

Article

Superpixel-Based Graph Convolutional Network for UAV Forest Fire Image Segmentation

Yunjie Mu, Liyuan Ou, Wenjing Chen, Tao Liu and Demin Gao * 

College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China; muyunjie@njfu.edu.cn (Y.M.); liyuanou@njfu.edu.cn (L.O.); wjchen@njfu.edu.cn (W.C.); liutao@njfu.edu.cn (T.L.)
* Correspondence: dmgaon@njfu.edu.cn

Abstract: Given the escalating frequency and severity of global forest fires, it is imperative to develop advanced detection and segmentation technologies to mitigate their impact. To address the challenges of these technologies, the development of deep learning-based forest fire surveillance has significantly accelerated. Nevertheless, the integration of graph convolutional networks (GCNs) in forest fire detection remains relatively underexplored. In this context, we introduce a novel superpixel-based graph convolutional network (SCGCN) for forest fire image segmentation. Our proposed method utilizes superpixels to transform images into a graph structure, thereby reinterpreting the image segmentation challenge as a node classification task. Additionally, we transition the spatial graph convolution operation to a GraphSAGE graph convolution mechanism, mitigating the class imbalance issue and enhancing the network's versatility. We incorporate an innovative loss function to contend with the inconsistencies in pixel dimensions within superpixel clusters. The efficacy of our technique is validated on two different forest fire datasets, demonstrating superior performance compared to four alternative segmentation methodologies.

Keywords: superpixel; segmentation; convolutional neural network; graph convolution network; forest fire



Citation: Mu, Y.; Ou, L.; Chen, W.; Liu, T.; Gao, D. Superpixel-Based Graph Convolutional Network for UAV Forest Fire Image Segmentation. *Drones* **2024**, *8*, 142. <https://doi.org/10.3390/drones8040142>

Academic Editors: Brian K. Gullett, Johanna Aurell, Pantelis Velanas, Diego González-Aguilera and Katerina Margariti

Received: 17 January 2024
Revised: 28 March 2024
Accepted: 28 March 2024
Published: 3 April 2024



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

1. Introduction

Forest fires, resulting from both natural and anthropogenic factors, pose a significant threat to global biodiversity and the ecological balance [1]. Globally, an average of over 200,000 forest fires occur annually, resulting in not only substantial economic losses but also long-term damage to ecosystems [2]. Against this backdrop, the development of efficient fire monitoring and imaging technologies has become key to addressing this challenge. Traditional monitoring methods, such as manual surveillance from watchtowers and detection via infrared instruments on helicopters, have played a role in early fire identification but face issues of high costs, low efficiency, and limited coverage. Consequently, utilizing imaging technology for fire surveillance and early warning not only provides real-time data on fire behavior but also aids in disaster assessment and the formulation of response strategies.

However, in the context of forest fire monitoring and management, merely detecting fires is often insufficient. Fire detection can quickly identify the presence of a fire, but for disaster response and management decisions, it is crucial to accurately determine the fire's specific location, extent, and intensity. Fire segmentation, which precisely delineates the fire area from non-fire areas through image analysis techniques, is a key step in achieving this goal. It not only helps assess the actual impact range of the fire but also provides important information on the fire's spread trend, thereby supporting the formulation of more effective firefighting strategies and resource allocation. Therefore, in forest fire monitoring and management, combining fire detection with further fire segmentation is indispensable for achieving rapid and accurate disaster response. But the processing of forest fire images presents unique challenges. The complex backgrounds, inadequate

contrast, visual obstructions, and irregular shapes of fires all increase the difficulty of image segmentation [3]. These factors lead to the inefficiency of traditional image processing methods in accurately identifying and segmenting fire areas, failing to meet the needs for rapid response [4,5]. Therefore, exploring new technologies and methods capable of overcoming these challenges is crucial for enhancing the accuracy and efficiency of forest fire monitoring and imaging.

Traditional image segmentation methods typically rely on pixel-level processing, treating each pixel as an independent unit. Such coarse-grained representation can increase computational complexity and lower algorithm efficiency when processing forest fire images. Furthermore, they fail to adequately capture the contextual information and spatial relationships within the image, resulting in inaccuracies in the segmentation outcome [6]. In recent years, deep learning has achieved significant success in various fields, including computer vision [7], machine translation [8,9], image recognition [10,11], and speech recognition [12], and has been widely employed in image segmentation tasks [13]. Among these, the graph convolutional network, a powerful deep learning model, can effectively learn the spatial relationships and semantic information in images. Early research on graph convolutional networks adhered to a recursive method, where vertex representation was learned by iteratively propagating information among its neighbors until a stable state was reached [14,15]. More recently, Liu et al. [16] introduced a novel, multifeature fusion network that combines multiscale graph convolutional networks (GCNs) and multiscale convolutional neural networks (CNNs) for the classification of hyperspectral imaging system (HIS) images. They demonstrated the effectiveness of their approach in achieving accurate classification results. Wang et al. [17] developed a similar architecture called UNet Transformer for real-time urban scene segmentation. In this model, they employed a lightweight ResNet18 encoder to capture global and local information. To simulate the integration of global and local information in the decoder, they designed a sophisticated global–local attention mechanism, which enhanced the segmentation performance of the model. In the work by Wu et al. [18], a graph neural network (GNN) model was proposed based on the feature similarity of multiview images. They established correlation nodes between multiview images and library images, enabling the transformation of graph node features into correlation features between images. Furthermore, they designed an image-based region feature extraction method, which simplified the image preprocessing process and better extracted important image characteristics for improved performance.

However, in traditional GCN models [19], each pixel is treated as a node, an approach that could lead to information loss and increased computational complexity. To overcome the limitations of conventional GCN models and enhance the accuracy and efficiency of forest fire image segmentation, we propose a graph convolutional network based on superpixels in this study. Superpixels, a technique for segmenting images, partition the image into contiguous regions exhibiting similar texture and color characteristics. This method not only provides precise edge information but also captures the object details effectively [2,20]. Several commonly used superpixel segmentation algorithms include simple linear iterative clustering (SLIC) [21], superpixels extracted via energy-driven sampling (SEEDS) [22], and superpixel segmentation using Gaussian mixture models (GMMSP) [23]. Researchers often utilize superpixel segmentation as a preprocessing step for image segmentation tasks. For instance, Belizario et al. [24] employed superpixels for pre-segmentation, derived feature matrices based on color information, and proposed an automatic image segmentation method based on weighted recursive label propagation (WRLP), which quantifies the similarity between superpixels by utilizing edge weights. Xiong and Yan [25] developed a novel superpixel merging technique to address the over-segmentation problem in single-frame video sequence images and employed support vector machines (SVMs) for spectral-based superpixel classification. There are also studies that combine CNNs with SLIC for image segmentation [26]. For example, digital methods of superpixel segmentation and convolutional neural networks were used to segment trees in a forest environment. However, CNNs are primarily designed for handling Euclidean

space data and may encounter limitations when applied to graph-structured data [27]. This is because CNNs typically assume that data are uniformly distributed in space, an assumption that may not hold when dealing with graph-structured data characterized by complex spatial distributions, such as forest fires. The dynamic and uncontrolled expansion of forest fires and their potential to damage ecosystems necessitate real-time monitoring and grading of these events to achieve accurate forecasting and control. However, traditional CNNs, due to their inability to handle non-uniformly distributed spatial data effectively, may not achieve this objective when processing such graph-structured data.

To overcome the aforementioned issues, we propose an algorithm based on graph convolutional networks (GCNs) utilizing superpixels. Initially, the algorithm utilizes the color and spatial proximity information of superpixels to partition the image into multiple superpixel blocks. Subsequently, each superpixel block is considered a graph node, and edges between these nodes are constructed based on the regional color space characteristics of the image. Then, features are extracted from each superpixel block using a convolutional neural network. Through iterative training of the GCN, graph node classification is performed, and node class labels are associated with the corresponding superpixel blocks, resulting in the final segmentation outcome. The novelty of our proposed algorithm includes the following aspects:

(1) We have developed a superpixel-based graph convolutional network model specifically for forest fire image segmentation. To address the inevitable loss of boundary information caused by resizing input images to their original sizes after passing through the encoder, we propose a preprocessing step that converts grid-structured images into graph-structured ones using superpixels. Specifically, our model performs node prediction for each image converted into a superpixel graph. This preprocessing step can preserve essential boundary information, thereby enhancing the overall performance of the segmentation process.

(2) We introduced a novel forest fire image segmentation approach based on both convolutional neural networks and graph convolutional networks. We enhance the graph convolutional operator of the GCN by utilizing GraphSAGE's operator. Specifically, the CNN is employed to extract features from superpixel blocks, while the GCN is used to predict node labels within the graph.

(3) To address the issues of class imbalance and varying pixel sizes within superpixel blocks, we introduce a novel loss function. This function imposes varying degrees of penalties on superpixel blocks of different classes and sizes, thereby effectively managing imbalanced data.

The rest of this paper is organized as follows. In Section 2, the forest fire dataset and the methods and modules used in the experiments are introduced. The forest fire segmentation model presented in this paper is also elaborated upon in this section. Section 3 presents the experimental results of each part of the improvements. Section 4 describes the discussion and analysis of the model, as well as the outlook for future work. A summary of the entire work is presented in Section 5.

2. Materials and Methods

This section provides an overview of the datasets utilized in this study, outlines the annotation process employed to generate ground truth labels, and introduces the forest fire image segmentation model along with a detailed explanation of its architecture.

2.1. Dataset

UAV photography provides a comprehensive understanding of the structure of forest terrain and enables the accurate identification of fire locations. Two datasets were selected as data sources to make the experiment more convincing:

(1) FLAME dataset

The Fire Luminosity Airborne-based Machine learning Evaluation (FLAME) dataset is an aerial photography-based forest fire image that was made public by scholars at Northern Arizona University and others in 2020. The FLAME dataset is formed by UAV photography, thus forming a large image perspective with small fire points, which is conducive to the study of forest fire image segmentation.

(2) Chongli dataset

This dataset was acquired from Chongli ($40^{\circ}47' - 41^{\circ}17'$ N, $114^{\circ}17' - 115^{\circ}34'$ E), Hebei Province, China. The forest fire was captured using a Longitude M300 RTK UAV equipped with an H20T gimbal camera. From the video footage, a total of 200 frames were sampled, resulting in 200 forest fire images.

Figures 1 and 2 provide representative samples of these forest fire images.



Figure 1. Frame samples of the normal spectrum palette.

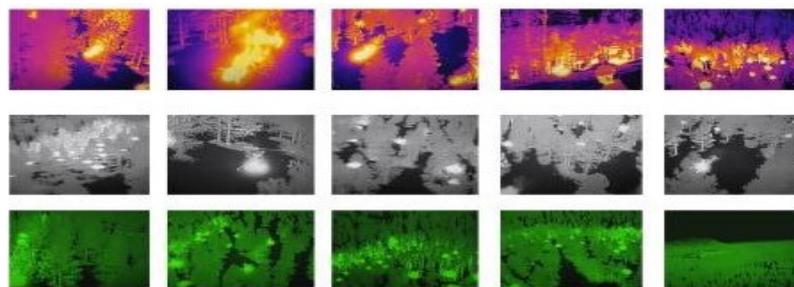


Figure 2. Frame samples of thermal images of Fusion, WhiteHot, and GreenHot palettes from top row to the bottom row.

2.2. Method

The SCGCN framework proposed is illustrated in Figure 3. The process begins with the input image undergoing SLIC processing, which generates multiple superpixel blocks that serve as nodes in the graph structure. Afterwards, a CNN is employed for feature extraction, extracting features from each superpixel block and generating a feature matrix for each node. The graph structure, represented as a sparse weighted graph, is then constructed by determining the weights of each superpixel block and establishing spatial adjacency relationships. The resulting data, including the adjacency matrix and node feature matrix, serve as inputs for the graph convolutional neural network (GCN). The nodes representing the superpixel blocks are trained to obtain their respective categories, and the image segmentation results are achieved by establishing correspondence between the category labels and the corresponding superpixel blocks.

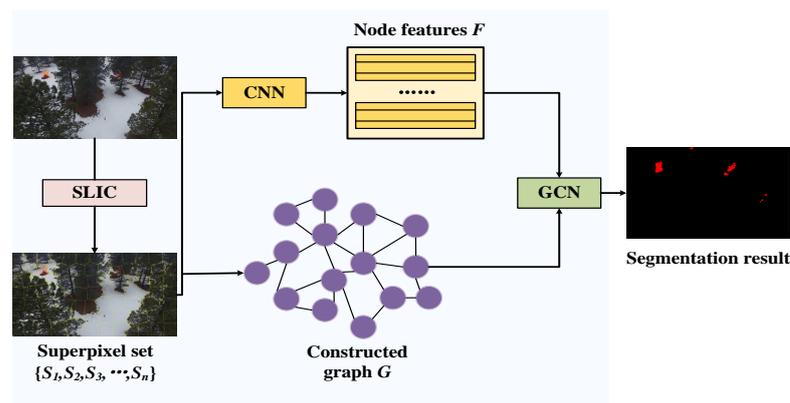


Figure 3. Overview of the proposed SCGCN framework.

2.2.1. Graph Construction

Prior to constructing the graph, it is essential to preprocess the image using superpixel segmentation. According to the SLIC proposed by [21], which is an unsupervised image segmentation method using k-means for local clustering of image pixels [28,29], superpixels offer a highly effective method that partitions the image into blocks of similar characteristics, including position, color, and texture. This approach facilitates the extraction of edge information from objects within the image.

After performing superpixel segmentation, it is necessary to construct a graph based on the superpixel nodes to facilitate training and learning processes. We define the graph structure as an undirected, weighted, sparse $G(V, E)$, where V denotes graph nodes and E stands for edges. Firstly, the image is partitioned into superpixel blocks S_i ($i = 1, \dots, v$) based on the SLIC algorithm, where v represents the number of superpixels. Then, we take the centroid of each superpixel as a graph node V_i ($i = 1, \dots, v$), that is,

$$V = [V_1, V_2, \dots, V_i] = \left[\frac{1}{N_1} \sum_{j=1}^{N_1} x_j^1, \frac{1}{N_2} \sum_{j=1}^{N_2} x_j^2, \dots, \frac{1}{N_i} \sum_{j=1}^{N_i} x_j^i \right] \quad (1)$$

where x_j^i represents the j th pixel in S_i and N_i is the number of pixels in S_i . And for each superpixel block S_i , we calculate the distance between its neighboring superpixel blocks and only construct edges between the nodes with the smallest distance. The steps are as follows:

- (1) Find the Euclidean distance between the node S_j adjacent to S_i , which are the points adjacent to the superpixel block.
- (2) Calculate the weight W between S_i and S_j .

$$W_{ij} = \sqrt{(r_j - r_i)^2 + (g_j - g_i)^2 + (b_j - b_i)^2} \quad (2)$$

- (3) Construct the edge E with the smallest weight, where r, g, b represent the color values of each pixel in the three channels of color space, respectively.

Figure 4 illustrates the steps involved in constructing a graph structure using the SLIC algorithm. Initially, we perform superpixel segmentation with the simple linear iterative clustering (SLIC) algorithm to obtain a series of superpixel blocks. The SLIC algorithm segments by optimizing the similarity between each pixel within a superpixel and the center of the superpixel, effectively reducing the complexity of the image while preserving boundary details. Subsequently, we consider the centroid of each superpixel block as a node in the graph. The calculation of the centroid takes into account the positions of all pixels within the block, thus representing the geometric center of the superpixel block, which provides an accurate reference point for subsequent graph construction. Equation (2) is employed to calculate the adjacency matrix, representing the connections between nodes

in the graph, for constructing the graph structure. In this matrix, the value of each element indicates the strength of the relationship between corresponding node pairs, such as the weight of an edge or the presence or absence of a connection. Utilizing the efficiency of the SLIC algorithm and the expressive power of the adjacency matrix, we are able to effectively capture key features of the image and perform information extraction and analysis at a higher level.

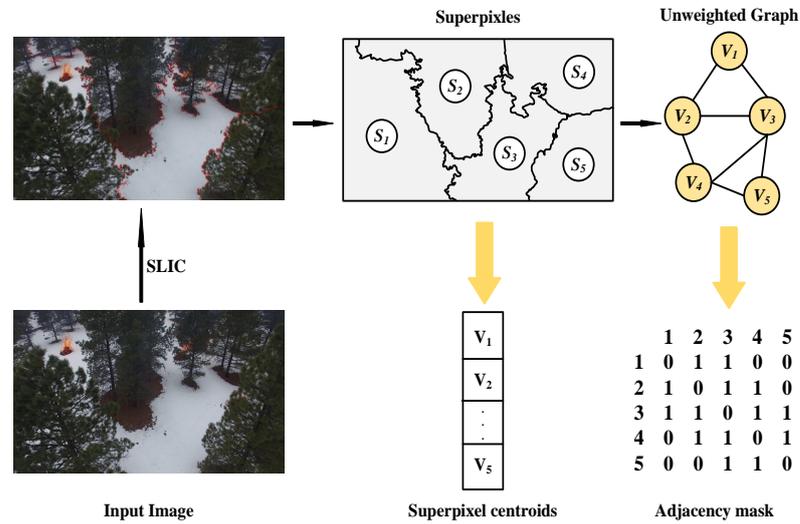


Figure 4. Similarity graph construction of image superpixel area.

2.2.2. Node Classification with GCN

The current definition of graph convolution can be divided into two categories: spectral-based graph convolution and spatial domain-based graph convolution. Spectral-based graph convolution, which maps nodes to the frequency domain space using Fourier transform, realizes convolution on the time domain by performing multiplication in the frequency domain. The features are subsequently mapped back to the time domain space. Spatial domain-based graph convolution, which is similar to our classic CNN, has at its core the aggregation of the information of neighboring nodes [30].

In general, graph convolution includes three steps. First, each node passes feature information to neighboring nodes to extract node feature information; the second step merges local structure and feature information; and the final step collects the previous information for nonlinear transformation to enhance the model capability. The structural architecture of the graph convolutional neural network is shown in Figure 5.

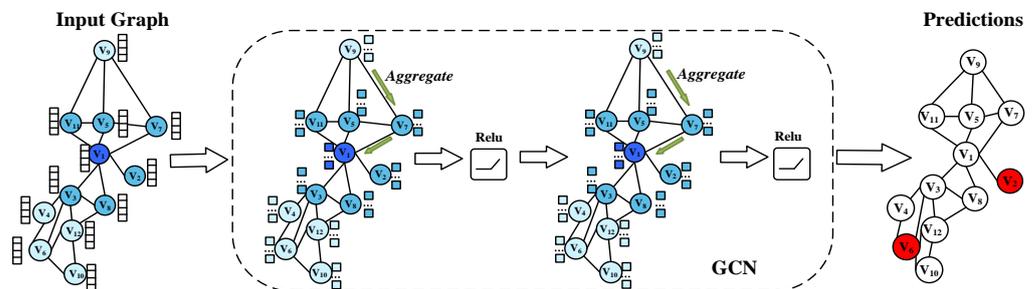


Figure 5. The structure of GCN.

To improve the generalization of the model to new nodes and to learn more spatial scales, we attempt to replace the GCN model’s graph convolution operator with the GraphSAGE convolution operator, which samples subgraphs and aggregates node information. Compared with other node classifications based on individual graphs, our method constructs a set of batch images for multiple images, and based on this, different target nodes are selected while ensuring inductivity. Furthermore, the number of sampled neighbor

nodes in each network layer is fixed and does not include all of them. As a result, the model's generality is increased while the partial edges between nodes are reduced. The constructed GCN structure consists of two convolutional layers activated by the ReLU function, which are computed as follows:

$$H^{l+1} = \sigma(*G \cdot H^{(l)}) \quad (3)$$

where σ represents the Relu activation function, $H^{(l)}$ is the input matrix of the graph convolution layer, $H^0 = F$, F is the node feature matrix, and $*G$ is the graph convolution operator. This differs from the traditional GCN in that the method of graph convolution in the convolution layer of the computational neural network has changed. Thus, the propagation between each layer is shown as follows. And because of the shortage and randomness of the samples collected by the traditional GraphSAGE model, we address the insufficient sampling points of image boundaries and the feature errors caused by sampling with put-back in the traditional GraphSAGE algorithm by sampling the k -order neighbor nodes of the target node v for each iteration and sampling the target node according to $k = 1, \dots, K$ in order to reach the aggregation process according to $k = K, \dots, 1$ for the aggregation.

$$h_v^k = \sigma\left(W \cdot MEAN\left(\left\{h_v^{k-1}\right\} \cup \left\{h_u^{k-1}, \forall u \in N(v)\right\}\right)\right) \quad (4)$$

$$h_v^k = \frac{h_v^k}{\|h_v^k\|_2}, \forall v \in V \text{ (Normalization of features)} \quad (5)$$

W is the trainable weight matrix or parameter matrix, $N : v = 2^V$ is the neighborhood function, and h_v^k is the feature vector of node v in layer k . This graph convolution operator connects the previous layer of the node h_v^{k-1} with the aggregated neighborhood vector $h_{N(v)}^k$ [31]. Finally, there is a softmax layer which is used to convert deep object features into the final classification output.

Algorithm 1 gives the pseudo-code of the proposed method and describes it in detail.

Algorithm 1: Training SCGCN for Image Segmentation

Input: the forest fire image dataset D .

1. Segment images from D by SLIC.
2. Use CNN to extract features $F = \{\vec{f}_1, \vec{f}_2, \dots, \vec{f}_v\}$.
3. Construct graph nodes V . Regions segmented by SLIC are used as graph nodes $V = \{v_1, v_2, \dots, v_n\}$.
4. Construct graph edges E . Take the first order adjacency relationship of a graph node with the smallest weight as the edge of the graph.
5. Classify the graph nodes when the GCN training ends.
6. Assign the class of each node to the superpixel of this node.

Output: the semantic segmentation.

2.2.3. Loss Function

In this paper, we use a new loss function that aims to solve the problems arising from the use of superpixels to generate graph structures and the use of GCN for graph node classification [32,33]. The first is that, when utilizing SLIC for superpixel segmentation, the number of pixel points included in each superpixel block is variable. Therefore, the information of the pixel points carried by each superpixel block is not equal. Secondly, there exists an imbalance in the classes of nodes in the graph, where samples from certain

classes (e.g., small target fire spots vs. background) may be overlooked, leading to a loss in performance. Given the aforementioned issues, the cross-entropy (CE) loss is defined as

$$L_i = -b_i q_i \log \frac{\exp(p_{i,q_i})}{\sum_{j=1}^J \exp(p_{i,j})} \quad (6)$$

where p is the ground truth, q is the target, b is a weight value that balances the instability of the classification, J is the number of class, and b is defined as

$$b_i = \frac{N - n_i}{N} \quad (7)$$

where N is the total number of pixel points and n is the number of pixel points in each class. After calculating the loss of CE for each graph node to achieve class equilibrium, we apply the weights of the superpixels to the loss

$$SL_i = S_i \cdot L_i \quad (8)$$

where SL_i is the superpixel loss in each node i . S_i is the superpixel weight in each node i , which is calculated as follows:

$$S_i = -\frac{1 + \epsilon}{\log y_i + \epsilon} \quad (9)$$

where y_i is the normalized number of pixels in the superpixel generated in each node i . ϵ is set at 10^{-5} to avoid zero division error. Finally, loss is calculated by the following equation.

$$SL = \frac{1}{N} ([L_1, \dots, L_N]^T [S_1, \dots, S_N]) \quad (10)$$

2.2.4. Evaluation Metrics

We introduced accuracy (Acc), F_1 -score [34], and MIoU to assess the segmentation results quantitatively. The fraction of properly assigned pixels among all ground truth pixels is represented by the recall, whereas the proportion of correctly predicted pixels is represented by precision. The F_1 -score is a weighted harmonic of precision and recall. A higher F_1 score indicates better performance of the algorithm.

$$F_1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (11)$$

The accuracy (Acc) and intersection (IOU) are given by the following two equations, respectively.

$$\text{Acc} = \frac{TP + TN}{TP + TN + FP + FN} \quad (12)$$

$$\text{IOU} = \frac{TP}{TP + FN + FP} \quad (13)$$

TP represents the number of images labeled as fire and classified as fire, FP indicates the number of images labeled as fire and classified as no fire, FN denotes the number of images labeled as no fire but classified as fire, and TN stands for the number of images labeled as no fire and classified as no fire.

Each class's intersection-over-union (IoU) is utilized as an assessment measure. IoU, also known as Jaccard's index, is a statistic that indicates sample similarity and diversity. IoU, which measures the spatial overlap between the predicted results and the ground truth in image segmentation, is the ratio of the intersection of pixel-wise classification results and ground truth. In order to measure the model fairly, we utilize the mean value of IoUs, which is indicated as

$$mIoU = \frac{1}{N} \sum_{i=1}^N \frac{P_i \cap G_i}{P_i \cup G_i} \quad (14)$$

where P_i and G_i refer to the predicted result and corresponding ground truth label for the i -th image, respectively.

2.2.5. Implementation Details

In this experiment, superpixel blocks of each image are generated using SLIC. Because the shape of the superpixels is irregular, we extract all of the superpixels from each image to save them as images individually. GCN is applied to segment forest fire images. The GCN model uses the edges around the nodes to train node characteristics; the process is the same as standard semantic segmentation except a graph network is used in place of the encoder–decoder structure.

For our model, GraphSAGE is used as the convolution filter. GraphSAGE consists of two graph convolution layers, an initial learning rate of 0.001, a batch size of four images, an Adam optimizer with an attenuation of 0.0001, two GCN layers, and 300 simulated training sessions. All experiments were performed on the Pytorch framework on NVIDIA GeForce RTX 3090 GPU.

In this paper, we use the CNN for deep feature extraction of superpixel blocks. Because of the irregular shape of the superpixel blocks, it is difficult to extract their features directly, so we extract the superpixel blocks of each image separately, set their size to 48×48 , and set the blank area pixel value to 0.

We use our CNN to extract the features of superpixels. The CNN network consists of five convolutional layers, three pooling layers, and one fully connected layer, where the fully connected layer converts the feature matrix into a one-dimensional feature vector. In the deep network, the image input size is 48×48 , the normalization layer (BN) and rectified linear unit (ReLU) activation function are used to improve the efficiency, and the maximum pooling layer is added after layers 1, 2, and 5 of the network to reduce the data dimensionality and improve the robustness of feature extraction. A 5×5 convolutional filter is used in layer 1, layer 2, and layer 3, a 3×3 convolution filter is used in layer 4, and a 2×2 convolution filter is used in layer 5. Finally, a 50-dimensional pixel feature is obtained as the feature of the superpixel block through a fully connected layer.

3. Experimental Results

In this section, we will introduce the experimental results, and analysis will follow.

3.1. Results of FLAME Dataset

To observe the segmentation effect of the dataset used in this paper under different models, we compared four existing methods with our SCGCN model, including Deeplabv3+ [35], Unet++ [36], HRnet [37], and PSPnet [38]. It is worth noting that, to ensure comparability of predictions, we maintained identical experimental configurations for training all models and ensured consistency in other parameters.

To validate our model, SCGCN was compared with three state-of-the-art deep learning methods: Deeplabv3+ [35], Unet++ [36], HRnet [37], and PSPnet [38]. The experimental results obtained from this dataset are shown in Table 1, where the F_1 score was chosen as the evaluation metric in order to directly reflect the segmentation effect. The experimental results show that our method not only outperforms other algorithms in terms of F_1 scores but also performs best in terms of MIoU and Acc.

Table 1. Comparison results between different segmentation models.

Method	MIoU (%)	Acc (%)	F_1 (%)
PSPnet [38]	34.65	47.56	74.32
Deeplabv3+ [35]	69.40	81.85	82.02
Unet++ [36]	79.52	86.68	90.60
HRnet [37]	77.71	85.70	88.28
SCGCN (ours)	79.87	87.53	91.69

To facilitate a visual comparison of the segmentation performance between our model and the four other models, we have carefully chosen four images from the test dataset to illustrate the prediction results, as depicted in Figure 6. Upon examining the figure, it becomes evident that Unet++ [36] and our SCGCN model exhibit superior segmentation effects, accurately delineating the shape of forest fires. In contrast, PSPnet [38] displays the poorest performance, only approximating the shape of the forest fires. Furthermore, our model demonstrates better segmentation recognition for small target fire points, as indicated by the blue boxes in the figure. However, recognition errors remain significant for heavily obscured fires.

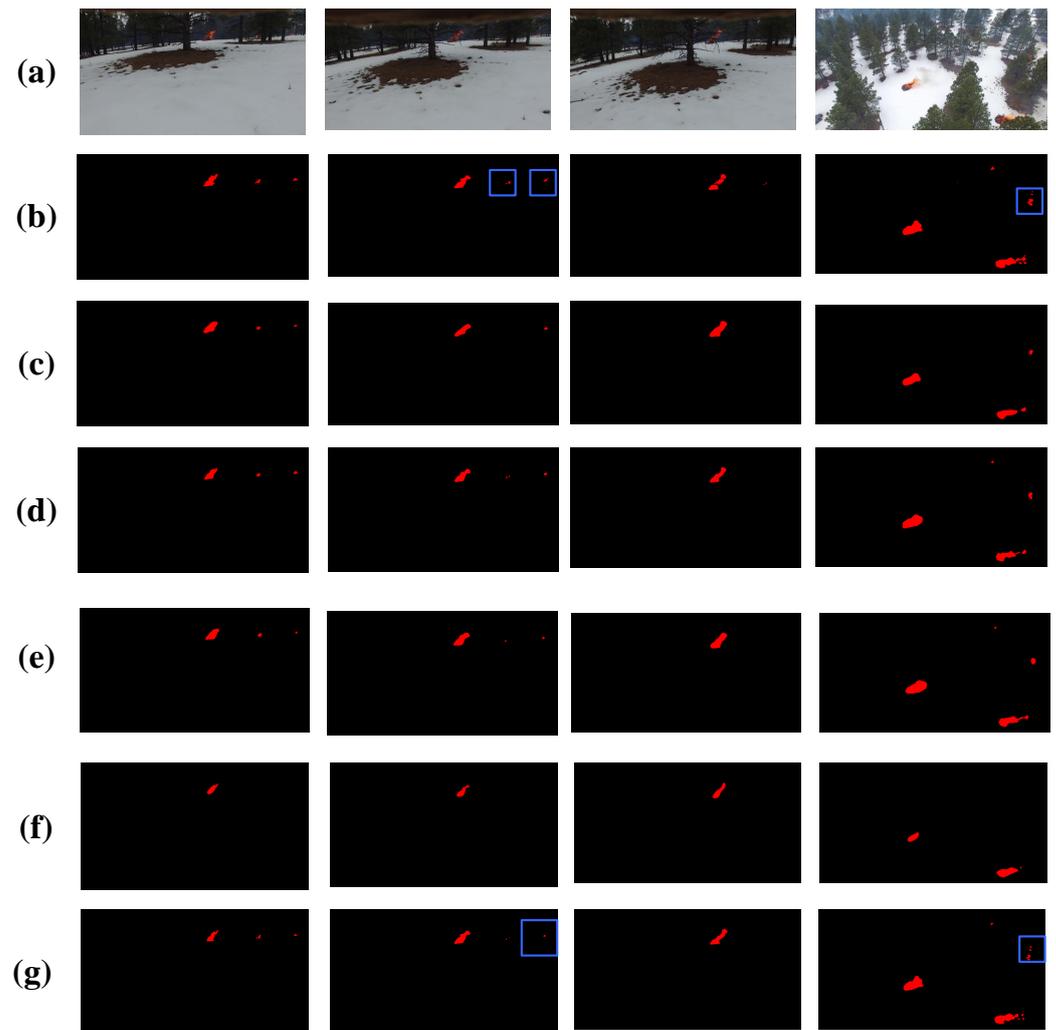


Figure 6. Results from testing images. (a) Original images, (b) ground truth, (c) DeepLabv3+, (d) Unet++, (e) HRnet, (f) PSPnet, (g) SCGCN.

3.2. Results of Chongli Dataset

We conducted experiments on the Chongli dataset to evaluate the performance of our technique, comparing it with existing segmentation models such as DeepLabv3+ [35], Unet++ [36], HRnet [37], and PSPnet [38]. Several representative images from the test dataset were selected, and the corresponding prediction results are presented in Figure 7.

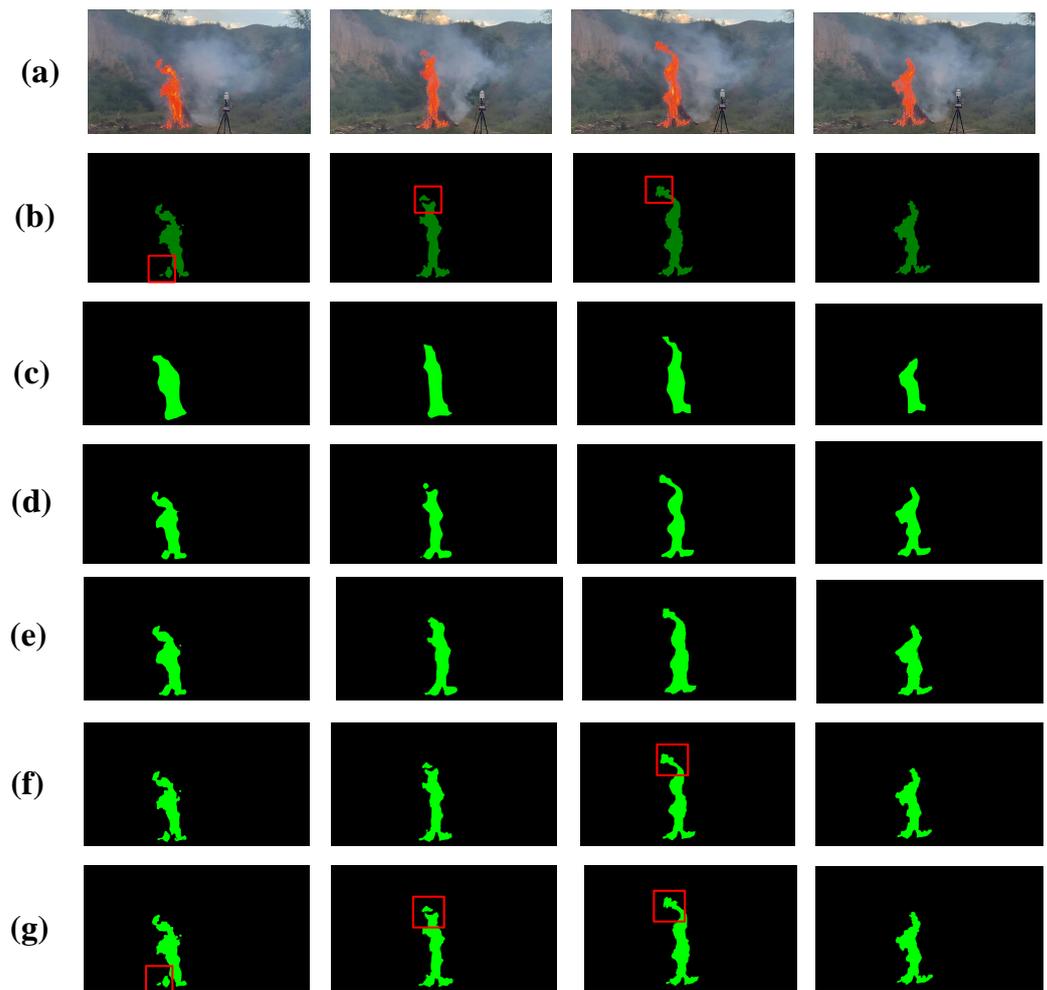


Figure 7. Results from testing images. (a) Original images, (b) ground truth, (c) PSPnet, (d) HRnet, (e) Deeplabv3+, (f) Unet++, (g) SCGCN.

By observing Figure 7, it becomes evident that PSPnet [38] only approximates the shape of the fire and lacks accurate segmentation. Conversely, Unet++ [36], HRnet [37], Deeplabv3+ [35], and our SCGCN can segment the fire more accurately, but Deeplabv3+ [35] has the phenomenon of under-segmentation, and while Unet++ [36] achieves a better segmentation effect, the model's fire shape lacks detail compared to our SCGCN model, as shown by the red boxes in the figure.

Table 2 demonstrates the segmentation results of Deeplabv3+ [35], Unet++ [36], HRnet [37], PSPnet [38], and SCGCN on the Chongli dataset. SCGCN clearly attained the best performance outcomes. Using the F_1 score as an example, SCGCN scored at 97.56%, which is higher than PSPNet (79.12%), Deeplabv3+ (91.72%), HRnet (92.31%), and Unet++ (96.09%), demonstrating that our model is suitable for forest fire segmentation.

Table 2. Comparison results between different segmentation models.

Method	MIoU (%)	Acc (%)	F_1 (%)
PSPnet [38]	56.0	65.71	79.12
Deeplabv3+ [35]	83.65	90.48	91.72
Unet++ [36]	91.50	95.04	96.09
HRnet [37]	88.54	93.65	92.31
SCGCN (ours)	92.34	96.69	97.56

3.3. Superpixel Number

Superpixels are built from a succession of hyperpixel blocks made up of pixel points that are next to each other and share characteristics such as color, brightness, texture, and others [39]. These superpixel blocks mainly preserve relevant information for image segmentation and generally do not damage object boundary information in the image. The resolution of the segmentation result is determined by the size of the superpixels. If the superpixel is too big, the segmentation result will lose a significant amount of edge information. On the contrary, superpixels that are too small will introduce more noise in the result and increase the computational burden of data segmentation [40]. With different numbers of superpixels, the features extracted by the nodes are different, and the image features contained in the whole picture are also different. In general, for image data, the greater the number of superpixels and the greater the number of nodes, the richer the information contained in the graph and the closer the features of the graph are to the features of the original image. Figures 8 and 9 depict the effect of superpixelation on the FLAME and Chongli datasets, where the number of superpixels from top to bottom is 50, 100, and 150, respectively. The visual characteristics are equally identical to those of the superpixelated picture with 150 superpixels.

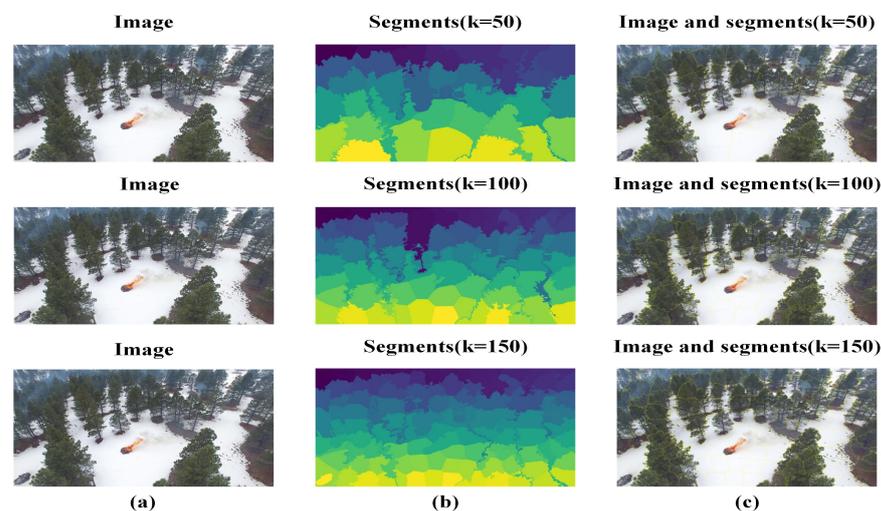


Figure 8. The segmentation by SLIC of FLAME dataset. (a) Examples of the original images. (b,c) The superpixel representation for FLAME dataset; K is the number of superpixels (nodes in our graphs).

Figure 10 illustrates the F_1 score variation with different numbers of superpixels on both the FLAME dataset and the Chongli dataset. The horizontal coordinate is the number of superpixels contained in a single test plot, the FLAME dataset ranges from 1000 to 2600, and the size of the superpixels decreases as the number increases. From the figure, we can observe that the overall accuracy shows an increasing trend from 1000 to 2000 and reaches a maximum value at 2000. Therefore, each image of our FLAME dataset was segmented into about 2000 superpixels. Since the image of the Chongli dataset was 1920*1080, the number of pixels in a superpixel ranged between 100 and 1000. The overall trend is up from 100 to 800, and the highest value is at 800. Therefore, each image of our Chongli dataset was segmented into about 800 superpixels. We also observed that the model performance tends to stabilize after surpassing a certain number of superpixels. This indicates a point of diminishing returns, where additional superpixels no longer contribute significantly to performance improvement. Our current hypothesis is that the complexity introduced by an excess of superpixels may not necessarily translate into better feature representation for the model. Additionally, the increase in computational costs and memory requirements with more superpixels may not be feasible for practical applications. While there may theoretically exist a possibility of achieving another peak in performance with further increases in the number of superpixels, our observations and current understanding of the

model suggest that the likelihood of achieving significant improvements is minimal. The model appears to be approaching an asymptote of performance, indicating that our choice is nearing an optimal balance between superpixel resolution and model capability.

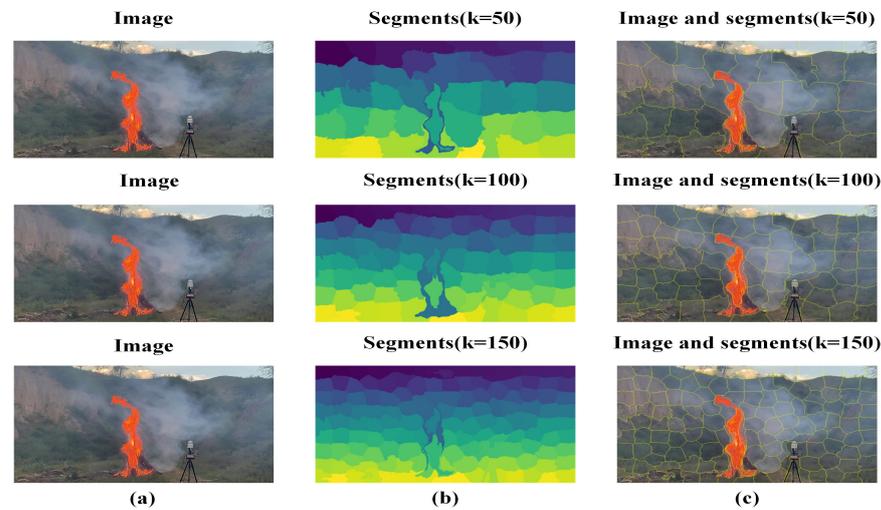


Figure 9. The segmentation by SLIC of Chongli dataset. (a) Examples of the original images. (b,c) The superpixel representation for Chongli dataset; K is the number of superpixels (nodes in our graphs).

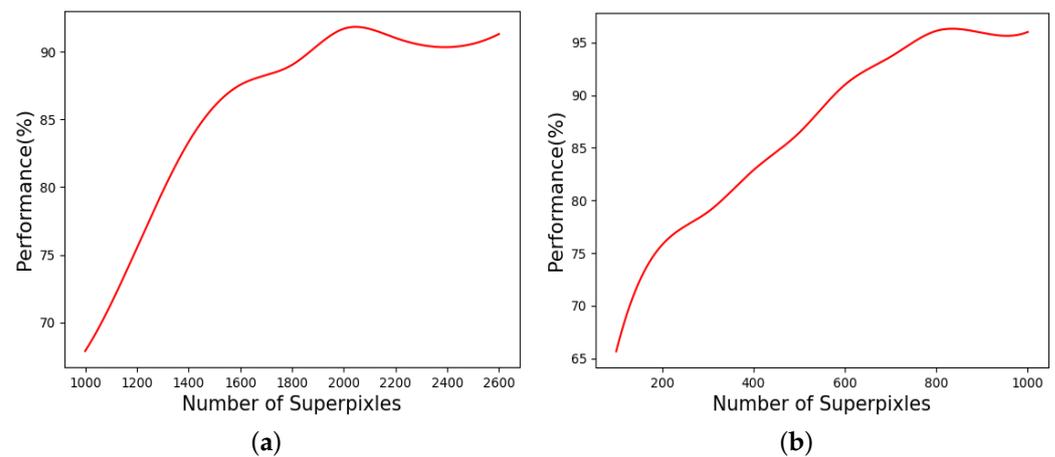


Figure 10. Performance comparisons of different superpixel numbers when evaluating with FLAME and Chongli datasets. (a) FLAME dataset; (b) Chongli dataset.

3.4. Ablation Experiment

GCN, as an advanced image segmentation model, has achieved good results in the classification and recognition of forest fires. However, there is still room for improvement in its performance. In this article, we improved the GCN model and conducted a series of ablation experiments to verify the effectiveness and rationality of our model. Table 3 shows the ablation studies on the FLAME dataset. Specifically, we compared the original GCN with the traditional loss function model after combining superpixels. From Table 3, we observed that when we replaced the convolution operator in GCN with the GraphSAGE convolution operator, the MIoU value of the model increased by 0.98%. When we replaced the loss function with the SL new loss function proposed in this paper, the MIoU value of the model increased by 3.36%, and our model achieved the best overall effect. Table 4 shows the ablation studies on the Chongli dataset. When compared in the same way, when we replaced the convolution operator in GCN with the GraphSAGE convolution operator, the MIoU value of the model increased by 3.14%. When we replaced the loss function with

the new SL loss function proposed in this paper, the MIoU value of the model increased by 4.69%. Overall, our model also achieved the best performance.

Table 3. Ablation study on FLAME dataset.

Method	MIoU (%)	Acc (%)	F ₁ (%)
GCN + SLIC + CE	76.51	83.46	86.23
GraphSAGE + SLIC + CE	77.49	86.23	89.61
GraphSAGE + SLIC + SL (ours)	79.87	87.53	91.69

Table 4. Ablation study on Chongli dataset.

Method	MIoU (%)	Acc (%)	F ₁ (%)
GCN + SLIC + CE	87.65	91.26	93.58
GraphSAGE + SLIC + CE	90.79	95.87	95.70
GraphSAGE + SLIC + SL (ours)	92.34	96.69	97.56

4. Discussion

Forest fires are dynamic objects with shapes that constantly change and textures that are difficult to accurately depict [41]. Traditional methods have advantages in forest fire recognition in terms of speed and accuracy. For instance, Zheng et al. [42] proposed an improved dynamic convolutional neural network (DCNN) model based on a traditional DCNN model that can accurately identify forest fires. However, in complex situations, such as capturing small fires from the perspective of a drone, traditional methods often face challenges. Small target fires and irregular fire shapes make it difficult for traditional methods to accurately segment fires, and edge processing is not detailed enough.

Cao et al. [43] proposed a segmentation detection algorithm called YOLO-SF. This algorithm combines instance segmentation technology with the YOLOv7-Tiny object detection algorithm, achieving high accuracy in detecting and segmenting large target fire images. However, this method has certain limitations in handling small target fires and capturing boundaries accurately. Therefore, based on these issues, this study proposes a new method, namely the SCGCN model, based on superpixels and graph convolutional networks, for forest fire image segmentation. The SCGCN model fully utilizes the information of graph structure and employs the SLIC method as a preprocessing technique to enhance the segmentation of forest fire edges. Additionally, we replace the graph convolution operator in the GCN with GraphSAGE to integrate additional spatial scale information and enhance the model's generalization capabilities for new nodes. Compared to traditional methods, our approach performs better at segmenting small target fires and capturing boundaries.

The experimental results demonstrate that using the SCGCN model for forest fire image segmentation has significant advantages and potential value in practical applications. This finding provides new methods and tools for forest fire monitoring and management. Future research can further explore how to optimize the model to adapt to different types of forest fires and environmental conditions.

5. Conclusions

We propose a novel approach, the superpixel-based graph convolutional network (SCGCN), to tackle the challenge of forest fire segmentation. Our contributions are as follows: (1) Utilizing the SLIC algorithm, we preprocess the image to convert it into a graph structure, enhancing segmentation efficiency and accurately capturing forest fire edges. (2) We introduce a new loss function to tackle the challenges posed by class and superpixel size imbalances. (3) We replace the graph convolution operator with the GraphSAGE operator, enabling the extraction of more subgraphs from smaller classes and improving model generalization by mitigating biased edges. The effectiveness of the SCGCN model has been validated on two datasets. Future research will focus on the

model's ability to identify objects in satellite remote sensing images, as well as integrating satellite multimodal data, such as data from different sensors or multispectral images, to support forest fire segmentation.

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

Funding: This research was funded by Jiangsu Province Graduate Research Innovation Program (KYCX23_1207).

Data Availability Statement: This study utilized two datasets: the FLAME dataset and the chongli dataset. The FLAME dataset is publicly available and can be accessed through the IEEE Dataport website (<https://iee-dataport.org/open-access/flare-dataset-aerial-imagery-pile-burn-detection-using-drones-uavs> (accessed on 16 January 2024)). On the other hand, the chongli dataset is currently not available for public access due to privacy constraints.

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

References

- Hu, Y.; Zhan, J.; Zhou, G.; Chen, A.; Cai, W.; Guo, K.; Hu, Y.; Li, L. Fast forest fire smoke detection using mvmmnet. *Knowl.-Based Syst.* **2022**, *241*, 108219. [CrossRef]
- Ghali, R.; Akhloufi, M.A.; Jmal, M.; Souidene Mseddi, W.; Attia, R. Wildfire segmentation using deep vision transformers. *Remote Sens.* **2021**, *13*, 3527. [CrossRef]
- Zhu, Y.; Li, D.; Fan, J.; Zhang, H.; Eichhorn, M.P.; Wang, X.; Yun, T. A reinterpretation of the gap fraction of tree crowns from the perspectives of computer graphics and porous media theory. *Front. Plant Sci.* **2023**, *14*, 1109443. [CrossRef] [PubMed]
- Xu, Y.; Sun, Y.; Zhang, F.; Jiang, H. Modeling Fire Boundary Formation Based on Machine Learning in Liangshan, China. *Forests* **2023**, *14*, 1458. [CrossRef]
- Gao, D.; Wang, H.; Guo, X.; Gui, G.; Wang, W.; Yin, Z.; Wang, S.; Liu, Y.; He, T. Federated learning based on CTC for heterogeneous internet of things. *IEEE Internet Things J.* **2023**, *10*, 22673–22685. [CrossRef]
- Xue, X.; Jin, S.; An, F.; Zhang, H.; Fan, J.; Eichhorn, M.P.; Jin, C.; Chen, B.; Jiang, L.; Yun, T. Shortwave radiation calculation for forest plots using airborne LiDAR data and computer graphics. *Plant Phenom.* **2022**, *2022*, 9856739. [CrossRef]
- Gao, D.; Wang, L.; Hu, B. Spectrum efficient communication for heterogeneous IoT networks. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3945–3955. [CrossRef]
- Shi, C.; Zhang, F. A forest fire susceptibility modeling approach based on integration machine learning algorithm. *Forests* **2023**, *14*, 1506. [CrossRef]
- Zhao, Y.; Komachi, M.; Kajiwara, T.; Chu, C. Region-attentive multimodal neural machine translation. *Neurocomputing* **2022**, *476*, 1–13. [CrossRef]
- Wang, S.; Zhao, J.; Ta, N.; Zhao, X.; Xiao, M.; Wei, H. A real-time deep learning forest fire monitoring algorithm based on an improved pruned+ kd model. *J. Real-Time Image Process.* **2021**, *18*, 2319–2329. [CrossRef]
- Khryashchev, V.; Larionov, R. Wildfire segmentation on satellite images using deep learning. In Proceedings of the 2020 Moscow Workshop on Electronic and Networking Technologies (Mwent), Moscow, Russia, 11–13 March 2020; pp. 1–8.
- Hori, T.; Watanabe, S.; Zhang, Y.; Chan, W. Advances in joint ctc-attention based end-to-end speech recognition with a deep cnn encoder and rnn-lm. *arXiv* **2017**, arXiv:1706.02737.
- Meng, Y.; Wei, M.; Gao, D.; Zhao, Y.; Yang, X.; Huang, X.; Zheng, Y. Cnn-gcn aggregation enabled boundary regression for biomedical image segmentation. In *Medical Image Computing and Computer Assisted Intervention—MICCAI 2020, Proceedings of the 23rd International Conference, Lima, Peru, 4–8 October 2020; Part IV 23*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 352–362.
- Scarselli, F.; Gori, M.; Tsoi, A.C.; Hagenbuchner, M.; Monfardini, G. The graph neural network model. *IEEE Trans. Neural Netw.* **2008**, *20*, 61–80. [CrossRef] [PubMed]
- Gallicchio, C.; Micheli, A. Graph echo state networks. In Proceedings of the 2010 International Joint Conference on Neural Networks (IJCNN), Barcelona, Spain, 18–23 July 2010; pp. 1–8.
- Liu, Q.; Xiao, L.; Yang, J.; Wei, Z. Cnn-enhanced graph convolutional network with pixel-and superpixel-level feature fusion for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 8657–8671. [CrossRef]
- Wang, L.; Li, R.; Zhang, C.; Fang, S.; Duan, C.; Meng, X.; Atkinson, P.M. Unetformer: A unet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery. *ISPRS J. Photogramm. Remote Sens.* **2022**, *190*, 196–214. [CrossRef]
- Wu, D.; Zhang, C.; Ji, L.; Ran, R.; Wu, H.; Xu, Y. Forest fire recognition based on feature extraction from multi-view images. *Trait. Du Signal* **2021**, *38*, 775–783. [CrossRef]
- Bruna, J.; Zaremba, W.; Szlam, A.; LeCun, Y. Spectral networks and locally connected networks on graphs. *arXiv* **2013**, arXiv:1312.6203.

20. Dong, X.; Zhang, C.; Fang, L.; Yan, Y. A deep learning based framework for remote sensing image ground object segmentation. *Appl. Soft Comput.* **2022**, *130*, 109695. [[CrossRef](#)]
21. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [[CrossRef](#)] [[PubMed](#)]
22. Bergh, M.V.d.; Boix, X.; Roig, G.; Capitani, B.d.; Gool, L.V. Seeds: Superpixels extracted via energy-driven sampling. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 13–26.
23. Ban, Z.; Liu, J.; Cao, L. Superpixel segmentation using gaussian mixture model. *IEEE Trans. Image Process.* **2018**, *27*, 4105–4117. [[CrossRef](#)]
24. Belizario, I.V.; Linares, O.C.; Neto, J.d.E.S.B. Automatic image segmentation based on label propagation. *IET Image Proc.* **2021**, *15*, 2532–2547. [[CrossRef](#)]
25. Xiong, D.; Yan, L. Early smoke detection of forest fires based on svm image segmentation. *J. For. Sci.* **2019**, *65*, 150–159. [[CrossRef](#)]
26. Martins, J.; Junior, J.M.; Menezes, G.; Pistori, H.; Sant, D.; Gonçalves, W. Image segmentation and classification with slic superpixel and convolutional neural network in forest context. In *Proceedings of the IGARSS 2019—2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019*; pp. 6543–6546.
27. Tian, Z.; Li, X.; Zheng, Y.; Chen, Z.; Shi, Z.; Liu, L.; Fei, B. Graph-convolutional-network-based interactive prostate segmentation in mr images. *Med. Phys.* **2020**, *47*, 4164–4176. [[CrossRef](#)] [[PubMed](#)]
28. Li, Y.; Chen, R.; Zhang, Y.; Li, H. A cnn-gcn framework for multi-label aerial image scene classification. In *Proceedings of the IGARSS 2020–2020 IEEE International Geoscience and Remote Sensing Symposium, Virtual, 26 September–2 October 2020*; pp. 1353–1356.
29. Zhao, C.; Qin, B.; Feng, S.; Zhu, W. Multiple superpixel graphs learning based on adaptive multiscale segmentation for hyperspectral image classification. *Remote Sens.* **2022**, *14*, 681. [[CrossRef](#)]
30. Peng, Z.; Liu, H.; Jia, Y.; Hou, J. Attention-driven graph clustering network. In *Proceedings of the 29th ACM International Conference on Multimedia, Virtual, 20–24 October 2021*; pp. 935–943.
31. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
32. Filtjens, B.; Vanrumste, B.; Slaets, P. Skeleton-based action segmentation with multi-stage spatial-temporal graph convolutional neural networks. *IEEE Trans. Emerg. Top. Comput.* **2022**, *12*, 202–212. . 10.1109/TETC.2022.3230912. [[CrossRef](#)]
33. Avelar, P.H.; Tavares, A.R.; da Silveira, T.L.; Jung, C.R.; Lamb, L.C. Superpixel image classification with graph attention networks. In *Proceedings of the 2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Porto de Galinhas, Brazil, 7–10 November 2020*; pp. 203–209.
34. Sasaki, Y. The truth of the f-measure. *Teach. Tutor. Mater.* **2007**, *1*, 1–5.
35. 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 (ECCV), Munich, Germany, 8–14 September 2018*; pp. 801–818.
36. Zhou, Z.; Rahman Siddiquee, M.M.; Tajbakhsh, N.; Liang, J. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 3–11.
37. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019*; pp. 5693–5703.
38. 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. 2881–2890.
39. Wang, Y.; Wei, H.; Ding, X.; Tao, J. Video background/foreground separation model based on non-convex rank approximation rpca and superpixel motion detection. *IEEE Access* **2020**, *8*, 157493–157503. [[CrossRef](#)]
40. Ma, F.; Gao, F.; Sun, J.; Zhou, H.; Hussain, A. Attention graph convolution network for image segmentation in big sar imagery data. *Remote Sens.* **2019**, *11*, 2586. [[CrossRef](#)]
41. Zhang, L.; Li, J.; Zhang, F. An efficient forest fire target detection model based on improved YOLOv5. *Fire* **2023**, *6*, 291. [[CrossRef](#)]
42. Zheng, S.; Gao, P.; Wang, W.; Zou, X. A Highly Accurate Forest Fire Prediction Model Based on an Improved Dynamic Convolutional Neural Network. *Appl. Sci.* **2022**, *12*, 6721. [[CrossRef](#)]
43. Cao, X.; Su, Y.; Geng, X.; Wang, Y. YOLO-SF: YOLO for fire segmentation detection. *IEEE Access* **2023**, *11*, 111079–111092. [[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.