TABLE VIII
IMPACT OF LGGM BEING INSERTED POSITION

Position	P_3	P_4	P_5
mAP50	**80.2%**	79.5%	79.3%

The bold entities highlight the best value.

The detection performance of the baseline model can be enhanced by both FICC and LGGM, as demonstrated in Table VI. The simultaneous usage of FICC and LGGM does not appear to result in any noticeable conflicts. FICC shows a 1.2% improvement in the baseline, while LGGM demonstrates a 1.1% improvement. In an ideal situation, the baseline would experience a 2.3% improvement, but the actual improvement observed is 2.0%. This discrepancy might be attributed to effect of FICC, which primarily focuses on constraining features to enhance model performance. In contrast, LGGM employs an auxiliary detection head to indirectly impose feature constraints. The potential overlap in constraining the intermediate features between FICC and LGGM may interference, thereby hindering the attainment of the desired improvement.

The findings presented in Table VIII align with the analysis conducted in Section III-B. This alignment can be attributed to the dataset containing a significant number of small targets. Specifically, the P_3 demonstrates the highest contribution to the detection of small targets, followed by the P_4. Conversely, the P_5 contributes the least to the detection of small targets.

LGGM is designed to be utilized in conjunction with an auxiliary detection head. When the auxiliary detection head is not present, only the FrequencyBlock in LGGM is active, while the SpatialBlock and FuseBlock do not contribute to the loss function. Consequently, they do not undergo learning during the training process and do not guide the learning of the FrequencyBlock. In Table VII, the performance results demonstrate the impact of adding different components to the network. When only LGGM is incorporated, equivalent to solely adding the FrequencyBlock, there is a performance improvement of +0.5% compared with the baseline. Besides, when only the auxiliary detection head is added, there is a performance improvement of +0.3% compared with the baseline. The most notable improvement is observed when both LGGM and the auxiliary detection head are integrated into the network, leading to a performance enhancement of +1.1% compared with the baseline. By incorporating both LGGM and the auxiliary detection head, the SpatialBlock and FuseBlock aptly learn and offer constructive guidance to the FrequencyBlock throughout the training process.

V. CONCLUSION

The detection of forest fires in UAV imagery poses significant challenges, particularly in terms of considering both speed and accuracy. We propose FCLGYOLO, a novel approach that tackles issues related to smoke occlusion and small objects in forest fire detection tasks. FCLGYOLO incorporates the FICC structure to constrain features based on object feature relationships, while LGMM expands the network's receptive field and utilizes local features to guide global features. Notably, the FICC structure does not introduce any additional inference cost and does not require extra training labels. In addition, LGGM introduces a small number of parameters and computational operations to extract object features with improved object information. Ultimately, FCLGYOLO outperforms state-of-the-art detection algorithms in terms of detection performance on forest fire datasets, while also exhibiting the smallest number of parameters and computational complexity.

REFERENCES

[1] F. Wen, J. Shi, G. Gui, H. Gacanin, and O. A. Dobre, "3-D positioning method for anonymous UAV based on bistatic polarized MIMO radar," *IEEE Internet Things J.*, vol. 10, no. 1, pp. 815–827, Jan. 2023.
[2] F. Wen, G. Gui, H. Gacanin, and H. Sari, "Compressive sampling framework for 2D-DOA and polarization estimation in mmwave polarized massive MIMO systems," *IEEE Trans. Wirel. Commun.*, vol. 22, no. 5, pp. 3071–3083, May 2023.
[3] L. Zhang, L. Song, B. Du, and Y. Zhang, "Nonlocal low-rank tensor completion for visual data," *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 673–685, Feb. 2021.
[4] Z. He, X. Zhang, Y. Wang, Y. Lin, G. Gui, and H. Gacanin, "A robust CSI-based Wi-Fi passive sensing method using attention mechanism deep learning," *IEEE Internet Things J.*, vol. 10, no. 19, pp. 17490–17499, Oct. 2023.
[5] L. Zhang and L. Zhang, "Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 270–294, Jun. 2022.
[6] Z. Zhang, J. Shi, and F. Wen, "Phase compensation-based 2D-DOA estimation for EMVs-MIMO radar," *IEEE Trans. Aerosp. Electron. Syst.*, to be published, doi: 10.1109/TAES.2023.3335194.

[7] X. Wang, Y. Guo, F. Wen, J. He, and T.-K. Truong, "EMVs-MIMO radar with sparse Rx geometry: Tensor modeling and 2-D direction finding," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 6, pp. 8062–8075, Dec. 2023.

[8] J. Wan et al., "Precise facial landmark detection by reference heatmap transformer," *IEEE Trans. Image Process.*, vol. 32, no. 144, pp. 1966–1977, Mar. 2023.

[9] S. Wang et al., "Forest fire detection based on lightweight YOLO," in *Proc. Chin. Control Decis. Conf.*, 2021, pp. 1560–1565.

[10] T. Hussain, H. Dai, W. Gueaieb, M. Sicklinger, and G. De Masi, "UAV-based multi-scale features fusion attention for fire detection in smart city ecosystems," in *Proc. IEEE Int. Smart Cities Conf.*, 2022, pp. 1–4.

[11] P. Nagababu, K. Dhakshitha, G. Chandrika, and U. R. Chowdary, "Automated fire detection system using image surveillance system (iss) and convolutional neural networks (CNN)," in *Proc. 9th Int. Conf. Adv. Comput. Commun. Syst.*, 2023, pp. 1366–1369.

[12] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 779–788.

[13] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[14] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[15] G. Jocher, "ultralytics/yolov5: V6.1," 2022. Accessed: Dec. 28, 2023. [Online]. Available: https://github.com/ultralytics/yolov5

[16] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 936–944.

[17] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.

[18] J. Zhang, J. Lei, W. Xie, Z. Fang, Y. Li, and Q. Du, "SuperYOLO: Super resolution assisted object detection in multimodal remote sensing imagery," *IEEE Trans. Geosci. Remote. Sens.*, vol. 61, no. 1482, Mar. 2023, Art. no. 5605415.

[19] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[20] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2020, pp. 9726–9735.

[21] A. Bardes, J. Ponce, and Y. LeCun, "VICReg: Variance-invariance-covariance regularization for self-supervised learning," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–23.

[22] H. Sun et al., "Multi-level feature interaction and efficient non-local information enhanced channel attention for image dehazing," *Neural Netw.*, vol. 163, pp. 10–27, 2023.

[23] Z. Wang, Z. Chen, and B. Du, "Active learning with co-auxiliary learning and multi-level diversity for image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 3899–3911, Aug. 2023.

[24] H. Sun et al., "Partial siamese with multiscale bi-codec networks for remote sensing image haze removal," *IEEE Trans. Geosci. Remote. Sens.*, vol. 61, no. 153, Oct. 2023, Art. no. 4106516.

[25] J. Wan et al., "Robust and precise facial landmark detection by self-calibrated pose attention network," *IEEE Trans. Cybern.*, vol. 53, no. 6, pp. 3546–3560, Jun. 2023.

[26] S. Chan, Y. Wang, Y. Lei, X. Cheng, Z. Chen, and W. Wu, "Asymmetric cascade fusion network for building extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, no. 62, Aug. 2023, Art. no. 2004218.

[27] H. Sun et al., "Scale-free heterogeneous cycleGAN for defogging from a single image for autonomous driving in fog," *Neural Comput. Appl.*, vol. 35, no. 5, pp. 3737–3751, 2023.

[28] Y. Rao, W. Zhao, Z. Zhu, J. Lu, and J. Zhou, "Global filter networks for image classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 980–993.

[29] C. Wang, J. Jiang, Z. Zhong, and X. Liu, "Spatial-frequency mutual learning for face super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2023, pp. 22356–22366.

[30] S. D. Thepade, J. H. Dewan, D. Pritam, and R. Chaturvedi, "Fire detection system using color and flickering behaviour of fire with Kekre's luv color space," in *Proc. Int. Conf. Comput. Commun. Control Autom.*, 2018, pp. 1–6.

[31] N. Chowdhury, D. R. Mushfiq, and A. E. Chowdhury, "Computer vision and smoke sensor based fire detection system," in *Proc. Int. Conf. Adv. Sci. Eng. Robot. Technol.*, 2019, pp. 1–5.

[32] K. Chen, Y. Cheng, H. Bai, C. Mou, and Y. Zhang, "Research on image fire detection based on support vector machine," in *Proc. Int. Conf. Fire Sci. Fire Protection Eng.*, 2019, pp. 1–7.

[33] Y. Yang, Z. Li, and J. Zhang, "Fire detection of satellite remote sensing images based on VGG ensemble classifier," in *Proc. IEEE Conf. Telecommun.Optics Comput. Sci.*, 2021, pp. 31–36.

[34] W. S. Mseddi, R. Ghali, M. Jmal, and R. Attia, "Fire detection and segmentation using YOLOv5 and U-Net," in *Proc. Eur. Signal Process. Conf.*, 2021, pp. 741–745.

[35] R. Ghali, M. A. Akhloufi, M. Jmal, W. S. Mseddi, and R. Attia, "Forest fires segmentation using deep convolutional neural networks," in *Proc. IEEE Int. Conf. Systems, Man, Cybern.*, 2021, pp. 2109–2114.

[36] A. Vaswani et al., "Attention is all you need," in *Proc*. Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.

[37] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 580–587.

[38] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.

[39] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–16.

[40] S. Liu et al., "DAB-DETR: Dynamic anchor boxes are better queries for DETR," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–20.

[41] F. Li, H. Zhang, S. Liu, J. Guo, L. M. Ni, and L. Zhang, "DN-DETR: Accelerate DETR training by introducing query denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2022, pp. 13619–13627.

[42] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[43] C. Liu et al., "Overcoming data limitations: A few-shot specific emitter identification method using self-supervised learning and adversarial augmentation," *IEEE Trans. Inf. Forensics Secur.*, vol. 19, no. 36, pp. 500–513, Oct. 2024.

[44] Y. Peng et al., "Supervised contrastive learning for RFF identification with limited samples," *IEEE Internet Things J.*, vol. 10, no. 19, pp. 17293–17306, Oct. 2023.

[45] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol.*, 2019, pp. 4171–4186.

[46] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy, "Tracking emerges by colorizing videos," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 402–419.

[47] M. Noroozi, A. Vinjimoor, P. Favaro, and H. Pirsiavash, "Boosting self-supervised learning via knowledge transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 9359–9367.

[48] C. Ledig et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 105–114.

[49] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 3733–3742.

[50] J. Grill et al., "Bootstrap your own latent—A new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 21271–21284.

[51] X. Chen and K. He, "Exploring simple Siamese representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 15750–15758.

[52] D. Yin, R. G. Lopes, J. Shlens, E. D. Cubuk, and J. Gilmer, "A Fourier perspective on model robustness in computer vision," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 13255–13265.

[53] Y. Yang and S. Soatto, "FDA: Fourier domain adaptation for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2020, pp. 4084–4094.

[54] Q. Xu, R. Zhang, Y. Zhang, Y. Wang, and Q. Tian, "A Fourier-based framework for domain generalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 14383–14392.

[55] D. Fuoli, L. V. Gool, and R. Timofte, "Fourier space losses for efficient perceptual image super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 2340–2349.

[56] L. Jiang, B. Dai, W. Wu, and C. C. Loy, "Focal frequency loss for image reconstruction and synthesis," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 13899–13909.
[57] H. Yu, N. Zheng, M. Zhou, J. Huang, Z. Xiao, and F. Zhao, "Frequency and spatial dual guidance for image dehazing," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 181–198.
[58] J. Huang et al., "Deep Fourier-based exposure correction network with spatial-frequency interaction," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 163–180.
[59] J. Lee-Thorp, J. Ainslie, I. Eckstein, and S. Ontañón, "FNet: Mixing tokens with Fourier transforms," in *Proc. 2022 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2022, pp. 4296–4313.
[60] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Z. Fulé, and E. Blasch, "Aerial imagery pile burn detection using deep learning: The flame dataset," *Comput. Netw.*, vol. 193, 2021, Art. no. 108001.
[61] B. Hopkins et al., "Flame 2: Fire detection and modeling: Aerial multi-spectral image dataset," 2022. Accessed: Dec. 28, 2023. [Online]. Available: https://dx.doi.org/10.21227/swyw-6j78
[62] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2020, pp. 9756–9765.
[63] B. Zhu et al., "AutoAssign: Differentiable label assignment for dense object detection," 2020. Accessed: Dec. 28, 2023. [Online]. Available: https://arxiv.org/abs/2007.03496
[64] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9626–9635.
[65] X. Zhang, F. Wan, C. Liu, R. Ji, and Q. Ye, "FreeAnchor: Learning to match anchors for visual object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 147–155.
[66] H. Zhang et al., "DINO: DETR with improved denoising anchor boxes for end-to-end object detection," in *Proc. Int. Conf. Learn. Representations*, 2023, pp. 1–19.
[67] H. Zhang, Y. Wang, F. Dayoub, and N. Sünderhauf, "VarifocalNet: An iou-aware dense object detector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 8514–8523.
[68] A. Bochkovskiy, C. Wang, and H. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arxiv:2004.10934*. Accessed: Dec. 28, 2023.
[69] C. Wang, A. Bochkovskiy, and H. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2023, pp. 7464–7475.
[70] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," 2021, *arxiv:2107.08430*. Accessed: Dec. 28, 2023.

Dong Ren received the B.E. degree in mechanical and electronic engineering and the Ph.D. degree in engineering from Jilin University, Jilin, China, in 1999 and 2007, respectively.

He is currently a Professor with the College of Computer and Information Technology, China Three Gorges University, Yichang, China. His research interests include pattern recognition, 3S technology, and Internet-of-Things technology.

Yang Zhang received the B.E. degree in geophysics from Wuhan University, Wuhan, China, in 2020.

His research interests include remote sensing image recognition and computer vision.

Lu Wang received the master's degree in electronic science and technology from the Harbin Institute of Technology, Harbin, China, in 2011. She is currently working toward the doctoral degree in electronic engineering with the College of Electrical Engineering and New Energy, China Three Gorges University, Yichang, China.

Her research interests include artificial intelligence and the application of image processing in electric engineering.

Hang Sun received the Ph.D. degree in school of computer science, Wuhan University, Wuhan, China, in 2017.

He previously worked as a Senior Engineer with Huawei Technology Company, Ltd., Shenzhen, China, responsible for the research and application of computer vision. He is currently an Associate Professor with the College of Computer and Information Technology, China Three Gorges University, Yichang, China. His works have been published in premier computer vision journals and conferences, including IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, *Neural Networks*, *Science China Information Sciences, Neural Computing and Applications*, *Chinese Journal of Electronics*, Pacific-Rim Conference on Multimedia, International joint Conference on Neural Networks, and so on. His major research interests include image dehazing, underwater image restoration, and object detection.

Shun Ren received the Ph.D. degree in agricultural biological environment and energy engineering from Jilin University, Changchun, China, in 2016.

He is currently working with the College of Computer and Information Technology, China Three Gorges University, Yichang, China. His research interests include artificial intelligence, Internet of Things, and wireless sensor network.

Jian Gu received the B.E degree in animal and plant quarantine from Huazhong Agricultural University, Wuhan, China, in 2008.

He is/was engaged in forestry pest management, disaster prevention and reduction, and quarantine services. Specifically he is responsible for: forestry pest control and new technology introduction, experimentation, and promotion; forestry pest monitoring, investigation, and quarantine; compilation, organization, and implementation of forestry pest-related projects; publicity, education, and training on forestry pest policies and technologies. His research focuses on the application of unmanned aerial vehicle remote sensing technology.