remote sensing

IMPACT FACTOR 5.0

CITESCORE 7.9

Article

Real-Time Detection of Full-Scale Forest Fire Smoke Based on Deep Convolution Neural Network

Xin Zheng, Feng Chen, Liming Lou, Pengle Cheng and Ying Huang

MDPI

https://doi.org/10.3390/rs14030536

remote sensing

MDPI

Article

Real-Time Detection of Full-Scale Forest Fire Smoke Based on Deep Convolution Neural Network

Xin Zheng [1], Feng Chen [2], Liming Lou [1], Pengle Cheng [1,*] and Ying Huang [3]

[1] School of Technology, Beijing Forestry University, Beijing 100083, China; zhengxin5173@bjfu.edu.cn (X.Z.); liminglou@bjfu.edu.cn (L.L.)
[2] School of Nature Conservation, Beijing Forestry University, Beijing 100083, China; chenfeng1208@bjfu.edu.cn
[3] Department of Civil, Construction, and Environmental Engineering, North Dakota State University, Fargo, ND 58102, USA; ying.huang@ndsu.edu
* Correspondence: chengpengle@bjfu.edu.cn

Abstract: To reduce the loss induced by forest fires, it is very important to detect the forest fire smoke in real time so that early and timely warning can be issued. Machine vision and image processing technology is widely used for detecting forest fire smoke. However, most of the traditional image detection algorithms require manual extraction of image features and, thus, are not real-time. This paper evaluates the effectiveness of using the deep convolutional neural network to detect forest fire smoke in real time. Several target detection deep convolutional neural network algorithms evaluated include the EfficientDet (EfficientDet: Scalable and Efficient Object Detection), Faster R-CNN (Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks), YOLOv3 (You Only Look Once V3), and SSD (Single Shot MultiBox Detector) advanced CNN (Convolutional Neural Networks) model. The YOLOv3 showed a detection speed up to 27 FPS, indicating it is a real-time smoke detector. By comparing these algorithms with the current existing forest fire smoke detection algorithms, it can be found that the deep convolutional neural network algorithms result in better smoke detection accuracy. In particular, the EfficientDet algorithm achieves an average detection accuracy of 95.7%, which is the best real-time forest fire smoke detection among the evaluated algorithms.

Keywords: forest fire smoke detection; convolutional neural networks; deep learning; real-time detection

Citation: Zheng, X.; Chen, F.; Lou, L.; Cheng, P.; Huang, Y. Real-Time Detection of Full-Scale Forest Fire Smoke Based on Deep Convolution Neural Network. *Remote Sens.* **2022**, *14*, 536. https://doi.org/10.3390/rs14030536

Academic Editor: Carlos Alberto Silva

Received: 17 October 2021
Accepted: 21 January 2022
Published: 23 January 2022

1. Introduction

Forest fire is one of the natural disasters with high frequency and great harmfulness in the world [1]. It usually spreads quickly and is difficult to control, causing intensive losses of human lives and properties. For example, "Lungs of the Earth" fires in the Amazon rainforest have burned to the ground a total of 4920 square kilometers of rainforest, larger than half a million football fields, which has brought incalculable damage to human beings and the natural environment. To control and mitigate forest fire effectively, early warning of the initiation of forest fire is of particular importance. For early warning of forest fires, compared with fire flames, smoke appears earlier, spreads faster, and has a larger volume, which can be easier to identify visually [2–4]. The fast advances and implementation of field surveillance cameras in forests and enhanced computational capacity can especially reduce the cost of monitoring of forest fires by using machine vision [5]. Doing so provides a potential economic and effective way for early detection of forest fires if efficient and effective machine vision algorithms are available.

In general, there are two categories of smoke detection algorithms including the traditional methods and deep machine learning methods [6–11]. Traditional smoke detection methods are usually based on manually extracted features such as color, texture, shape, and motion. For instance, the color features are usually extracted from different color spaces, such as RGB, HSV, and YCbCr, and the texture features are extracted by fractal analysis [12–17], wavelet decomposition [18–22], Gabor Transform [23–27], and histogram of local gradient direction [28–31] methods. In addition to the single feature extraction, multiple smoke features can also be considered together to improve the robustness of detection algorithm. Han et al. [32] combined Gaussian mixture model and multi-color space to detect forest fire detection method with improved detection accuracy. Recently, Gao et al. [33,34] developed a smoke root extraction strategy in full-scale conditions through the fluid mechanics model, which is effective in early smoke detection, although it still has challenges in leak detection of the candidate connected domain of smoke root in near and far conditions. However, since these traditional algorithms rely on intensive knowledge for artificial feature selection algorithms, they may be highly subjective and complex in operation. In addition, due to the fact that these artificially extracted features vary greatly in different scenes, the detection cannot meet the required accuracy and has poor robustness for wide applications. To date, there still has a great need to develop artificial feature algorithms that can effectively detect complex and changeable field scenes.

In the past decade, wireless communication has made it possible for users to obtain a large amount of remote camera vision data. In addition, the continuous advances in computer computation capacity lower the computation cost, especially with the development of Graphics Processing Unit (GPU), which make it within the reach for applying deep learning neural networks and algorithms for various applications. Therefore, the neural networks have been introduced to develop self-learning algorithms for feature collection of fire images [1–5,35–38]. Based on various CNN models such as AlexNet, VGG, Inception, ResNet, etc., the smoke and flame detection algorithms were also investigated [36,38]. Time series information was introduced into the algorithm [6] to detect the smoke and flame simultaneously through reforming the VGG network [7]. In addition, multi-layer convolutional neural networks were also investigated to detect smoke and fire [39–41].

However, the deep learning algorithms also have their limitations. As shown in Figure 1, most of the existing deep learning algorithms consider fire detection as a classification problem and ignore the region identification process such that the entire image was classified into one category. However, during the early stages of the fire, smoke and flames covered only a small portion of the image and do not show the smoke and flame characteristics to be very obvious. The use of features from the entire image without region recommendations reduces detection accuracy and delays the detection and alarm of fire events.

To address the inaccurate early fire detection, some algorithms were developed to generate suggested regions through artificial selection of features and to classify suggested regions through neural networks. In such algorithms, the suggested regions were generated through separate calculations, and the global fire detection was not conducted using neural networks, resulting in a large amount of computation and slow detection speed.

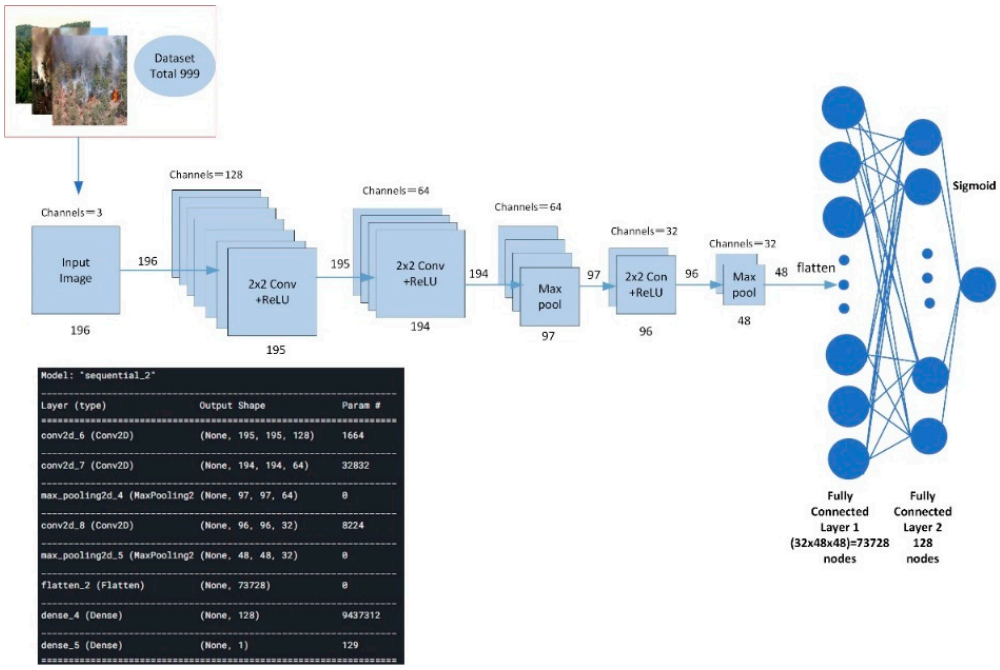


Figure 1. Deep learning convolutional neural network for forest fire smoke classification.

A combination of CNNs with these regions and features extracting algorithms may solve the above-mentioned challenges for early fire detection. Although the color, texture, and shape of smoke are effective methods to distinguish smoke from background, no single artificial feature can cover all scenes. Thus, the CNN algorithms can automatically extract features from a large number of samples followed by other deep learning algorithms for fire detection. As shown in Figure 2, Wang et al. [42] converted RGB into HSI images and input them into two residual networks for fire detection. Zhao et al. [43] obtained candidate areas through saliency technology and then determined whether there was smoke in the candidate areas through the AlexNet network [44–46].

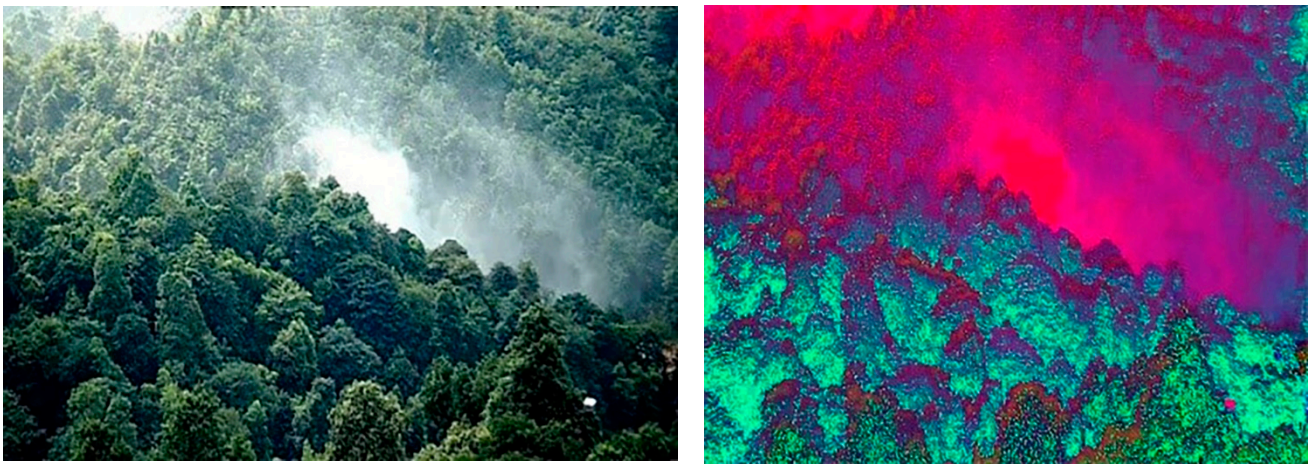


Figure 2. RGB diagram to HISI diagram.

To investigate which deep CNN algorithm can perform the best for early fire detection, this paper implements and compares four deep CNN algorithms for fire detection in real time. These algorithms were developed and trained using a huge fire data base with more 12,000 images. Based on the validation testing, the optimal detection performance among the four algorithms was determined, which can provide some alternative ways to detect forest fire accident prevention with high accuracy in real time.

2. The proposed Framework

2.1. Convolutional Neural Network

Figure 3 shows the design of the algorithm flow for forest fire smoke detection based on a convolutional neural network (CNN).

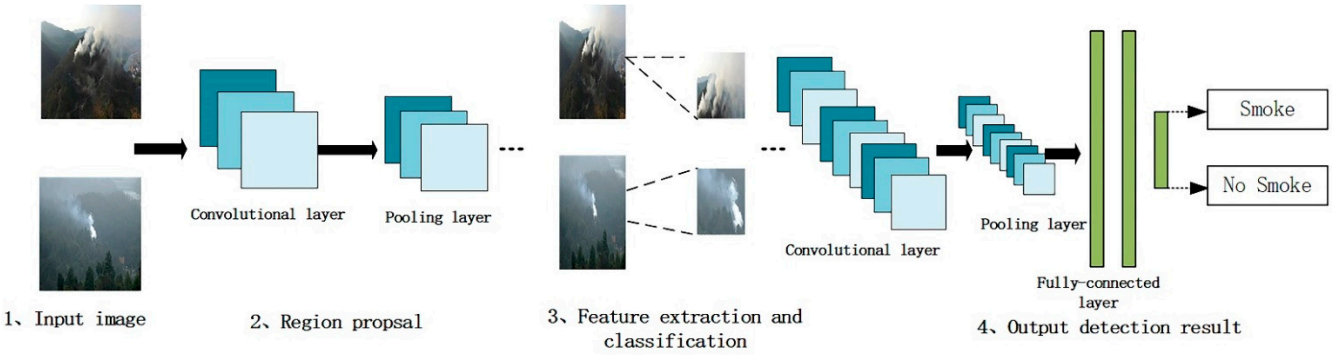


Figure 3. Flow of forest fire smoke detection algorithm based on CNN.

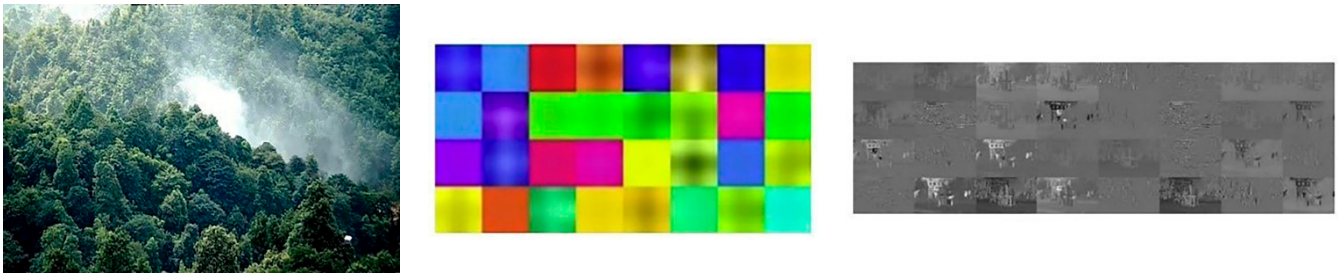
As shown in Figure 3, the CNN forest fire smoke detection includes several steps. Firstly, the CNN algorithm analyzes the input images and suggests different functional regions through methods such as convolution and pooling. Secondly, it uses region-based target detection to determine whether there is fire in the proposed region through the convolution layer, the pooling layer, and the fully connected layer. The convolutional layer is the core of the central nervous system. Unlike other neural networks that use concatenated weights and weighted sums, the convolution layer uses an image transform filter called a convolution kernel to generate a feature map of the original image. The convolution layer is actually a set of convolution kernels. The convolution kernel slides over the image and generates feature maps by floating the weights of pixels and computing new pixels. The feature map reflects one aspect of the original image. The output feature graph (y) of the convolution layer can be computed as:

$$y = \sum_{j=0}^{J-1} \sum_{i=0}^{I-1} w_{ij} x_{m+1,n+1} + b, (0 \le m \le M, 0 \le n \le N) \quad (1)$$

where $W \times H$ represents the input image with a size of $W \times H$; W represents the width of the image; H represents the height of the image; W_{ij} represents the convolution kernel of size $J \times I$; and b represents the bias. In practice, the values of W and b can be determined by training on the image data sets.

In accordance with Equation (1), Figure 4a–c show an example of forest fire smoke, its detection using the CNN algorithm for the 32 cores of the first convolutional layer in the Inception ResNet, and its responding 32 feature maps of the fire images generated by these cores. From Figure 4, it can be seen that the number of Eigen maps equals the number of the convolution kernels. For example, if there are three convolution kernels in this layer, three feature graphs will be generated. Additionally, the color of the pixel illustrates the degree of activation, with black pixels representing strong negative activation, gray pixels representing weak activations, and white pixels indicating strong activation. Compared with the original image in Figure 4a, Figure 4c shows that, in this example, the feature graph generated by the convolution kernel number 14 of this layer was activated at the edge. In addition, the feature graph generated by convolution kernel number 26 was activated, which was on the orange areas in the conventional kernels. Thus, the feature detection of the early layers mainly learns and extracts simple features such as colors, edges, etc. However, this example demonstrated that these simple features may not be able to distinguish the fire from the disturbances in complex scene or with multiple disturbing

events. Therefore, it is necessary to develop more advanced fire detection algorithms that can extract complex image features for fire detection in practical scenes.



(**a**) (**b**) (**c**)

Figure 4. This is a figure. Schemes follow another format. If there are multiple panels, they should be listed as: (**a**) Original image; (**b**) Convolution kernel; (**c**) Feature map.

2.2. Deep Convolutional Neural Network

To address the limitations of the CNN algorithm, deep convolutional neural networks can be used to detect forest fire smoke. For the same example as in Figure 4a–c, compare the kernel samples in the first, third, and sixth convolutional layers of Inception Resnet V2 using the deep CNN algorithm.

Specifically, in this paper, two deep CNN feature extraction networks were selected, including the Inception ResNet V2 [11] and Darknet-53 [47]. For each feature extraction network, 235 and 53 convolutional layers were used, respectively. In addition, four image target detection networks including the Faster R-CNN, SSD, YOLOv3, and EfficientDet [47,48] were selected to construct the image fire detection algorithm. These image target detection networks are expected to have excellent performance in detection accuracy and speed. In the following sections, the fundamentals of these four image target detection networks are introduced.

2.2.1. Faster R-CNN

Figure 5 shows the structure of the Faster R-CNN algorithm. It can be seen that the Faster R-CNN has two stages. In the first stage, the feature map of the original image is generated through the feature extraction network such as VGG, ResNet, Inception, Inception ResNet, and the regional proposal network (RPN). The proposed regions with target fractions and positions are predicted using the feature graphs obtained from some selected intermediate convolution layers. This stage outputs only scores that estimate the probability of each proposed object or non-object and box regression through two types of SoftMax layers and a robust loss function (smoothing L1).

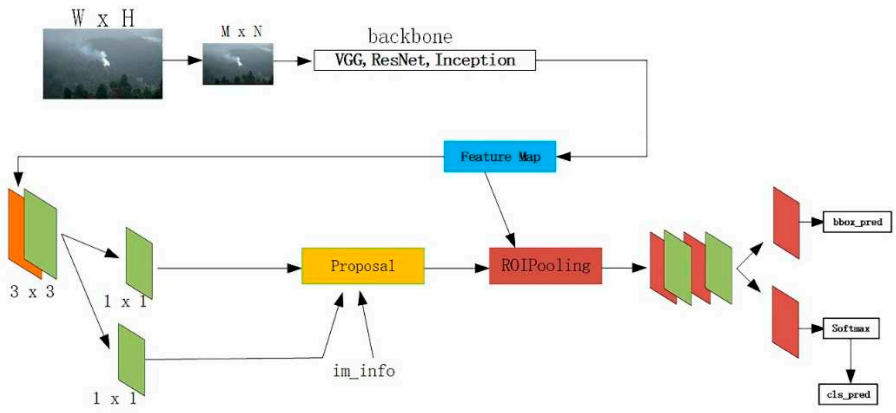


Figure 5. The Faster R-CNN schematic diagram of fire detection algorithm based on deep CNN.

In the second stage, the location of the proposed area is used to crop features from the same intermediate feature map through the ROI pooling. The area feature maps for each suggested area are fed to the rest of the network to predict scores for a particular category and refine the box locations. Such a network achieves partial computation sharing by pruning suggestions in the first stage from feature graphs generated by the same intermediate convolution layer. This method avoids the input of each proposed region into the front CNN calculation region feature map. However, each proposal area must be entered into the rest of the network for separate calculations. Therefore, the detection speed is highly dependent on the number of suggested areas from the RPN. In addition, due to the fact that the Faster R-CNN is a two-stage target detection network, the detection speed is relatively low.

2.2.2. SSD

The SSD is a one-stage target detection network that predicts object classes and locations through a forward CNN. The SSD structure can be divided into three steps including:

1. The basic convolutional layer consisting of VGG, ResNet, Inception, Inception RESnet-V2, and other feature extraction networks. The middle convolution layer of this step generates a large-scale feature map that can be divided into more units and has a smaller receptive field to detect smaller objects;
2. The additional convolutional layer is connected to the last layer of the basic convolutional network, which generates a multi-scale feature map that has a larger receptive field for larger object detection;
3. The prediction convolution layer of small convolution kernel is used to predict the position and confidence of bounding boxes of multiple categories.

From the operational steps, it can be seen that to maintain translation variance, the SSD network selects earlier layers to generate large-scale feature maps for detecting small objects. The features in the images of these early layers may not be complex enough, resulting in relatively poor detection for smaller objects.

2.2.3. YOLOv3

To improve the detection accuracy for smaller objects, YOLOv3 was developed by referring to the residual network. The YOLOv3 is also a one-stage strategy, which has high detecting speed. The architectural details of the YOLOv3 algorithm are as follows: It uses Darknet-53 without the last three layers to generate a small-scale feature image that is 32 times down-sampled from the original image. For example, if the original image is 416×416 in size, the element map will be 13×13 in size. Small-scale feature maps are used to detect large objects. Unlike the SDD, which selects an earlier layer to generate a largescale element map, the YOLOv3 generates a large-scale element map by up-sampling a small-scale element map and connecting it with an earlier layer's element map. Such large-scale feature maps with earlier layers of location information and deeper complex features are used to detect small objects. The three scales of the feature map are 8, 16, and 32 times down-sampled from the original image.

2.2.4. EfficientDet

The EfficientDet is also a two-stage feature extraction network that has a unique feature. This network is developed based on three or more great characteristics from some other excellent neural networks. Below are some examples of the three characteristics which had been combined to develop EfficientDet:

1. The residual neural network, which can increase the depth of neural network and realize feature extraction through a deeper neural network;
2. Changing the number of feature layers extracted from each layer to achieve more feature extraction and obtain more features in addition to improving the width;
3. Increasing the resolution of the input picture so that the network can learn and express more abundantly, which is conducive to improving accuracy.

The EfficientDet will also scale the baseline model while adjusting the depth, width, and input image resolution to complete an excellent network design. In MobileNet, the scaling model is realized by using a scaling factor, α. Different α results in different precision. $\alpha = 1$ represents the baseline model. The ResNet also has a baseline model, which is implemented by changing the depth of the image.

3. Algorithm Training

3.1. Image Dataset

Although advances in deep learning provide potential new solutions for visual forest fire detection, due to the limitation of budget, it may not be possible to conduct a large number of experiments to obtain the real forest fire image data set. Thus, this study used three different types of data sources including a computer-simulated smoke based on fluid dynamics, a crawler to crawl open data on the web, and forest fire smoke data taken by the authors. With all the three sources, 17,840 smoke image data sets were obtained using the data image augmentation technique. These data sets included 12,640 "forest fire smoke" images and 5200 non-forest fire smoke images. Figure 6 shows an example of part of the used data. Among these data, 70% was used for training, and 30% was used for testing. The training and testing data were randomly selected. The data set that we used in the experiments can be freely download via ZHENG data set 2021.

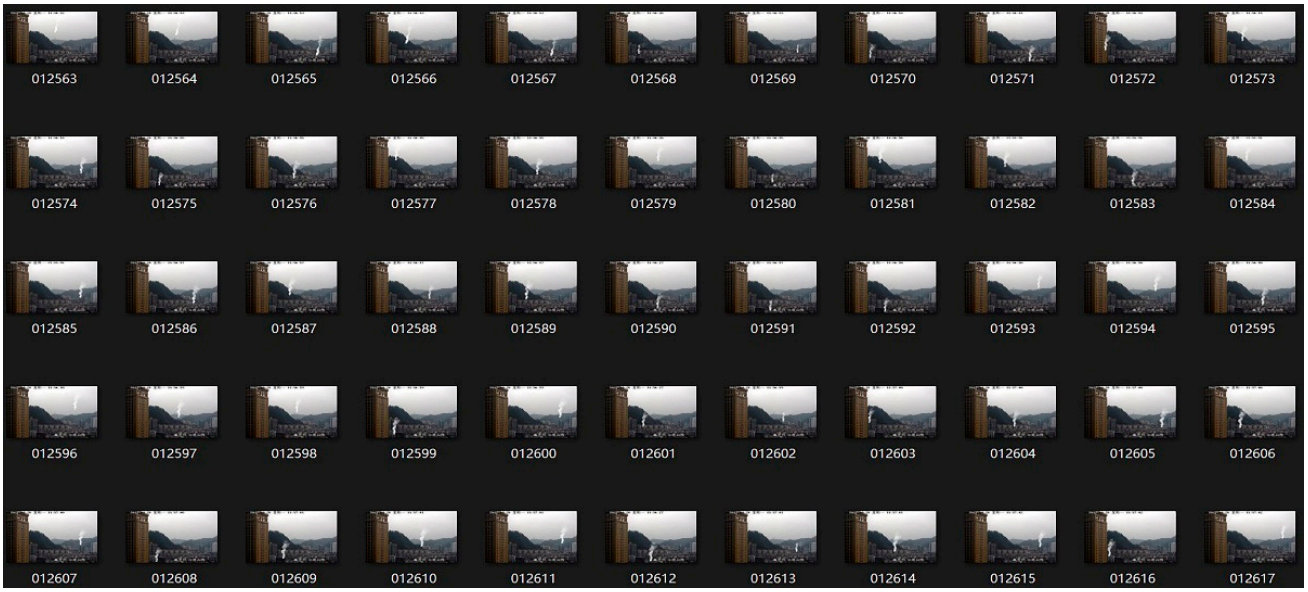


Figure 6. Example of part of the used data sets.

3.2. Image Pre-Processing

Image pre-processing is a necessary step for image recognition and classification. Image normalization is a typical image pre-process that can prevent affine transformation and accelerate gradient descent to find the optimal solution. In this study, the experimental data set was pre-processed by normalization using Equation (2).

$$img_n = \frac{img}{255.0} \tag{2}$$

where img_n and img are the normalized and original pixel values of the image, respectively. The normalization using Equation (2) was computed pixel by pixel.

After normalization, flipping and clipping were applied. Flipping and clipping is one of the earliest and most widely used methods of image augmentation. Flipping images left and right usually does not change the category of the object. In addition, through random clipping of the image, the object appears in different positions of the image in

different proportions, which reduces the sensitivity of the model to the target location. Additional pre-processing by changing image color characteristics such as brightness, contrast, saturation, and hue was also conducted, followed by the image augmentation. Figures 7–10 show the process of loss decline during 100 training iterations, from which it can be seen that Efficient-Det converged the fastest.

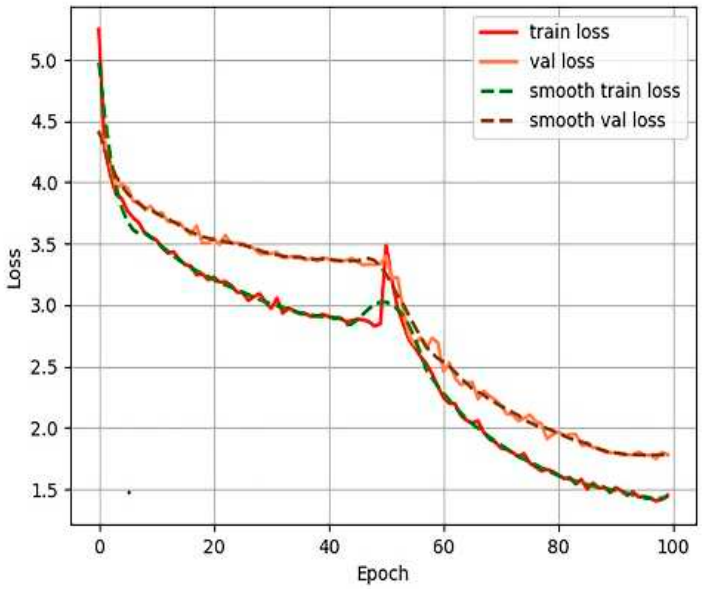


Figure 7. Loss curve of SSD.

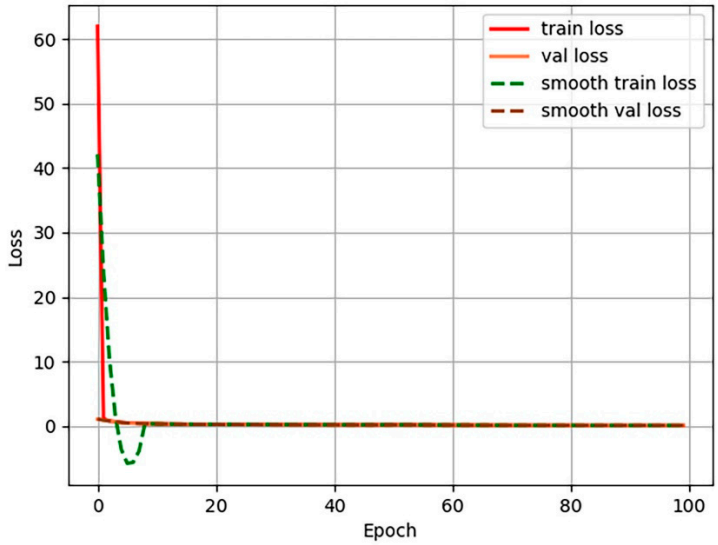


Figure 8. Loss curve of EfficientDet.

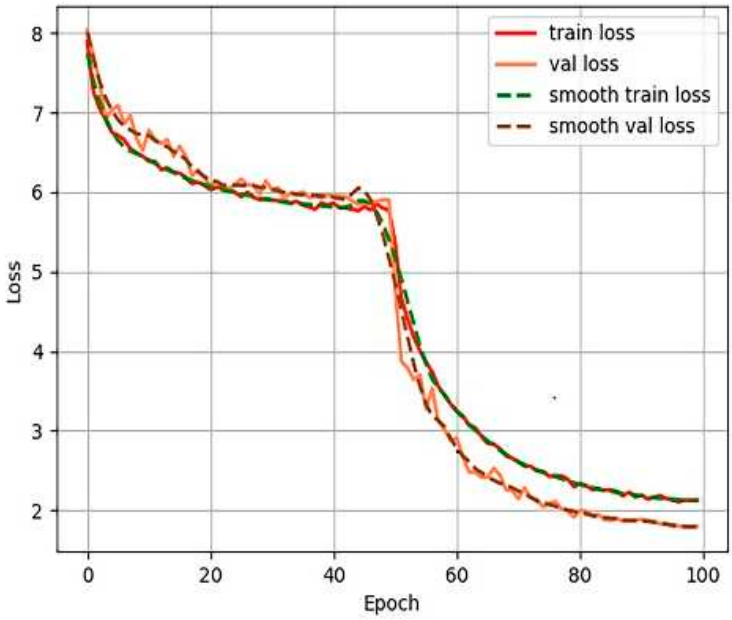


Figure 9. Loss curve of YOLOv3.

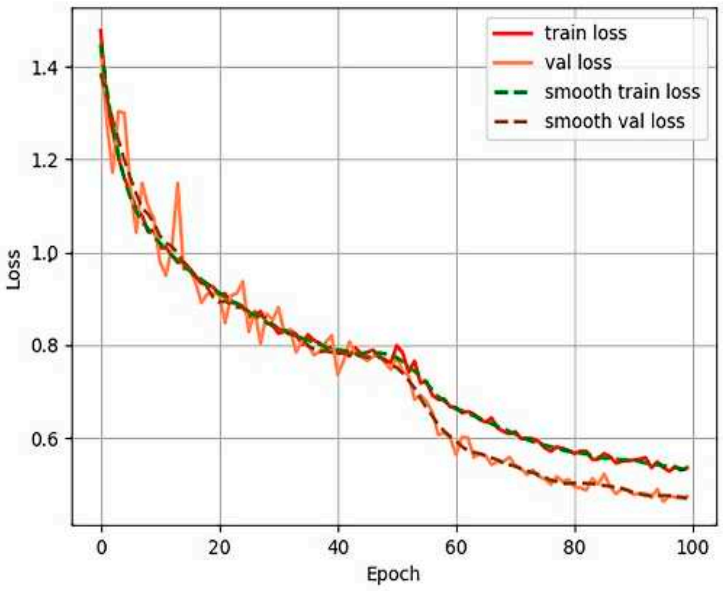


Figure 10. Loss curve of Faster R-CNN.

3.3. Transfer Learning and Training

Since the four deep CNN algorithms introduced in Section 2 were trained based on the large-scale image data sets and showed excellent performance in image target detection, this paper transferred the pre-trained network to large-scale image data. The transfer learning strategy is the front end of the reserved feature extraction network, only fine-tuning the network on the training and verification.

The training was then performed using the platform of the Intel(R) Xeon(R) W-2223 CPU @ 3.60 GHz, 16 GB DDR4 RAM 2400 MHz, CUDA10.2 GPU NVIDIA Quadro RTX 5000, and Quadro RTX 4000 distributed training unit. The operating system is Ubuntu 18.0.4 workstation. Figures 11–14 show some example smoke detection after the 100 iterations of training. It can be found that the confidence of Faster R-CNN is lower than that of YOLOv3, SSD, and EfficientDet.

Figure 11. Faster R-CNN close-range detection effect of forest fire smoke.

Figure 12. YOLOv3 remote detection effect of forest fire smoke.

Figure 13. EfficientDet remote detection effect of small-target forest fire smoke.

Figure 14. SSD remote detection effect of forest fire smoke.

4. Validation Testing and Discussion

With the training completed, validation testing was then performed on the remaining 30% of the total data to evaluate the performance of the deep CNN algorithms. Data composition is shown in the following Table 1.

Table 1. Data sets' quantity statistics and division.

Dataset	Smoke Image	Interference Image	Total	Divide
Data1	12,640	5200	17,840	Training
Data2	4210	1730	5940	Validating
Data3	4210	1730	5940	Test

5. Evaluation Index

To better evaluate the accuracy of forest fire smoke recognition, three indicators were used herein, including pixel accuracy, category average accuracy, and FPS for model performance evaluation. Larger values of the four indicators corresponded to superior recognition effects. Precision (P) and Recall (R) are the two simplest evaluation indicators and represent the proportion of correctly classified images out of the total number of images and the number of correctly classified images out of the images that should be correctly classified. The specific equations are shown below:

$$P = \frac{TP}{TP + FN} \tag{3}$$

$$R = \frac{TP}{TP + FP} \tag{4}$$

The Mean Average Precision (MAP) provides a comprehensive measure of the average accuracy of the detected target, and it indicates the average of each category of Average Precision (that is, the average accuracy of all categories is summed and divided by all categories). The specific equation for MAP is as follows:

$$MAP = \frac{\sum Average\ Precison}{N(Class)} \tag{5}$$

Figure 15 shows the average accuracy and detection time for fire smoke detection based on the four deep CNN algorithms. All the four deep CNN methods achieved high average accuracy more than 85%, indicating that it is feasible to detect forest fires in images by using deep CNNs. Among the investigated algorithms, the EfficientDet method showed the highest accuracy of 95.7%.

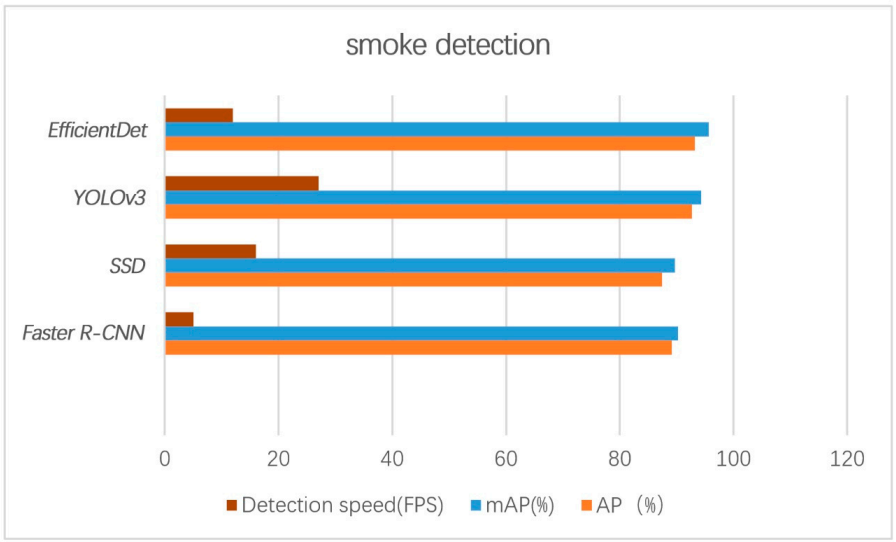


Figure 15. Histogram of forest fire smoke mAP and AP.

In addition, the detection speeds of the one-stage algorithms were shown to be faster, with more than 15 frames per second, indicating that they can detect fire smoke in real time. Among the four algorithms, the YOLOv3 showed the highest detection speed of 27 FPS.

The average measurement accuracy and detection speed of the four investigated deep CNN algorithms compares the measurement accuracy and its mean value, in addition to the detection time of the four algorithms. It can be seen that the EfficientDet method showed the highest average detection accuracy and the YOLOv3 has the fastest real-time detection speed.

6. Discussion

Faster R-CNN has higher detection accuracy, while the YOLO series is faster. Faster R-CNN uses a two-stage scheme to detect the target. The feature was discovered using the best network followed by adjusting the frame. However, the two-stage scheme can only be completed in one stage when the YOLO series method is applied. The core of the Faster R-CNN is to find the network with best performance and then assemble networks together to produce better results. Based on a multi-network fusion scheme, features of Faster R-CNN are very precise, but it yields slow computation, which is detrimental to the real-time nature of the forest fire smoke detection.

The emergence of YOLOv3 solved this challenge. The most significant features of the YOLOv3 are that it is faster and more accurate than the Faster R-CNN. The forest fire smoke detection results in Figure 10 showed that the maximum detection speed of YOLOv3 reached 27 frames/second, while the real-time detection performance of the Faster R-CNN was the worst among the four methods of target detection, with detection speed of 5 frames/second. Compared to the Faster R-CNN detection model requiring object proposals, the SSD method completely eliminates the stages of proposals generation, pixel resampling, or feature resampling, making it easier to optimize training and to integrate the detection model into the system. Although the detection speed for the SSD method of 16 frames per second meets the requirements for real-time detection, its detection accuracy is 87.5%, which is the lowest compared with other models.

Based on the results of the Scalable Neural Network (EfficientNet), EfficientDet can be combined with a new bi-directional feature network (BiFPN) and new scaling rules to achieve SOTA accuracy. Compared to the previous most cutting-edge detection algorithm, EfficientDet's volume is reduced to one-ninth of the original, and the computation time is also greatly reduced. This study developed a small EfficientDet-D0 baseline from the D0 to D7 models to improve the detection accuracy gradually while the computation effort was also decreased. According to the experimental results, the detection speed is 12 frames/second, and the detection accuracy is up to 95.7%.

Deep learning neural networks enable the capability of detecting the forest fire smoke without dependence on manual feature extraction through their special network architecture. To achieve such an end-to-end detection, a neural network model is constructed using generalized patterns, and a large number of environmental data sets is introduced for training for an effective detection. In this study, there are two major contributions to the fields:

1. As the forest fire smoke has its special nature, a real forest fire smoke data set is hard to obtain through experiments. This paper develops a computer simulation model based on the Navier–Stokes equation of fluid dynamics to simulate smoke, which can be used to supplement the forest fire smoke training data set by combining different smoke patterns obtained from the simulation with real field scenes. Such a simulation model solves the challenge that the forest fire smoke data set is difficult to obtain.

2. By constructing a large amount of forest fire smoke data sets to train deep learning target detection models, four forest fire smoke detection models with good generalization performance were obtained in this study, which makes the detection of forest fire smokes in complex scenes feasible.

This study shows that it is very effective to use simulated smoke to supplement the data set for training of neural networks in the absence of forest fire smoke data. With the trained models, the network model can detect the forest fire smoke in real time through the front-end remote video monitoring equipment. In practice, it is very easy to implement the trained models on the server with support of onsite cameras to monitor the all-weather fire occurrence and provide early warning of fire in forest areas.

Forest fire smoke has very large variations in color, texture, and shape, and it is crucial to establish a standard forest fire smoke database. Usually, data enhancement techniques are used to expand the smoke data set, but the data enhancement techniques do not increase the data or video surveillance scenes. It may reduce the robustness and effectiveness of the trained model in recognizing forest fire smoke scenes that are not included in the training set. Thus, accurate recognition of smoke is still challenging. In addition, the increase of the recognition accuracy of the target detection model requires computation power and memory. Therefore, in the future, while ensuring the recognition accuracy, more studies are needed to improve the computation efficiency of the detection algorithm network structure and make them more convenient to be deployed in practical field scenes.

7. Conclusions

To improve the performance of machine vision forest fire smoke detection, this paper investigated the feasibility of using the advanced object detection deep convolutional neural network of Faster R-CNN, SSD, YOLOv3, and EfficientDet to detect forest fire smoke. The deep CNN algorithm can automatically extract complex image fire features for fire detection in different scenes. The experimental evaluation results show that the four investigated algorithms all achieved acceptable average accuracy, with the EfficientDet showing the highest accuracy of mAP, up to 95.7%. The one-stage algorithms including YOLOv3 and SSD achieved real-time detection of more than 16 frames/s with the YOLOv3 the fastest of up to 27 frames/s.

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

Funding: This research work was supported by the Natural Science Foundation of China, grant number 31800549 and 32171797.

Institutional Review Board Statement: Not applicable.

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

References

1. Yuan, D.; Chang, X.; Huang, P.Y.; Liu, Q.; He, Z. Self-supervised deep correlation tracking. *IEEE Trans. Image Processing* **2020**, *30*, 976–985. [CrossRef] [PubMed]
2. Hu, S.; Zhu, F.; Chang, X.; Liang, X. Updet: Universal multi-agent reinforcement learning via policy decoupling with transformers. *arXiv* **2021**, arXiv:2101.08001.
3. Wang, F.; Zhu, L.; Liang, C.; Li, J.; Chang, X.; Lu, K. Robust optimal graph clustering. *Neurocomputing* **2020**, *378*, 153–165. [CrossRef]
4. Bai, X.; Zhu, L.; Liang, C.; Li, J.; Nie, X.; Chang, X. Multi-view feature selection via nonnegative structured graph learning. *Neurocomputing* **2020**, *387*, 110–122. [CrossRef]
5. Singh, A.K.; Lv, Z.; Lu, H.; Chang, X. Guest editorial: Recent trends in multimedia data-hiding: A reliable mean for secure communications. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *11*, 1795–1797. [CrossRef]
6. Zhang, D.; Yao, L.; Chen, K.; Wang, S.; Chang, X.; Liu, Y. Making sense of spatio-temporal preserving representations for EEG-based human intention recognition. *IEEE Trans. Cybern.* **2019**, *50*, 3033–3044. [CrossRef]
7. Yan, C.; Zheng, Q.; Chang, X.; Luo, M.; Yeh, C.H.; Hauptman, A.G. Semantics-preserving graph propagation for zero-shot object detection. *IEEE Trans. Image Processing* **2020**, *29*, 8163–8176. [CrossRef]
8. Liu, W.; Chang, X.; Chen, L.; Phung, D.; Zhang, X.; Yang, Y.; Hauptmann, A.G. Pair-based uncertainty and diversity promoting early active learning for person re-identification. *ACM Trans. Intell. Syst. Technol.* **2020**, *11*, 1–15. [CrossRef]

9. Pan, J.; Ou, X.; Xu, L. A Collaborative Region Detection and Grading Framework for Forest Fire Smoke Using Weakly Supervised Fine Segmentation and Lightweight Faster-RCNN. *Forests* **2021**, *12*, 768. [CrossRef]
10. Nebot, À.; Mugica, F. Forest Fire Forecasting Using Fuzzy Logic Models. *Forests* **2021**, *12*, 1005. [CrossRef]
11. Chen, K.; Yao, L.; Zhang, D.; Wang, X.; Chang, X.; Nie, F. A semisupervised recurrent convolutional attention model for human activity recognition. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 1747–1756. [CrossRef] [PubMed]
12. Fujiwara, N.; Terada, K. Extraction of a smoke region using fractal coding. In Proceedings of the IEEE International Symposium on Communications and Information Technology 2004, Alexandria, Egypt, 28 July–1 August 2004; Volume 2, pp. 659–662.
13. Li, C.; Peng, J.; Yuan, L.; Wang, G.; Liang, X.; Lin, L.; Chang, X. Block-wisely supervised neural architecture search with knowledge distillation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1989–1998.
14. Liu, C.; Chang, X.; Shen, Y.D. Unity style transfer for person re-identification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6887–6896.
15. Zhang, M.; Li, H.; Pan, S.; Chang, X.; Su, S. Overcoming multi-model forgetting in one-shot NAS with diversity maximization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 7809–7818.
16. Zhu, F.; Zhu, Y.; Chang, X.; Liang, X. Vision-language navigation with self-supervised auxiliary reasoning tasks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10012–10022.
17. Zhu, Y.; Zhu, F.; Zhan, Z.; Lin, B.; Jiao, J.; Chang, X.; Liang, X. Vision-dialog navigation by exploring cross-modal memory. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10730–10739.
18. Maruta, H.; Nakamura, A.; Yamamichi, T.; Kurokawa, F. Image based smoke detection with local hurst exponent. In Proceedings of the 2010 IEEE International Conference on Image Processing, Hong Kong, China, 26–29 September 2010; pp. 4653–4656.
19. Han, M.; Wang, Y.; Chang, X.; Qiao, Y. Mining inter-video proposal relations for video object detection. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 431–446.
20. Zhang, J.; Wang, M.; Li, Q.; Wang, S.; Chang, X.; Wang, B. Quadratic Sparse Gaussian Graphical Model Estimation Method for Massive Variables. In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, Yokohama, Japan, 11–17 July 2020; pp. 2964–2972.
21. Li, Z.; Chang, X.; Yao, L.; Pan, S.; Zongyuan, G.; Zhang, H. Grounding visual concepts for zero-shot event detection and event captioning. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 297–305.
22. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; Zhang, C. Connecting the dots: Multivariate time series forecasting with graph neural networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 753–763.
23. Appana, D.K.; Islam, R.; Khan, S.A.; Kim, J.M. A video-based smoke detection using smoke flow pattern and spatial-temporal energy analyses for alarm systems. *Inf. Sci.* **2017**, *418*, 91–101. [CrossRef]
24. Huang, P.Y.; Chang, X.; Hauptmann, A.; Hovy, E. Forward and backward multimodal NMT for improved monolingual and multilingual cross-modal retrieval. In Proceedings of the 2020 International Conference on Multimedia Retrieval, Dublin, Ireland, 8–11 June 2020; pp. 53–62.
25. Huang, P.Y.; Chang, X.; Hauptmann, A.; Hovy, E. Memory-based network for scene graph with unbalanced relations. In Proceedings of the 28th ACM International Conference on Multimedia, Dublin, Ireland, 8–11 June 2020; pp. 2400–2408.
26. Cheng, X.; Zhong, Y.; Harandi, M.; Dai, Y.; Chang, X.; Drummond, T.; Li, H.; Ge, Z. Hierarchical neural architecture search for deep stereo matching. *arXiv* **2020**, arXiv:2010.13501.
27. Zhang, M.; Li, H.; Pan, S.; Chang, X.; Ge, Z.; Su, S.W. Differentiable Neural Architecture Search in Equivalent Space with Exploration Enhancement. *NeurIPS* **2020**. Available online: https://proceedings.neurips.cc/paper/2020/file/9a96a2c73c0d477ff2a6da3bf538f4f4-Paper.pdf (accessed on 10 January 2021).
28. Yuan, F.; Shi, J.; Xia, X.; Fang, Y.; Fang, Z.; Mei, T. High-order local ternary patterns with locality preserving projection for smoke detection and image classification. *Inf. Sci.* **2016**, *372*, 225–240. [CrossRef]
29. Liu, W.; Kang, G.; Huang, P.Y.; Chang, X.; Qian, Y.; Liang, J.; Gui, L.; Wen, J.; Chen, P. Argus: Efficient activity detection system for extended video analysis. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops, Snowmass Village, CO, USA, 1–5 March 2020; pp. 126–133.
30. Wu, M.; Pan, S.; Zhou, C.; Chang, X.; Zhu, X. Unsupervised domain adaptive graph convolutional networks. In Proceedings of the Web Conference 2020, Taipei, Taiwan, 20–24 April 2020; pp. 1457–1467.
31. Ren, P.; Xiao, Y.; Chang, X.; Huang, P.Y.; Li, Z.; Chen, X.; Wang, X. A comprehensive survey of neural architecture search: Challenges and solutions. *arXiv* **2020**, arXiv:2006.02903. [CrossRef]
32. Han, X.F.; Jin, J.S.; Wang, M.J.; Jiang, W.; Gao, L.; Xiao, L.P. Video fire detection based on Gaussian Mixture Model and multi-color features. *Signal Image Video Process.* **2017**, *11*, 1419–1425. [CrossRef]
33. Gao, Y.; Cheng, P. Forest fire smoke detection based on visual smoke root and diffusion model. *Fire Technol.* **2019**, *55*, 1801–1826. [CrossRef]

34. Gao, Y.; Cheng, P. Full-Scale Video-Based Detection of Smoke from Forest Fires Combining ViBe and MSER Algorithms. *Fire Technol.* **2021**, *57*, 1637–1666. [CrossRef]
35. Liu, H.; Zheng, Q.; Luo, M.; Chang, X.; Yan, C.; Yao, L. Memory transformation networks for weakly supervised visual classification. *Knowl. -Based Syst.* **2020**, *210*, 106432. [CrossRef]
36. Ge, Z.; Mahapatra, D.; Chang, X.; Chen, Z.; Chi, L.; Lu, H. Improving multi-label chest X-ray disease diagnosis by exploiting disease and health labels dependencies. *Multimed. Tools Appl.* **2020**, *79*, 14889–14902. [CrossRef]
37. Zhang, L.; Chang, X.; Liu, J.; Luo, M.; Prakash, M.; Hauptmann, A.G. Few-shot activity recognition with cross-modal memory network. *Pattern Recognit.* **2020**, *108*, 107348. [CrossRef]
38. Chang, X.; Liang, X.; Yan, Y.; Nie, L. Guest editorial: Image/video understanding and analysis. *Pattern Recognit. Lett.* **2020**, *130*, 1–3. [CrossRef]
39. Frizzi, S.; Kaabi, R.; Bouchouicha, M.; Ginoux, J.M.; Moreau, E.; Fnaiech, F. Convolutional neural network for video fire and smoke detection. In Proceedings of the IECON 2016—42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; pp. 877–882.
40. Yin, Z.; Wan, B.; Yuan, F.; Xia, X.; Shi, J. A deep normalization and convolutional neural network for image smoke detection. *IEEE Access* **2017**, *5*, 18429–18438. [CrossRef]
41. Yin, M.; Lang, C.; Li, Z.; Feng, S.; Wang, T. Recurrent convolutional network for video-based smoke detection. *Multimed. Tools Appl.* **2019**, *78*, 237–256. [CrossRef]
42. Wang, Z.; Huang, M.; Zhu, Q.; Jiang, S. Smoke detection in storage yard based on parallel deep residual network. *Laser Optoelectron. Prog.* **2018**, *55*, 051008. [CrossRef]
43. Zhao, Y.; Ma, J.; Li, X.; Zhang, J. Saliency detection and deep learning-based wildfire identification in UAV imagery. *Sensors* **2018**, *18*, 712. [CrossRef]
44. Yuan, D.; Fan, N.; Chang, X.; Liu, Q.; He, Z. Accurate bounding-box regression with distance-iou loss for visual tracking. *arXiv* **2020**, arXiv:2007.01864. [CrossRef]
45. Ren, P.; Xiao, Y.; Chang, X.; Huang, P.Y.; Li, Z.; Gupta, B.B. A survey of deep active learning. *arXiv* **2020**, arXiv:2009.00236. [CrossRef]
46. Yan, C.; Chang, X.; Luo, M.; Zheng, Q.; Zhang, X.; Li, Z.; Nie, F. Self-weighted robust LDA for multiclass classification with edge classes. *Proc. ACM Trans. Intell. Syst. Technol.* **2020**, *12*, 1–19. [CrossRef]
47. Chen, J.; Wang, Y.; Tian, Y.; Huang, T. Wavelet based smoke detection method with RGB Contrast-image and shape constrain. In Proceedings of the 2013 Visual Communications and Image Processing (VCIP), Kuching, Malaysia, 17–20 November 2013; pp. 1–6.
48. Qi, X.; Ebert, J. A computer vision based method for fire detection in color videos. *Int. J. Imaging* **2009**, *2*, 22–34.