drones

MDPI

Article

FBC-ANet: A Semantic Segmentation Model for UAV Forest Fire Images Combining Boundary Enhancement and Context Awareness

Lin Zhang [1], Mingyang Wang [1,*], Yunhong Ding [1], Tingting Wan [2], Bo Qi [3] and Yutian Pang [4]

[1] College of Information and Computer Engineering, Northeast Forestry University, Harbin 150040, China; lawrence@nefu.edu.cn (L.Z.)
[2] Department of Information Engineering, Heilongjiang Institute of Construction Technology, Harbin 150025, China
[3] School of Astronautics, Harbin Institute of Technology, Harbin 150010, China
[4] School for Engineering of Matter, Transport & Energy, Arizona State University, Tempe, AZ 85287, USA
* Correspondence: wangmingyang@nefu.edu.cn

Abstract: Forest fires are one of the most serious natural disasters that threaten forest resources. The early and accurate identification of forest fires is crucial for reducing losses. Compared with satellites and sensors, unmanned aerial vehicles (UAVs) are widely used in forest fire monitoring tasks due to their flexibility and wide coverage. The key to fire monitoring is to accurately segment the area where the fire is located in the image. However, for early forest fire monitoring, fires captured remotely by UAVs have the characteristics of a small area, irregular contour, and susceptibility to forest cover, making the accurate segmentation of fire areas from images a challenge. This article proposes an FBC-ANet network architecture that integrates boundary enhancement modules and context-aware modules into a lightweight encoder–decoder network. FBC-Anet can extract deep semantic features from images and enhance shallow edge features, thereby achieving an effective segmentation of forest fire areas in the image. The FBC-ANet model uses an Xception network as the backbone of an encoder to extract features of different scales from images. By transforming the extracted deep semantic features through the CIA module, the model's feature learning ability for fire pixels is enhanced, making feature extraction more robust. FBC-ANet integrates the decoder into the BEM module to enhance the extraction of shallow edge features in images. The experimental results indicate that the FBC-ANet model has a better segmentation performance for small target forest fires compared to the baseline model. The segmentation accuracy on the dataset FLAME is 92.19%, the F1 score is 90.76%, and the IoU reaches 83.08%. This indicates that the FBC-ANet model can indeed extract more valuable features related to fire in the image, thereby better segmenting the fire area from the image.

Keywords: forest fires; semantic segmentation; boundary enhancement; contextual information awareness; encoder–decoder

check for updates

Citation: Zhang, L.; Wang, M.; Ding, Y.; Wan, T.; Qi, B.; Pang, Y. FBC-ANet: A Semantic Segmentation Model for UAV Forest Fire Images Combining Boundary Enhancement and Context Awareness. *Drones* **2022**, *7*, 456. https://doi.org/10.3390/drones7070456

Academic Editor: Diego González-Aguilera

Received: 16 May 2023
Revised: 29 June 2023
Accepted: 30 June 2023
Published: 9 July 2023

1. Introduction

Forest fires are one of the most destructive and widespread natural disasters in the world, causing huge ecological and economic losses to forest resources and human society [1,2]. Especially in recent years, with the impact of global warming, forest fires have been constantly occurring. For instance, the data released by the Chilean Disaster Prevention and Response Center show that, in March 2023, the forest fires in multiple areas in the central and southern of Chile severely affected 438 thousand hectares of land, destroying more than 2500 houses and affecting 7770 people in total. With the impact of global climate warming, the frequency of forest fires has been significantly increasing. Thus, recognition and detection for early warning are of great significance. The early detection of forest fires is an effective way to reduce losses.

Traditional fire monitoring methods use lookout towers equipped with surveillance cameras and other sensors to monitor and locate fires or a surge in temperature in the forest through visual imaging, or use high-altitude helicopters satellites to evaluate large fires on a broader scale [3,4]. Remote sensing satellites have the advantages of a strong continuity and wide coverage, and can continuously track and detect large-scale forest fires. Especially in remote and sparsely populated areas, remote sensing satellites can provide important data sources for identifying forest fires. However, satellite images may be affected by atmospheric conditions, such as cloud or fog interference. Meanwhile, the resolution of satellite images is limited, making it difficult to identify small or hidden fire spots. In the early stages of a fire, the range of fire points is usually small. If the fire points can be identified and put out in a timely manner, it will effectively prevent the spread of the fire and reduce the losses caused by forest fires.

In recent years, the application of UAVs equipped with various sensors and cameras to monitor forest fires using their low-altitude flight characteristics is increasingly favored by forestry personnel and firefighters. UAVs provide new solutions for fire monitoring due to their powerful flexibility, high maneuverability, and adjustable field of view [5–9]. Note that UAV images exhibit different attributes, such as high-resolution microscopic images from low altitudes, which are significantly different from macroscopic images captured by high-altitude helicopters or satellites. The UAV has the advantage of a flexible aerial patrol as the remote distance of UAVs is approximately 4 to 7 km. This allows firefighters and forestry managers to quickly monitor the forest fire without blind space by operating UAVs without delving deep into the forest, thereby reducing the risks that firefighters and forestry managers face. With the rapid development of machine vision and deep learning, real-time classification and detection based on images are widely applied in this field of forest fires [10–15]. Modern UAVs can be equipped with small CPUs and GPUs, as well as pre-trained deep network models onboard [16,17], in order to detect fires as early as possible. Some supervised learning methods such as [18,19] rely on the use of public fire dataset CorsicanFire [20]. However, this dataset is based on ground fire images and loses its significance in helping firefighters detect the occurrence of fires. In the current research, we utilized the dataset FLAME [21], the abbreviation for Fire Luminosity Airborne Machine Learning Evaluation, to train and validate the proposed semantic segmentation model. To our knowledge, the FLAME dataset is the only fire analysis dataset captured and imaged by multiple UAVs. This dataset is a set of fire video and image data collected by unmanned aerial vehicles during the combustion of prescribed combustion deposits in Arizona pine forests, containing videos captured by thermal infrared cameras from different angles, scales, and camera types. In addition, FLAME is equipped with a deep network model for fire detection and segmentation, which can serve as a benchmark model for fire semantic segmentation.

Semantic detection and segmentation on image or video frames is a classic problem and also a hot topic in the field of computer vision research, the purpose of which is to separate the diverse objects from an ambient background in the image; in addition, it is widely used in image understanding, video surveillance, medical image analysis, and other applications. Semantic detection is to identify all interested objects in an image and determine their categories and positions, while semantic segmentation, as a downstream task of semantic detection, is a pixel-level classification, meaning that the categories have semantics in real world, such as cars, trees, and a crowd of people. The semantic detection and segmentation of forest fires is a challenging and difficult task. The shape, color, brightness, and other features of flames have strong uncertainty and variability, making it difficult to accurately segment the flame region from images. In addition, there are a large number of interference factors such as smoke and dust in fire scenes, which makes it difficult to extract flame features in real time and to segment flames. Researchers has proposed some deep-learning-based models for the semantic detection and segmentation of forest fires. Recently, Norkobil Saydirasulovich et al. [22] and Avazov et al. [23] proposed a series of improved fast fire detection learning methods based on the family of YOLO framework deep networks to accurately detect fires from fire-like surroundings in complex scenes [22,23]. The MaskSU R-

CN [24] was proposed, which combines mask scoring R-CNN [25] and a U-shaped network to detect and locate wildfires. U-shaped models based on Transformer [18,26] are also proposed to improve the segmentation accuracy of flame edges in complex backgrounds. Inspired by various encoder–decoder architectures, this study proposes a new semantic segmentation model FBC-ANet based on the encoder–decoder architecture that combines a boundary enhancement module and a context awareness module to identify and segment forest fires from UAVs. The main contributions of this paper are as follows:

(1) Xception [27] is used as the backbone network of the encoder. In the decoder section, a boundary enhancement module is proposed to generate enhanced features to restore boundary information in order to improve the effectiveness of semantic segmentation for forest fires.

(2) With regard to the bottleneck part, the proposed contextual information awareness module is utilized to perform segmentation,which enhances the feature learning ability of the fire pixels and makes feature extraction more robust.

(3) In the experimental environment, we verified that the FBC-ANet model obtained a prediction accuracy of 0.9219, an IoU of 0.8308, and an F1 score of 0.9076 on the FLAME dataset.

2. Related Works

There are many types of objects detection and semantic segmentation models, which can be divided into different categories based on different network structures. In this study, we are chiefly concerned with the models using fully convolutional networks with an encoder–decoder architecture, such as FCN [28], U-Net [29], PSPNet [30], and RefineNet [31], as well as DeepLabV3+ [32], which are widely used in the tasks of objects detection and semantic segmentation. Among them, FCN [28] opened a new age of image segmentation, where it was the first instance of end-to-end deep full convolutional networks being applied to the community of semantic segmentation by using deconvolution layers to restore image resolution and achieve end-to-end pixel-level classification. U-Net [29] is a method based on the improvement of FCN and is proposed for medical image segmentation, adopting up-sampling and jump connection to achieve high-precision medical segmentation and solving the problem of partial pixel spatial information loss through the encoder–decoder structure. PSPNet [30] is a semantic segmentation model based on a pyramid pooling network that uses pooling layers of different scales to extract and fuse the global context and local information, improving the semantic segmentation performance. RefineNet [31] is a multi-path refining network that utilizes multi-level abstract features for high-resolution semantic segmentation and recursively extracts low-resolution features to generate high-resolution features. Unlike U-Net, which directly cascades the feature map of the encoder after upsampling, RefineNet uses the features generated by the encoder and the output of the previous stage of the encoder as inputs simultaneously and performs a series of convolutions, resulting in the fusion of multiscale features. DeepLabV3+ [32] is an improved version of DeepLabV3. It uses multi-scale dilated convolution and atrous spatial pyramid pooling (ASPP) modules to capture multi-scale context information while using deep separable convolution to reduce the number of parameters and computations. DeepLabV3+ also uses a decoder module to further improve the segmentation performance, manifesting a remarkable advancement in several benchmarks [33–36]. FastFCN [37] is a fast semantic segmentation network based on dilated convolution and separable convolution, using the joint pyramid upsampling module to accelerate the feature fusion process.

3. Materials and Methods

In this section, the forest fire dataset used for training, validation, and testing in this work was first introduced. Then, the structure of the FBC-ANet model for the semantic segmentation of forest fires was presented.

3.1. Datasets

The publicly available dataset FLAME (Fire Luminosity Airborne-based Machine learning Evaluation) (the dataset is available at https://ieee-dataport.org/open-access/flame-dataset-aerial-imagery-pile-burn-detection-using-drones-uavs) was used to validate the effectiveness of the FBC-ANet model in semantic segmentation of forest fires in UAVs. FLAME is a fire video dataset collected by different types of UAVs and cameras during the burning of sediment in a pine forest in Arizona. The videos in the FLAME dataset were captured by Phantom 3 PRO and Matrice 200 V1 equipped with Vue Pro R and Phantom 3 cameras. The Vue Pro R cameras were used to capture the thermal infrared (TIR) data (i.e., videos in whitehot, greenhot, and fusion palettes) with a resolution of 640 × 512 pixels under a frame rate of 30 FPS, while the Phantom 3 cameras were used to capture the normal visible spectrum data (i.e., videos in RGB palettes). Figure 1 shows a couple of frames as examples of the two types of data. Analyzing thermal infrared images can also be used as a means of detecting flames, as thermal infrared images are more sensitive to the temperature characteristics of flames. With the help of TIR data, it will greatly facilitate the design of forest fire segmentation models [38–40]. Unfortunately, the FLAME dataset does not provide annotations for the TIR data. However, the fire videos in RGB palette were converted into a collection of frames with a resolution of 3840 × 2160 pixels for the purpose of semantic segmentation. Tables 1 and 2 report more information about the diverse types of data as well as the UAVs used in the dataset FLAME. For simplicity, we refer to the collection of frames for semantic segmentation as "FLAME-Seg". FLAME-Seg consists of 2003 images that have been annotated with masks as the groundtruth for each image. Due to the videos in TIR palette not exactly corresponding to those in RGB palette, we can only use the annotated RGB flames; that is, the sub-dataset FLAME-Seg. In the current work, 85% of the images in the sub-dataset FLAME-Seg were used for training and 15% for testing.

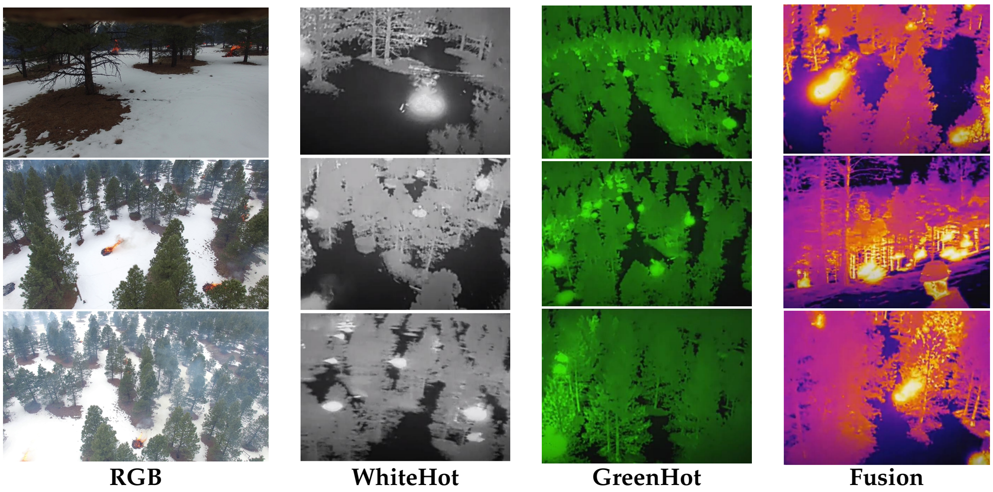


Figure 1. Examples of frames in normal visible spectrum and 3 kinds of thermal infrared (TIR) palettes.

Table 1. Information about the various kinds of data in dataset FLAME [21].

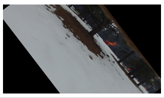
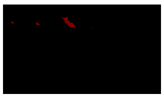
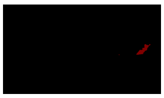
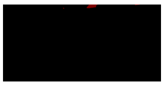
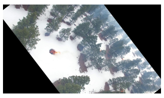
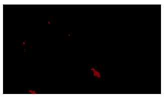
Type	Format	Palette	Duration	FPS	Resolution	Label	Shot by
Video	MOV	WhiteHot	89 s	30	640 × 512	-	Vue Pro R, FLIR
Video	MOV	GreenHot	305 s	30	640 × 512	-	Vue Pro R, FLIR
Video	MOV	Fusion	25 min	30	640 × 512	-	Vue Pro R, FLIR
Video	MOV	RGB	17 min	30	3840 × 1920	-	Phantom, DJI
Image	JPEG	RGB	2003 frames	-	3840 × 1920	Fire	Phantom, DJI
Mask	PNG	Binary	2003 frames	-	3840 × 1920	Fire	-

Table 2. The technical parameters of the UAVs used [21,41,42].

	Type	Horizontal Speed	Remote Distance	Wheelbase	Weight
	PHANTOM 3	<61.2 km/h	3500 m to 5000 m	350 mm	1.28 kg
	MATRICE 200	<57.6 km/h	4000 m to 7000 m	643 mm	3.80 kg

As is well known, the training of deep convolutional networks largely relies on sufficient datasets. In the case of limited annotated images, it is necessary to use data augmentation techniques to increase the amount of data. Data enhancement is mainly achieved by performing photometric or geometric transformations on the image, as well as mosaic augmentation (synthesis). The FLAME dataset is enhanced by flipping, rotating, translating, and clipping images. It is worth noting that flipping the fire image horizontally is reasonable. However, vertical flipping is not allowed because there is no inverted flame in the real world. Similarly, the rotation of the image is not allowed to exceed 90 degrees. Based on the same requirements, masking operations were performed to generate augmented masks. Table 3 provides a schematic diagram of the data enhancement process for two images as examples.

Table 3. The examples of data augmentation.

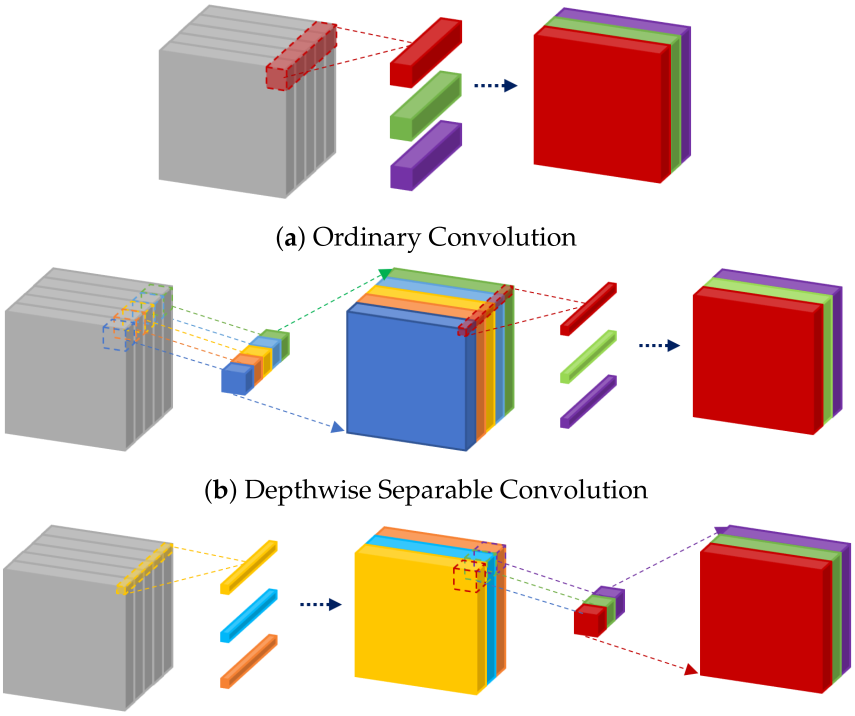
	Raw	Flipping	Rotating	Translating	Clipping
Image					
Mask					
Image					
Mask					

3.2. Feature Extraction Module (FEM)

In the FBC-ANet model that we proposed, Xception [27] is used as the backbone network of the encoder. This section first briefly reviews the basic ideas of Xception, and then discusses the details of the feature extraction module based on Xception.

Xception is a lightweight neural network based on InceptionV3 [43] and is considered a high-end version of the Inception series [43–45]. Xception uses depthwise separable convolution to extract features from spatial convolution and channel convolution, respectively. Research has confirmed that the feature extraction performance of Xception is superior to Inception V3 on ImageNet [46]. Therefore, Xception was used to extract the characteristics of fires in UAVs in the current work.

Depthwise separable convolution decomposes ordinary convolution operations into two processing processes: depthwise operation and pointwise by point operation. Depthwise operations perform spatial convolution on each input channel separately to obtain the same number of output channels, so as to reduce the computational workload and parameters in the spatial dimension. The pointwise operation performs 1 × 1 convolution on the output of the depthwise operation to increase the number of output channels and enhance the expression ability in the channel dimension. For instance, with regard to an input $H \times W \times C_{in}$, the output is $H \times W \times C_{out}$, and the kernel size of convolution layer is $k \times k$. Ordinary convolution requires $k \times k \times C_{in} \times C_{out}$ parameters and $H \times W \times k \times k \times C_{in} \times C_{out}$ instances of multiplication operation; however, depthwise separable convolution only requires $k \times k \times C_{in} + C_{in} \times C_{out}$ parameters and $H \times W \times (k \times k + C_{out}) \times C_{in}$ instances of multiplication operation. Thus, compared with ordinary convolution, depth-separable convolution can greatly reduce the parameters and calculation amount of operation, which means that depth-separable convolution can effectively reduce the redundancy and overfitting risk in the training process. Xception also adopts the idea of deep separable convolution. However, in Xception, the order of depthwise convolution and pointwise convolution is swapped as shown in Figure 2. We compared these two depthwise separable convolution schemes in the experiment.



(**a**) Ordinary Convolution

(**b**) Depthwise Separable Convolution

(**c**) Depthwise Separable Convolution in Xception

Figure 2. Schematic diagrams of three convolution strategies, among which (**a**) provides the process of ordinary convolution; (**b**) provides the process of depthwise separable convolution; (**c**) provides the process of depthwise separable convolution used in Xception. It can be seen that ordinary convolution involves convolution of all input channels in both spatial and channel dimensions to obtain the output of each channel. Depthwise separable convolution first performs spatial convolution on each input channel, and then performs 1 × 1 convolution on all output channels. In Xception, the order of depthwise convolution and pointwise convolution is swapped.

In an image semantic segmentation model with an encoder–decoder framework, the role of the decoder is opposite that of the encoder. The encoder downsamples the image through convolutional layers, thereby reducing the size of the feature map and increasing the number of channels. The decoder uses deconvolution or upsampling operations to restore the feature map extracted by the encoder to the size of the original image, while classifying each pixel. Table 4 lists the details of each layer in the proposed encoder–decoder framework.

Table 4. The detailed layers of feature extraction module in our proposed encoder–decoder framework.

Block	Operation	Kernel Size	Stride/Padding	Output Size
①	`Conv + ReLU`	3 × 3 × 32	2 × 2/1	1920 × 1080 × 32
	`Conv + ReLU`	3 × 3 × 64	1 × 1/2	1920 × 1080 × 64
②	`Residual`	1 × 1 × 128	2 × 2/0	960 × 540 × 128
	`SeparableConv`	3 × 3 × 128	1 × 1/2	1920 × 1080 × 128
	`ReLU + SeparableConv`	3 × 3 × 128	1 × 1/2	1920 × 1080 × 128
	`MaxPooling`	3 × 3 × 128	2 × 2/1	960 × 540 × 128
③	`Residual`	1 × 1 × 256	2 × 2/0	480 × 270 × 256
	`ReLU + SeparableConv`	3 × 3 × 256	1 × 1/2	960 × 540 × 256
	`ReLU + SeparableConv`	3 × 3 × 128	1 × 1/2	960 × 540 × 256
	`MaxPooling`	3 × 3 × 256	2 × 2/1	480 × 270 × 256
④	`Residual`	1 × 1 × 728	2 × 2/0	240 × 135 × 728
	`ReLU + SeparableConv`	3 × 3 × 728	1 × 1/2	480 × 270 × 728
	`ReLU + SeparableConv`	3 × 3 × 728	1 × 1/2	480 × 270 × 728
	`MaxPooling`	3 × 3 × 728	2 × 2/1	240 × 135 × 728
⑤–⑫	`ReLU + SeparableConv`	3 × 3 × 728	1 × 1/2	240 × 135 × 728
	`ReLU + SeparableConv`	3 × 3 × 728	1 × 1/2	240 × 135 × 728
	`ReLU + SeparableConv`	3 × 3 × 728	1 × 1/2	240 × 135 × 728
⑬	`Residual`	1 × 1 × 1024	2 × 2/0	120 × 67 × 1024
	`ReLU + SeparableConv`	3 × 3 × 728	1 × 1/2	240 × 135 × 728
	`ReLU + SeparableConv`	3 × 3 × 1024	1 × 1/2	240 × 135 × 1024
	`MaxPooling`	3 × 3 × 1024	2 × 2/1	120 × 67 × 1024
⑭	`SeparableConv + ReLU`	3 × 3 × 1536	2 × 2/1	120 × 67 × 1536
	`SeparableConv + ReLU`	3 × 3 × 2048	1 × 1/2	120 × 67 × 2048
⑮	`UpSampling`	–	–	240 × 135 × 2048
	`SeparableConv + ReLU`	3 × 3 × 1024	2 × 2/1	240 × 135 × 1024
	`Residual`	1 × 1 × 1024	1 × 1/0	240 × 135 × 1024
	`SeparableConv + ReLU`	3 × 3 × 728	1 × 1/2	240 × 135 × 728
	`SeparableConv + ReLU`	3 × 3 × 728	1 × 1/2	240 × 135 × 728
⑯	`UpSampling`	3 × 3 × 728	2 × 2/1	480 × 270 × 728
	`SeparableConv + ReLU`	3 × 3 × 728	2 × 2/1	480 × 270 × 728
	`Residual`	1 × 1 × 728	1 × 1/0	480 × 270 × 728
	`SeparableConv + ReLU`	3 × 3 × 728	1 × 1/2	480 × 270 × 728
	`SeparableConv + ReLU`	3 × 3 × 728	1 × 1/2	480 × 270 × 728
⑰	`UpSampling`	–	–	960 × 540 × 728
	`SeparableConv + ReLU`	3 × 3 × 256	2 × 2/1	960 × 540 × 256
	`Residual`	1 × 1 × 256	1 × 1/0	960 × 540 × 256
	`SeparableConv + ReLU`	3 × 3 × 256	1 × 1/2	960 × 540 × 256
	`SeparableConv + ReLU`	3 × 3 × 256	1 × 1/2	960 × 540 × 256
⑱	`UpSampling`	–	–	1920 × 1080 × 256
	`SeparableConv + ReLU`	3 × 3 × 128	2 × 2/1	1920 × 1080 × 128
	`Residual`	1 × 1 × 128	1 × 1/0	1920 × 1080 × 128
	`SeparableConv + ReLU`	3 × 3 × 128	1 × 1/2	1920 × 1080 × 128
	`SeparableConv + ReLU`	3 × 3 × 128	1 × 1/2	1920 × 1080 × 128
⑲	`Conv + ReLU`	3 × 3 × 64	1 × 1/2	1920 × 1080 × 64
	`UpSampling`	–	–	3840 × 2160 × 64
	`Conv + ReLU`	3 × 3 × 32	1 × 1/2	3840 × 2160 × 32
	`Conv + Sigmoid`	3 × 3 × 2	1 × 1/2	3840 × 2160 × 2

In the FBC-ANet model, the decoder network consists of 22 convolutional layers, which are divided into 4 blocks. Each block starts from the upper sampling layer to double the size of the feature map. There are linear residual connections between the encoder and the decoder. It is worth noting that the residuals transformed by a 1 × 1 convolution kernel have a stride of 1 × 1 rather than 2 × 2 like the encoder because the feature size has been zoomed in twice. As in the encoder, the same depthwise separable convolution layer was adopted in decoder to replace ordinary convolution. However, unlike the encoder, the depthwise separable convolution operation was performed followed by ReLU to form a structure symmetrical to the encoder. Lastly, the decoder network outputs a 2-channel feature map of the original image size activated by sigmoid function to determine whether each pixel belongs to a forest flame or backgrounds.

3.3. Boundary Enhancement Module (BEM)

Boundary information is crucial for improving the performance of semantic segmentation. The shallow network can better extract the boundary information of the target in the image, while the deep network can help to obtain semantic information. Deep semantic information can optimize shallow boundary and contour information. Therefore, effective fusion of shallow and deep information can avoid the loss of image information. In FBC-ANet model, a boundary enhancement module (BEM) was designed to enhance shallow features with deep features to obtain more boundary information, guide or constrain segmentation results, and better locate target boundaries to improve the accuracy of semantic segmentation results.

Figure 3 shows a schematic diagram of the boundary enhancement module (BEM). First, 1 × 1 convolution was performed to connect the shallow layer of the encoder and the upsampling layer of the decoder to achieve a linear combination of the information, so as to obtain the fused features (labelled as $Input$). Then, residual structure was used to obtain boundary-enhanced features.The residual network can strengthen the recognition ability of each stage, inspired from the architecture of ResNet [47], which added batchnorm to prevent overfitting and accelerate the convergence rate and added Relu to avoid the disappearance of gradient, can learn the difference between output and input features, and can amplify the gradient flow through skip layer connections, which can alleviate the degradation problem of the neural network and make deep training possible. The enhanced feature is output as the following formula:

$$Output = \mathrm{ReLU}(Input + \mathrm{Conv}_{3\times3}(\mathrm{BN}(\mathrm{ReLU}(\mathrm{Conv}_{3\times3}(Input))))). \tag{1}$$

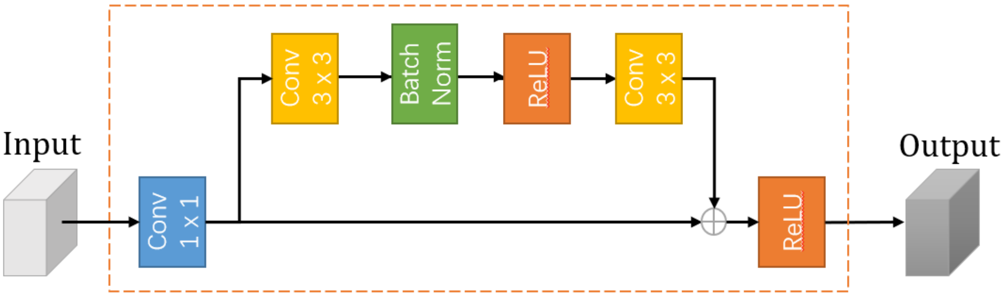


Figure 3. The diagram of the boundary enhancement module (BEM).

3.4. Contextual Information Awareness (CIA) Module

A contextual information awareness (CIA) module was designed in FBC-ANet model to capture features of different scales in the image feature map so as to enhance the feature extraction ability and help to identify fire source targets of different sizes. The input features of the CIA module are from the encoder of the feature extraction module, represented as $\mathbf{I} \in \mathbb{R}^{H\times W\times C}$. Firstly, an adaptive average pooling operation was used to convert input features into features of a specific scale size. The adaptive average pooling can generate features into tensors of a specified size. Inspired by the spatial pyramid pooling in PSPNet [30], a pooling layer was used to halve and quarter the resolution of feature maps. The original feature size was reduced by 1/2 as well as by 1/4, and the different scales of feature maps are indicated as $\mathbf{I}_1, \mathbf{I}_2$ and $\mathbf{I}_3$, respectively. A 3 × 3 convolution operation was performed followed by a batch-normalization layer and an ReLU layer on each scale of feature maps. Next, a convolution layer of 1 × 1 learns the linear combination of input channels by projecting tensors into a higher-dimensional space, thereby generating the feature $\mathbf{U} \in \mathbb{R}^{H \times W \times C'}$, where $C' > C$. After owning the effective receptive field of $H \times W$, the network also needs to model the long-distance non-local dependence. In order to enable the network to learn the global representation, the characteristic $\mathbf{U} \in \mathbb{R}^{H \times W \times C'}$ was divided into N blocks with the size of $h \times w \times C'$ pixels, and all blocks were flattened through reconstruction to obtain the feature $\mathbf{V} \in \mathbb{R}^{M \times N \times C'}$, where $M = wh, N = HW/M$, and h, w are the height and width of each block, respectively. Non-local self-attention mechanism encodes the relationship among the pixels of each block and generates the feature $\mathbf{N} \in \mathbb{R}^{M \times N \times C'}$. Non-local modules are a self-attention mechanism that can capture the global dependencies of input features [48]. As shown in the following formula, the function of non-local modules is to calculate the similarity between each position and other positions, obtain the weight of each position, and then sum the weighted features to obtain the output features.

$$\mathrm{NonLocal}(\mathbf{X})_i = \sum_{\forall j} \mathrm{SoftMax}\left(\frac{\phi(\mathbf{X}_i)^{\mathrm{T}} \psi(\mathbf{X}_j)}{\sqrt{M}}\right) \theta(\mathbf{X}_j) = \frac{1}{\sum_{\forall j} e^{\frac{1}{\sqrt{M}} \phi(\mathbf{X}_i)^{\mathrm{T}} \psi(\mathbf{X}_j)}} \sum_{\forall j} e^{\frac{\phi(\mathbf{X}_i)^{\mathrm{T}} \psi(\mathbf{X}_j)}{\sqrt{M}}} \theta(\mathbf{X}_j), \quad (2)$$

where ϕ, ψ, and θ are three kinds of linear embeddings attained by different 1 × 1 convolution kernels that operate on each slice indexed by subscript $_i$ of the feature maps $\mathbf{X}$. For simplicity, the three types of linear embeddings are labelled as $\mathbf{V}_1$, $\mathbf{V}_2$, and $\mathbf{V}_3$ respectively. Non-local modules can be embedded into any convolutional neural network as a component to improve the expression ability of the network. The above operation enables the network to simultaneously encode the relationships between local pixels and between each pixel block without losing the spatial order between pixels or the sequential association between pixel blocks. Since each pixel can sense other pixels, the overall effective receptive field of the module is $H \times W$. Next, the features were reshaped to their respective scales and restored to the original feature size through 1 × 1 convolutional layer. Then, the transformed features $\mathbf{O}_2$ at quarter scale and $\mathbf{O}_3$ at half scale were upsampled to the same size as $\mathbf{O}_1$. Finally, a 3 × 3 convolutional layer was used to fuse local and global features from different scale tensors, as shown in Figure 4.

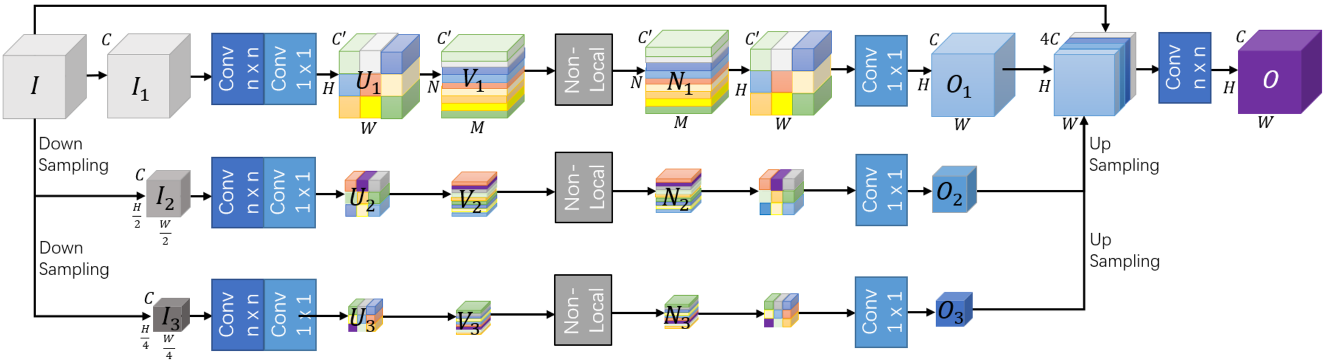


Figure 4. The diagram of our proposed contextual information awareness module.

In this way, the output features contain global information, and the contribution of different positions to the current position is adjusted according to the similarity.

3.5. Loss Function

The aim of forest fire segmentation is to classify every pixel into two semantic categories, the fire or the surrounding backgrounds, according to its region that the pixel belong to. Therefore, the loss function composed of two kinds of losses was used to train the model. The first is binary cross entropy (BCE) loss, whose formula is as follows:

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{i=1}^{N} \log P[i] \cdot \mathbf{I}\{Y[i] = 1\} + \log(1 - P[i]) \cdot \mathbf{I}\{Y[i] = 0\}, \tag{3}$$

where $N = H \times W$ is the number of pixels of the image to be input and i is the position index of each pixel; $Y[i]$ represents the ground truth labeled as 1 or 0, meaning that it is annotated as a part of fire or not; $P[i]$ denotes the probability that the predicted pixel i belongs to the fire area. In order to further improve the performance of the model, dice loss [49] was introduced as the second kind of loss, as shown below:

$$\mathcal{L}_{Dice} = 1 - 2\frac{\sum_{i}^{N} Y[i] \cdot P[i]}{\sum_{i}^{N} Y[i] + P[i]}, \tag{4}$$

Finally, the ensemble loss $\mathcal{L}$ was computed as:

$$\mathcal{L} = \lambda\mathcal{L}_{BCE} + (1 - \lambda)\mathcal{L}_{Dice}, \tag{5}$$

where λ is a trade-off factor between the binary cross entropy loss and the Dice loss. In the subsequent section, we will conduct corresponding experiments on the value of λ. We set λ to 0, 1, and 0.5, which means using only Dice loss and only BCE loss, as well as both to participate in the supervision of our networks training, respectively.

3.6. Overall Architecture of the FBC-ANet Model

Figure 5 shows the overall architecture of the FBC-ANet model proposed in this article, where FEM, BEM, and CIA correspond to the three proposed improvement modules. The FBC-ANet model also has an encoder–decoder architecture. The encoder level is used to extract features from images. Decoder level conducts upsampling to restore the feature map output by the encoder to the approximate size of the original image. Compared with the faster UNet model, the FBC-ANet model has mainly improved in the following aspects.

First of all, the encoder part consists of the full convolution part of the Xception network. Except for the first two convolution layers, which reduce the input image to the 1/2-sized feature graph, the remaining convolution modules are replaced by the conventional convolution to simplify the parameters of the model by deep separable convolution. Four layers of undersampling are interspersed with thirteen depth-separable convolution modules, and the input feature size is reduced to 1/16, which reaches its bottleneck, and residuals are associated with adjacent undersampling to fuse features of different sizes.

Secondly, it adds a CIA module to a bottleneck section, which transforms deep semantic features. Through CIA, it can capture features of different sizes in the image feature graph, enhance its feature extraction ability, and help to identify fire source targets of different sizes.

Again, corresponding to the encoder, the decoder part also goes through four layers of upper sampling to restore the input feature size. Unlike UNet, PSPNet and other methods use concatenation to merge information from encoder. Here, we used 'add' to merge residual error, which is opposite to encoder. It is worth noting that, since the size has been doubled after upsampling, in order to maintain the same size, the residual in the decoder part uses the convolution kernel with a stride of 1, while the residual in the encoder part uses the convolution kernel with a stride of 2.

Finally, after the fusion of residuals, a BEM module was added to strengthen the shallow edge features. This module can obtain boundary information from the low-order network and semantic information from the high-order network, and can then fuse them to avoid the absence of certain information. Higher-order semantic information can optimize lower-order edge information.

In addition, it needs to be noted that the output feature was restored to the input image size through the 5th upper sampling layer and the conventional convolution layer in the decoder part, and sigmoid function was executed to determine whether each pixel is a flame or the surrounding background.

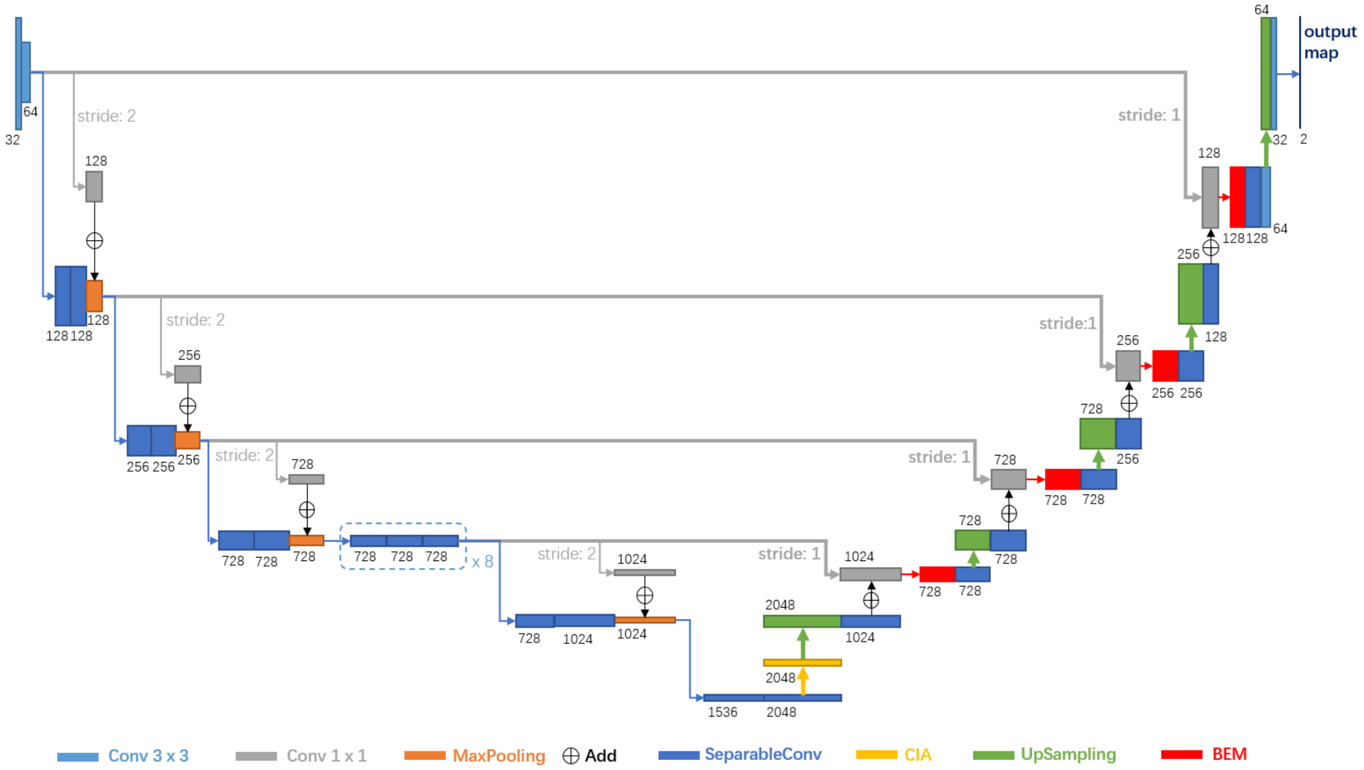


Figure 5. Overall architecture of the FBC-ANet model.

4. Results and Discussion

This section will display the semantic segmentation results of forest fires on the FLAME-Seg dataset. Table 5 lists the experimental environments for the training and testing stages.

Table 5. Experimental environments.

Environment	Type
Operating System	Ubuntu 18.04
Framework	TensorFlow 2.6.0 and Keras 2.6.0
Language	Python 3.7
CPU	Intel(R) Xeon(R) Silver 4110
GPU	GeForce RTX 2080Ti

Table 6 shows the parameter configuration when the FBC-ANet model achieves the best semantic segmentation results.

Table 6. Training configuration.

Configuration	Value
Batch Size	32
Optimizer	Adam
Learning Rate	1×10^{-3}
UpSampling	bi-linear
Epochs	50

4.1. Evaluation Metrics

In order to demonstrate the performance of the FBC-ANet model in detail, accuracy, precision, recall, F1 score, and intersection over union (IoU) [28] were selected as the evaluation indicators for a comprehensive analysis. The following are the calculation formulas for these indicators: *Accuracy* is used to calculate the ratio of the number of correctly segmented pixels to the total number of image pixels. The specific formula is shown as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}. \tag{6}$$

Precision is used to calculate the ratio of all correctly retrieved pixels of fire to all actually retrieved pixels of fire. Like *Accuarcy*, a value closer to 1 means a better performance. The specific formula is shown as follows:

$$Precision = \frac{TP}{TP + FP}. \tag{7}$$

Recall, also known as sensitiveness, indicates the proportion of all correctly retrieved pixels of fire to all actually retrieved pixels of fire. Like *Precision*, the closer its value is to 1, the better the performance of the model. The specific formula is shown as follows:

$$Recall = \frac{TP}{TP + FN}. \tag{8}$$

$F1$ *score* is the harmonic mean of the *Precision* and the *Recall*, namely

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{TP}{TP + (FP + FN)/2}. \tag{9}$$

IoU is the intersection-over-union ratio, which is an indicator used to measure the similarity of the overlap between the predicted pixels of fire and the ground truth, which are the most commonly used and most frequent evaluation metrics in the community of semantic image segmentation. The higher the IoU metric, the better the semantic segmentation effect. The specific formula is shown as follows:

$$IoU = \frac{TP}{TP + FP + FN}. \tag{10}$$

Among them, TP is the abbreviation for true positive, representing the number of pixels predicted to belong to a fire and the actual number of pixels belonging to a fire; FP is the abbreviation for false positive, indicating the number of pixels predicted to belong to a fire but actually belonging to the background; FN is the abbreviation for false negative, indicating the number of pixels predicted to belong to the background but actually belonging to the fire; TN is the abbreviation for true negative, which refers to the number of pixels predicted to belong to the background and actually also belonging to the background.

4.2. Parameters Settings and Ablation Experiments

Figure 6 shows the loss curve and accuracy curve of the FBC-ANet model. It can be seen that the loss curve of the FBC-ANet model shows a stable convergence state after 10 epochs, both in the training and validation sets. Correspondingly, its accuracy curve also reaches stability after 10 epochs, with accuracy values close to 1 on both the training and validation sets. Figure 7 shows the precision and recall curve of the FBC-ANet model. In the first 30 epochs, the precision curve and recall curve of the model show significant fluctuations in the validation set, although they are smooth in the training set. This indicates that, as the number of epochs increases, the FBC-ANet model exhibits a similar stable performance in the validation set as in the training set; that is, the FBC-ANet model does not produce overfitting on the training set, which reflects its relatively reliable generalization ability. However, it should be pointed out that, due to the limitations of the dataset size, the performance may not be satisfactory in other practical environments. However, given the good performance on the FLAME dataset, we believe that training and testing on larger datasets can still achieve a good performance.

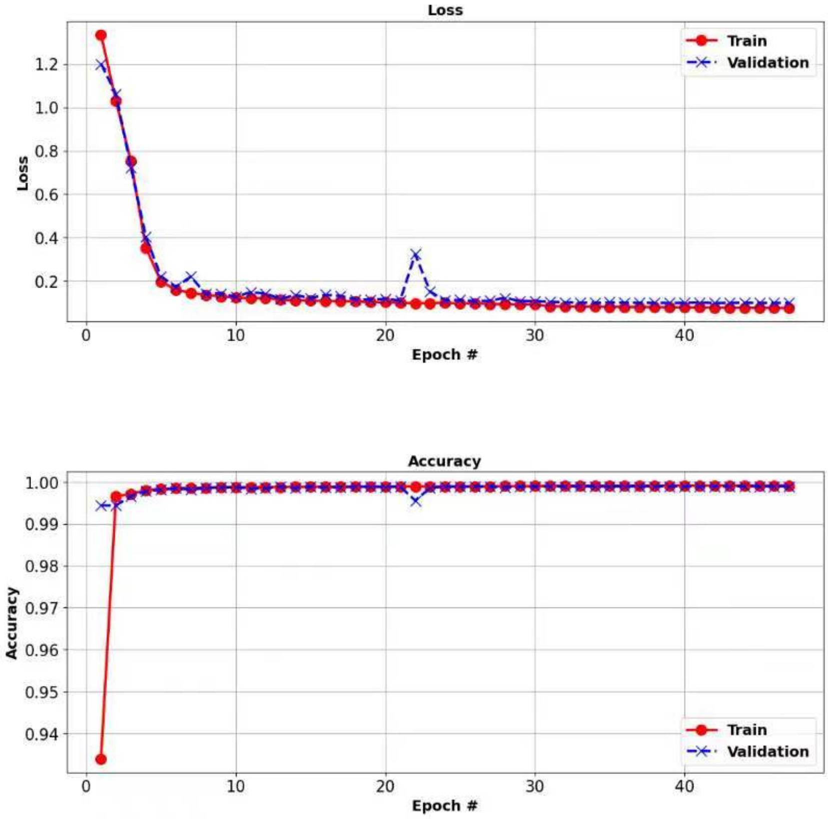


Figure 6. The plots of loss and accuracy regressing on the training set and the validation set.

To demonstrate the contribution of each module to the performance of the FBC-ANet model, ablation experiments were conducted. Table 7 shows the performance of the semantic segmentation of forest fire images under different model architectures. Among them, model 0 represents the baseline method PSPNet [30], and its feature extraction backbone is ResNet50 [47]. Model 1 is a model that uses Xception [27] as the feature extraction backbone network. Model 2 and Model 3 add a boundary enhancement module (BEM) and context information awareness (CIA) module based on Model 1, respectively. Model 4 is a model that includes all improved modules, namely the FBC-ANet model proposed in the current work. Comparing the effectiveness of various models in the semantic segmentation of UAVs fire images, it can be seen that using Xception instead of ResNet50 as the feature extraction backbone network significantly improves the effectiveness of the semantic segmentation of fire images. Continuing to add BEM and CIA modules on top of the Xception module can also help to improve the performance of semantic segmentation. The FBC-ANet model generated by integrating three modules achieves the best semantic segmentation performance, with an F1 value of 90.76% and an IoU value of 83.08%. In the FBC-ANet model, each module complements each other and has a beneficial additive effect on the semantic segmentation performance of the model.

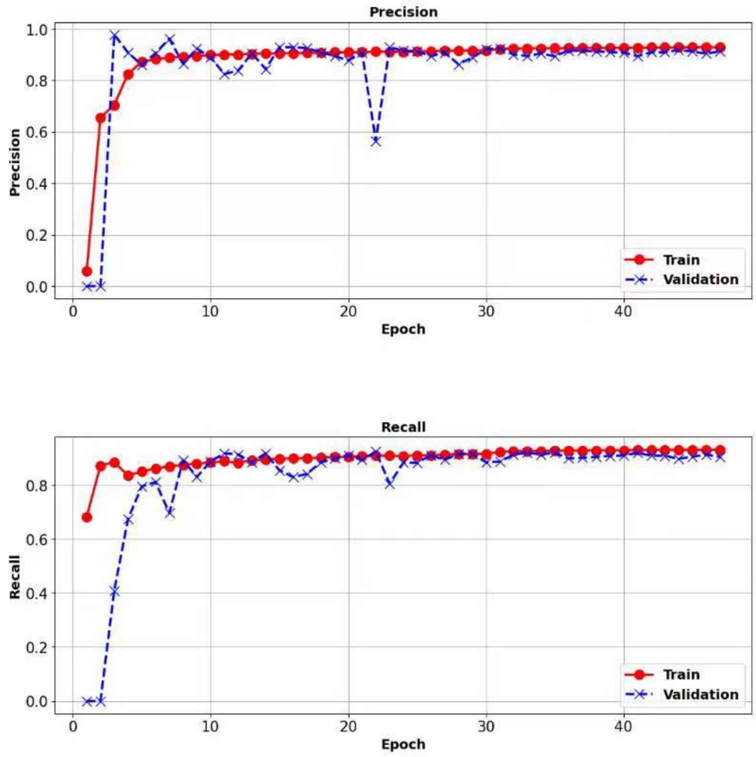


Figure 7. The plots of precision rate and recall rate on the training set and the validation set.

Table 7. Comparison of segmentation performance among different models. The best is shown in **bold** font.

Model	FEM	BEM	CIA	Precision (%)	Recall (%)	F1 Score (%)	IoU (%)
0				87.89	85.02	85.91	76.43
1	✓			91.62	87.59	89.56	81.09
2	✓	✓		91.91	86.94	89.36	80.76
3	✓		✓	91.76	87.48	89.57	81.11
4	✓	✓	✓	**92.19**	**89.37**	**90.76**	**83.08**

There are multiple optional settings during the training process of the FBC-ANet model. Comparative experiments were conducted on various settings to select the optimal configuration parameters.

Three loss functions and two upsampling modes were compared when training the FBC-ANet model. The segmentation performance on the FLAME-Seg dataset is shown in Table 8. The three kinds of loss function are: binary cross entropy loss $\mathcal{L}_{BCE}$, dice loss $\mathcal{L}_{Dice}$, and the loss function combining them. The two upsampling modes are bi-linear interpolation (Bi-Linear) and deconvolution operation (DeConv). The results show that, regardless of whether Bi-Linear or DeConv is used for upsampling, the FBC-ANet model with joint loss function achieves the best semantic segmentation performance. Using $\mathcal{L}_{Dice}$ alone will bring adverse effects on backpropagation and lead to unstable training. Therefore, the combination of $\mathcal{L}_{BCE}$ and $\mathcal{L}_{Dice}$ can achieve the highest segmentation performance. In addition, Bi-Linear has a slight advantage in indicators of precision and F1 score, while DeConv is better in recall and IoU. In order to reduce the number of training parameters, Bi-Linear with fewer parameters was selected for up-sampling in the FBC-ANet model.

Table 8. Comparison of different upsampling strategies and loss functions. The best is shown in **bold** font.

UpSampling	Loss Function	Precision (%)	Recall (%)	F1 Score (%)	IoU (%)
Bi-Linear	$\mathcal{L}_{BCE}$	91.93	89.17	90.53	82.70
	$\mathcal{L}_{Dice}$	91.92	89.53	90.71	83.01
	$\mathcal{L}_{BCE} + \mathcal{L}_{Dice}$	**92.19**	**89.37**	**90.76**	**83.08**
DeConv	$\mathcal{L}_{BCE}$	91.83	89.06	90.44	82.56
	$\mathcal{L}_{Dice}$	91.99	89.19	90.57	82.77
	$\mathcal{L}_{BCE} + \mathcal{L}_{Dice}$	92.02	89.57	90.74	83.11

Table 9 shows the semantic segmentation performance of the FBC-ANet model under different batches. Generally, the larger the batch size, the more accurate the descent direction of the loss function. The smaller the batch size, the stronger the generalization ability of the model, but it is also prone to falling into local optimization. It can be seen that, when the batch size is set to 32, the FBC-ANet model achieves the best semantic segmentation performance. It is worth noting that the batch size is also related to the selection of the learning rate. Choosing an appropriate learning speed also helps the model to achieve a better semantic segmentation performance.

Table 9. Effect of different batch sizes on the performance of forest fire semantic segmentation. The best is shown in **bold** font.

Batch Size	Precision (%)	Recall (%)	F1 Score (%)	IoU (%)
4	92.08	88.22	90.11	81.99
8	92.13	89.02	90.54	82.76
16	91.92	89.45	90.67	82.94
32	**92.19**	**89.37**	**90.76**	**83.08**

Figure 8 shows the experimental results of two depth-separable convolution strategies applied to the test set. Among them, Mask1 represents the results of adopting the depthwise separable convolution, and Mask2 represents the strategy of first pointwise convolution and then depthwise convolution adopted in Xception. Each image contains 512 × 512 pixels, most of which are background pixels, where only a small portion are flame pixels. Figure 8 shows that both depth-separable convolution strategies can accurately divide the area of forest fires. We compared the two constructions of deeply separable convolution in order to exclude the effect of structural differences because Xception is an extreme version of a natural extrapolation from the inception network's perspective. Except for some differences in fire details, the performance of the two strategies is almost the same.

Figure 9 shows the confusion matrix on the test set. It can be seen that the TP and the TN are much higher than the FP and the FN. This result strongly demonstrates the powerful ability of the FBC-ANet model in distinguishing and segmenting fire pixels, demonstrating its effectiveness in solving difficulties such as an imbalanced class distribution and complex environmental backgrounds.

Figure 10 shows the PR curve and ROC curve to further validate the performance of the FBC-ANet model. PR curves and ROC curves are commonly used graphical tools for evaluating the classifier performance. In the PR curve, the horizontal axis represents the recall rate and the vertical axis represents the accuracy rate, giving the model the ability to recognize the correct samples. In Figure 10, average precision (AP) is defined as the area enclosed by the coordinate axis under the PR curve. The closer the AP is to 1, the better the classification performance of the model. In the ROC curve, the horizontal axis represents the false positive rate and the vertical axis represents the true positive rate, which can be used to characterize the classifier's ability to distinguish between positive and negative samples. The area under curve (AUC) in Figure 10 is defined as the area under the ROC curve around the coordinate axis. The closer the AUC is to 1, the stronger the discriminative ability of the model. The PR curve focuses more on positive samples, while the ROC curve considers both positive and negative samples. When the ratio of flame to background is similar, the difference between the two is not significant. However, in the FLAME-Seg dataset, the ratio of flame to background is extremely unbalanced; that is, the ratio of background pixels is extremely high. From Figure 10, it can be seen that the FBC-ANet model can still correctly identify the relatively small ratio of fire pixels. The AP value of the FBC-ANet model is 0.9662, which intuitively demonstrates the model's ability to recognize and distinguish fire pixels.

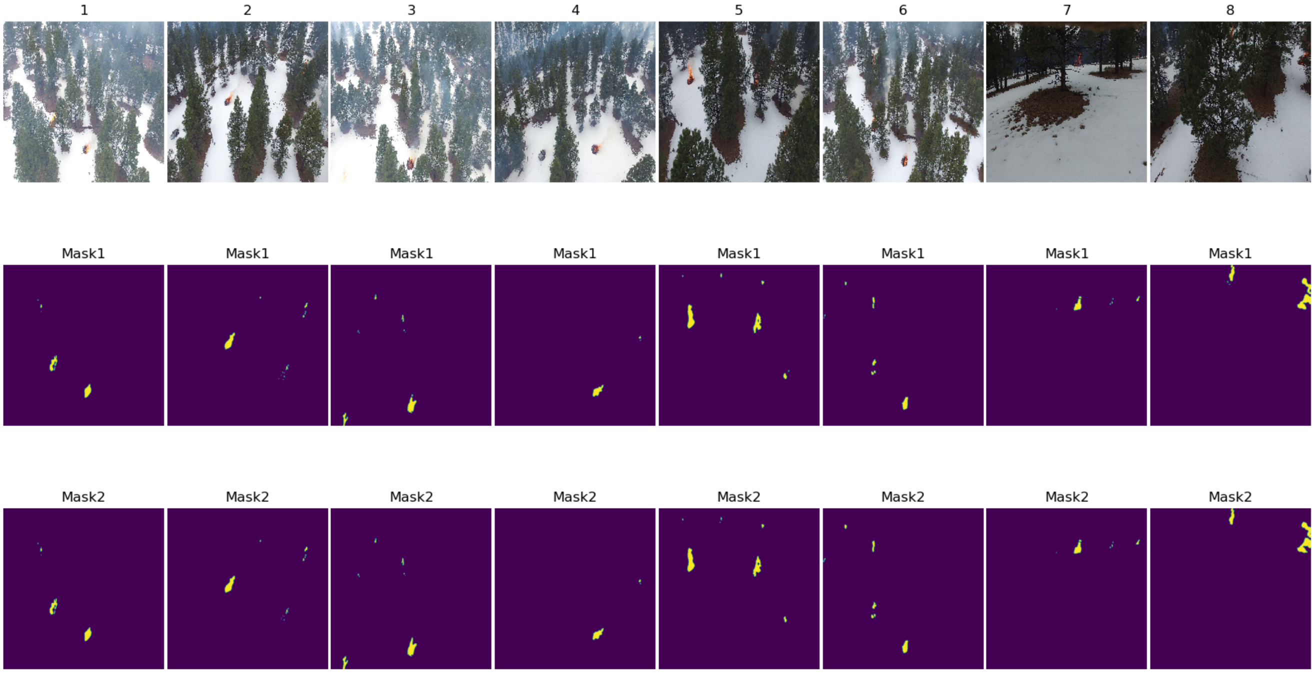


Figure 8. The results of adopting two schemes of separable convolution. Mask1 represent the results of adopting the depthwise separable convolution, while Mask2 represent the results of adopting separable convolution in Xception.

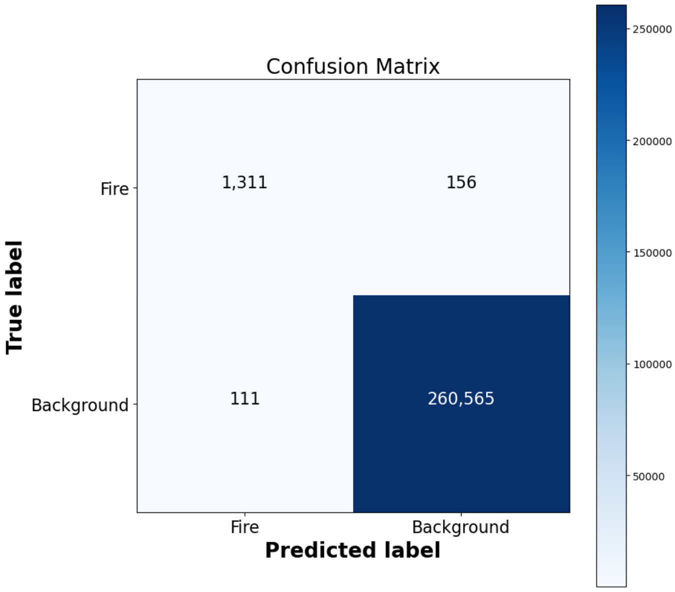


Figure 9. Confusion matrix.

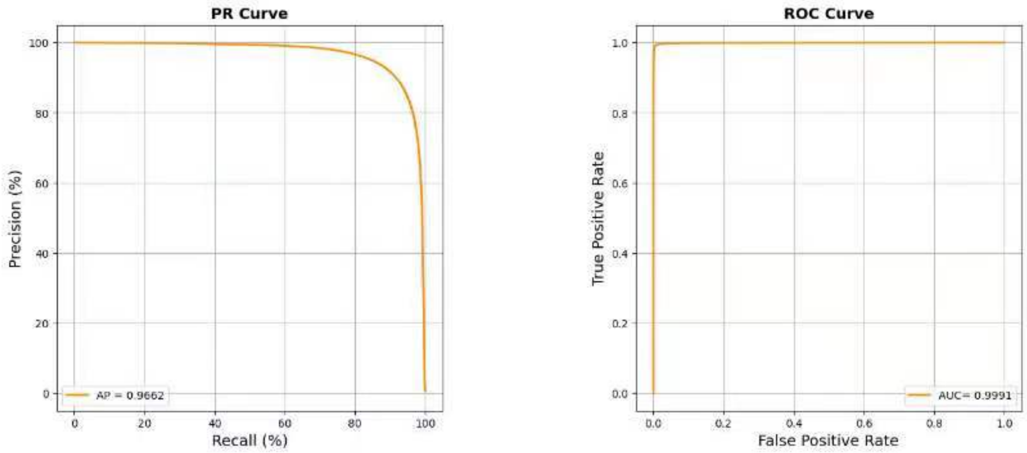


Figure 10. PR curve and ROC curve.

Comparison with Other Segmentation Methods

Table 10 shows the semantic segmentation performance of the FBC-ANet model and the baseline model on the FLAME-Seg dataset. In Table 10, the first five models are five classic semantic segmentation models, Mask SU-RCNN [24] is a recently proposed SOTA semantic segmentation model specifically for forest fires, where a dual semantic attention (DSA) mechanism is proposed and the DSA module is merged into ResNet as the backbone network to enhance the representation ability of feature channels. From Table 10, it can be seen that, compared with various baseline models, the model outperforms the classical segmentation model, and the FBC-ANet model achieves the best semantic segmentation performance, demonstrating competitive advantages under all evaluation indicators. Compared to these baseline models, the outstanding advantages of the FBC-ANet model in forest fire semantic segmentation tasks come from the three core modules of the model: the feature extraction module, boundary enhancement module, and context information perception module. It is precisely the mutual supplementation and strengthening of information among these three modules that enables the FBC-ANet model to accurately segment fire pixels from complex fire backgrounds, which will lay an important technical foundation for the construction of subsequent forest fire monitoring systems.

Table 10. Quantitative comparisons of FBC-ANet with other SOTA methods on FLAME dataset. w/o means without, w/means with. The best is shown in **bold** font.

Method	Precision (%)	Recall (%)	F1 Score (%)	IoU (%)
UNet	84.75	76.22	80.82	67.23
SegNet [50]	85.21	78.65	81.80	71.12
RefineNet	88.80	82.95	85.78	76.22
PSPNet	87.89	85.02	85.91	76.43
DeepLab	90.01	85.10	87.01	80.20
FLAME	91.99	83.88.	87.75	78.17
MaskSU R-CNN (w/o DSA)	88.63	88.89	88.76	80.77
MaskSU R-CNN (w/DSA)	91.85	88.81	90.30	82.31
FBC-ANet	**92.19**	**89.37**	**90.76**	**83.08**

4.3. *Visualization of Segmentation Results*

Figure 11 shows the visualization results of fire semantic segmentation for each model on the FLAME dataset. It can be seen that the FBC-ANet model can clearly recognize the position of flames and accurately segment flame pixels even in complex backgrounds obscured by thick smoke. Relatively speaking, the PSPNet model, which serves as the main reference for the FBC-ANet model, may lead to discontinuity in target prediction and fail to recognize small forest fire areas hidden behind trees, such as the second row of flames (image #561). Therefore, the FBC-ANet model can more accurately locate the fire source target and can have a clearer target contour after introducing a boundary enhancement module and a context information perception module.

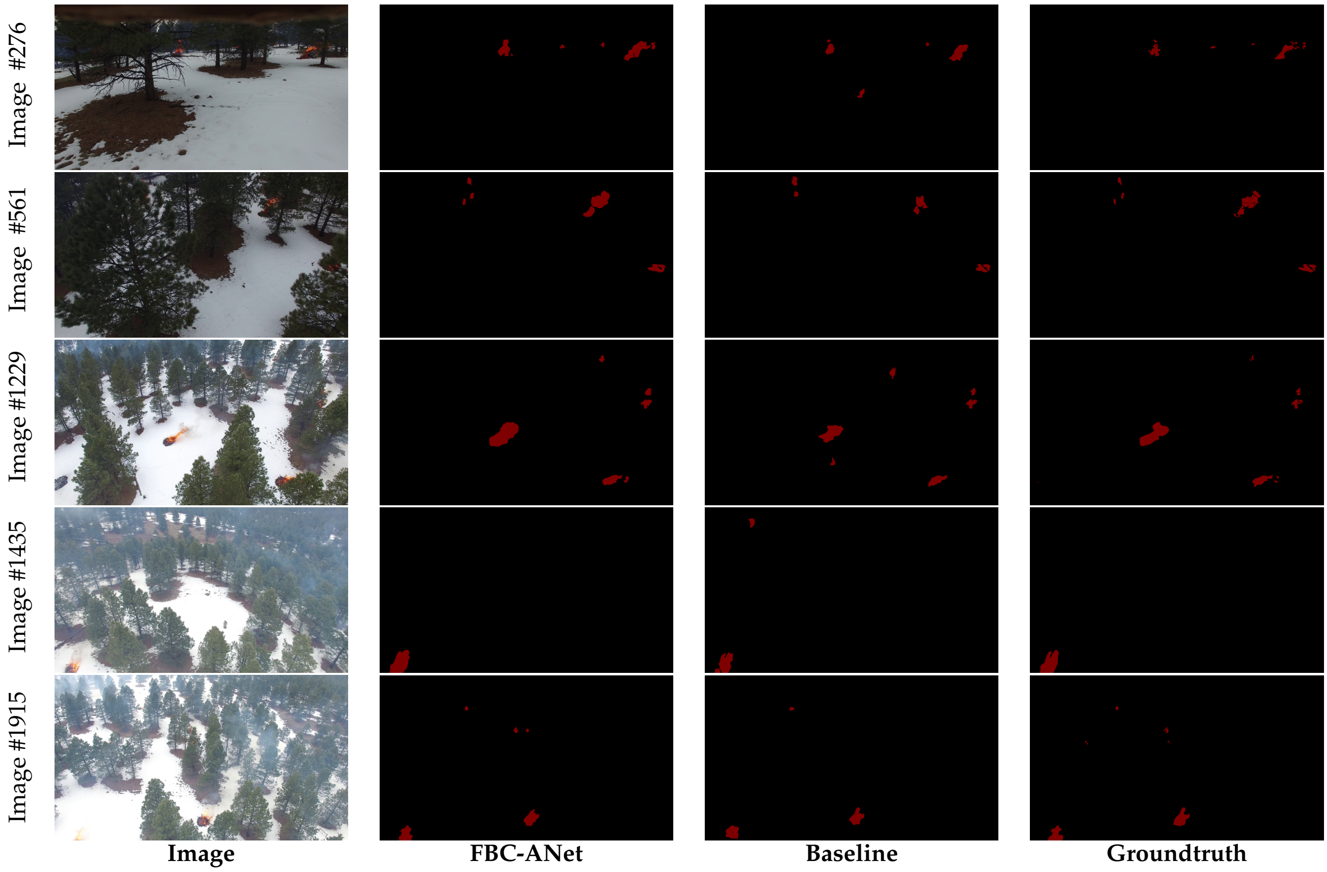


Figure 11. Visualization of forest fire image segmentation results.

5. Conclusions

Forests are the green treasure trove of humanity and the material foundation for human survival. However, at the same time, forest resources are also seriously threatened by fires. The early monitoring and prevention of forest fires based on UAVs is an effective means to avoid the spread of fires and significant losses. The current work proposed an FBC-ANet model based on an encoder–decoder framework for the semantic segmentation of fire images to accurately segment fire pixels in UAV images, thereby providing technical support for the timely identification of fires. The FBC-ANet model uses depthwise separable convolution as the backbone feature extraction network combined with a boundary enhancement module and context information perception module to enhance the model's recognition and segmentation performance for forest fire pixels. Compared with classic semantic segmentation models such as U-Net, PSPNet, SegNet, RefineNet, and DeepLabv3+, as well as the recently proposed SOTA model specifically for forest fire semantic segmentation such as FLAME and MaskSU R-CNN, the FBC-ANet model has shown strong competitive advantages. FBC-ANet can effectively overcome challenging issues such as small fire areas and background complexity, and has the ability to more accurately segment fire pixels and detect the precise shape of fires. Nowadays, deep learning technology is widely applied to the field of sensing and discovering forest fires. There are still some challenging research directions in this field. In future research, we would engage in expanding our model to classify whether flames occur in forests at night and dawn, even in complex urban environments, and to determine how to accurately detect and segment them assisted by TIR images as well.

Author Contributions: Conceptualization, L.Z.; methodology, L.Z.; software, M.W.; validation, L.Z.; formal analysis, L.Z. and T.W.; investigation, L.Z.; resources, L.Z. and Y.P.; data curation, L.Z.; writing—original draft preparation, L.Z.; writing—review and editing, M.W.; visualization, M.W.; supervision, M.W. and B.Q.; funding acquisition, M.W.; Project administration, Y.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 71473034, and the Heilongjiang Provincial Natural Science Foundation of China, grant number LH2019G001.

Data Availability Statement: The FLAME dataset is available at https://ieee-dataport.org/open-access/flame-dataset-aerial-imagery-pile-burn-detection-using-drones-uavs (accessed on 19 January 2023).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

UAV	Unmanned aerial vehicle
TP	True positive
TN	True negative
FP	False positive
FN	False negative
IoU	Intersection over union

References

1. Dimitropoulos, S. Fighting fire with science. *Nature* **2019**, *576*, 328–329. [CrossRef]
2. Aytekin, E. Wildfires Ravaging Forestlands in Many Parts of Globe. 2021. Available online: https://www.aa.com.tr/en/world/wildfires-ravaging-forestlands-in-many-parts-of-globe/2322512 (accessed on 20 February 2023).
3. Huang, Q.; Razi, A.; Afghah, F.; Fule, P. Wildfire Spread Modeling with Aerial Image Processing. In Proceedings of the 2020 IEEE 21st International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Cork, Ireland, 31 August–3 September 2020; pp. 335–340.
4. Friedlingstein, P.; Jones, M.; O'Sullivan, M.; Andrew, R.; Hauck, J.; Peters, G.; Peters, W.; Pongratz, J.; Sitch, S.; Le Quéré, C.; et al. Global carbon budget 2019. *Earth Syst. Sci. Data* **2019**, *11*, 1783–1838. [CrossRef]
5. Erdelj, M.; Natalizio, E.; Chowdhury, K.R.; Akyildiz, I.F. Help from the sky: Leveraging UAVs for disaster management. *IEEE Pervasive Comput.* **2017**, *16*, 24–32. [CrossRef]
6. Shamsoshoara, A.; Afghah, F.; Razi, A.; Mousavi, S.; Ashdown, J.; Turk, K. An Autonomous Spectrum Management Scheme for Unmanned Aerial Vehicle Networks in Disaster Relief Operations. *IEEE Access* **2020**, *8*, 58064–58079. [CrossRef]
7. Mousavi, S.; Afghah, F.; Ashdown, J.D.; Turck, K. Use of a quantum genetic algorithm for coalition formation in large-scale uav networks. *Hoc Netw.* **2019**, *87*, 26–36. [CrossRef]
8. Mahmudnia, D.; Arashpour, M.; Bai, Y.; Feng, H. Drones and Blockchain Integration to Manage Forest Fires in Remote Regions. *Drones* **2022**, *6*, 331. [CrossRef]
9. Saffre, F.; Hildmann, H.; Karvonen, H.; Lind, T. Monitoring and Cordoning Wildfires with an Autonomous Swarm of Unmanned Aerial Vehicles. *Drones* **2022**, *6*, 301. [CrossRef]
10. Gaur, A.; Singh, A.; Kumar, A.; Kumar, A.; Kapoor, K. Video flame and smoke based fire detection algorithms: A literature review. *Fire Technol.* **2020**, *56*, 1943–1980. [CrossRef]
11. Ghali, R.; Jmal, M.; Souidene Mseddi, W.; Attia, R. Recent advances in fire detection and monitoring systems: A review. In Proceedings of the 18th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT'18), Genoa, Italy, 20–22 December 2018; pp. 332–340.
12. Huang, L.; Liu, G.; Wang, Y.; Yuan, H.; Chen, T. Fire detection in video surveillances using convolutional neural networks and wavelet transform. *Eng. Appl. Artif. Intell.* **2022**, *110*, 104737. [CrossRef]
13. Ahmad Khan, Z.; Hussain, T.; Min Ullah, F.U.; Gupta, S.K.; Lee, M.Y.; Baik, W.S. Randomly Initialized CNN with Densely Connected Stacked Autoencoder for Efficient Fire Detection. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105403. [CrossRef]
14. Lin, J.; Lin, H.; Wang, F. STPM_SAHI: A Small-Target Forest Fire Detection Model Based on Swin Transformer and Slicing Aided Hyper Inference. *Forests* **2022**, *13*, 1603. [CrossRef]
15. Harkat, H.; Nascimento, J.; Bernardino, A.; Thariq Ahmed, H.F. Fire images classification based on a handcraft approach. *Expert Syst. Appl.* **2023**, *212*, 118594. [CrossRef]

16. Guede-Fernández, F.; Martins, L.; de Almeida, R.V.; Gamboa, H.; Vieira, P. A Deep Learning Based Object Identification System for Forest Fire Detection. *Fire* **2021**, *4*, 75. [CrossRef]
17. Alipour, M.; La Puma, I.; Picotte, J.; Shamsaei, K.; Rowell, E.; Watts, A.; Kosovic, B.; Ebrahimian, H.; Taciroglu, E. A Multimodal Data Fusion and Deep Learning Framework for Large-Scale Wildfire Surface Fuel Mapping. *Fire* **2023**, *6*, 36. [CrossRef]
18. Ghali, R.; Akhloufi, M.A.; Jmal, M.; Souidene Mseddi, W.; Attia, R. Wildfire Segmentation Using Deep Vision Transformers. *Remote Sens.* **2021**, *13*, 3527. [CrossRef]
19. Harkat, H.; Nascimento, J.M.P.; Bernardino, A.; Thariq Ahmed, H.F. Assessing the Impact of the Loss Function and Encoder Architecture for Fire Aerial Images Segmentation Using Deeplabv3+. *Remote Sens.* **2022**, *14*, 2023. [CrossRef]
20. Toulouse, T.; Rossi, L.; Campana, A.; Celik, T.; Akhloufi, M.A. Computer vision for wildfire research: An evolving image dataset for processing and analysis. *Fire Saf. J.* **2017**, *92*, 188–194. [CrossRef]
21. Shamsoshoara, A.; Afghah, F.; Razi, A.; Zheng, L.; Fulé, P.Z.; Blasch, E. Aerial Imagery Pile Burn Detection Using Deep Learning: The FLAME Dataset. *Comput. Netw.* **2021**, *193*, 142–149. [CrossRef]
22. Avazov, K.; Mukhiddinov, M.; Makhmudov, F.; Cho, Y.I. Fire Detection Method in Smart City Environments Using a Deep-Learning-Based Approach. *Electronics* **2022**, *11*, 73. [CrossRef]
23. Norkobil Saydirasulovich, S.; Abdusalomov, A.; Jamil, M.K.; Nasimov, R.; Kozhamzharova, D.; Cho, Y.-I. A YOLOv6-Based Improved Fire Detection Approach for Smart City Environments. *Sensors* **2023**, *23*, 3161. [CrossRef]
24. Guan, Z.; Miao, X.; Mu, Y.; Sun, Q.; Ye, Q.; Gao, D. Forest fire segmentation from aerial imagery data using an improved instance segmentation model. *Remote Sens.* **2022**, *14*, 3159. [CrossRef]
25. Huang, Z.; Huang, L.; Gong, Y.; Huang, C.; Wang, X. Mask scoring R-Cnn. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6409–6418.
26. Ghali, R.; Akhloufi, M.A.; Mseddi, W.S. Deep Learning and Transformers Approaches for UAV Based Wildfire Detection and Segmentation. *Sensors* **2022**, *22*, 1977. [CrossRef] [PubMed]
27. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1800–1807.
28. Shelhamer, E.; Long, J.; Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651. [CrossRef]
29. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
30. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6230–6239.
31. Lin, G.; Milan, A.; Shen, C.; Reid, I.D. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5168–5177.
32. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 833–851.
33. Mottaghi, R.; Chen, X.; Liu, X.; Cho, N.G.; Lee, S.W.; Fidler, S.; Urtasun, R.; Yuille, A. The role of context for object detection and semantic segmentation in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 891–898.
34. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3213–3223.
35. Zhou, B.; Zhao, H.; Puig, X.; Fidler, S.; Barriuso, A.; Torralba, A. Scene parsing through ade20k dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5122–5130.
36. Caesar, H.; Uijlings, J.; Ferrari, V. COCO-Stuff: Thing and stuff classes in context. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1209–1218.
37. Wu, H.; Zhang, J.; Huang, K.; Liang, K.; Yu, Y. FastFCN: Rethinking dilated convolution in the backbone for semantic segmentation. *arXiv* **2019**, arXiv:1903.11816.
38. Allison, R.S.; Johnston, J.M.; Craig, G.; Jennings, S. Airborne optical and thermal remote sensing for wildfire detection and monitoring. *Sensors* **2016**, *16*, 1310. [CrossRef] [PubMed]
39. Valero, M.M.; Rios, O.; Mata, C.; Pastor, E.; Planas, E. An integrated approach for tactical monitoring and data-driven spread forecasting of wildfires. *Fire Saf. J.* **2017**, *91*, 835–844. [CrossRef]
40. Paul, S.E.; Salvaggio, C. A polynomial regression approach to subpixel temperature extraction from a single-band thermal infrared image. *Proc. SPIE* **2011**, *8013*, 801302.
41. DJI. Phantom 3 Professional. Available online: https://www.dji.com/phantom-3-pro (accessed on 16 April 2023).
42. DJI. Matrice 200 V1. Available online: https://www.dji.com/matrice-200-series/info#specs (accessed on 16 April 2023).
43. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.

44. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
45. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
46. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
47. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
48. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-Local Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
49. Sudre, C.H.; Li, W.; Vercauteren, T.; Ourselin, S.; Jorge Cardoso, M. Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer International Publishing: Cham, Switzerland, 2017; pp. 240–248.
50. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [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.