


Smoke Detection on Video Sequences Using 3D Convolutional Neural Networks

Gaohua Lin, Yongming Zhang, Gao Xu, and Qixing Zhang* , State Key Laboratory of Fire Science, University of Science and Technology of China, Hefei 230026, China

Received: 5 May 2018/**Accepted:** 13 February 2019

Abstract. Research on video smoke detection has become a hot topic in fire disaster prevention and control as it can realize early detection. Conventional methods use handcrafted features rely on prior knowledge to recognize whether a frame contains smoke. Such methods are often proposed for fixed fire scene and sensitive to the environment resulting in false alarms. In this paper, we use convolutional neural networks (CNN), which are state-of-the-art for image recognition tasks to identify smoke in video. We develop a joint detection framework based on faster RCNN and 3D CNN. An improved faster RCNN with non-maximum annexation is used to realize the smoke target location based on static spatial information. Then, 3D CNN realizes smoke recognition by combining dynamic spatial-temporal information. Compared with common CNN methods using image for smoke detection, 3D CNN improved the recognition accuracy significantly. Different network structures and data processing methods of 3D CNN have been compared, including Slow Fusion and optical flow. Tested on a dataset that comprises smoke video from multiple sources, the proposed frameworks are shown to perform very well in smoke location and recognition. Finally, the framework of two-stream 3D CNN performs the best, with a detection rate of 95.23% and a low false alarm rate of 0.39% for smoke video sequences.

Keywords: Video smoke detection, Faster RCNN, 3D convolutional neural networks, Optical flow, Support vector machine

1. Introduction

Fire has posed a serious threat to personal safety and property throughout the ages. Early detection is important as the damage caused by fire often increases exponentially over time, and once a fire becomes large, it is difficult to control. Compared to traditional fire detection methods, such as point-based sensors that are sensitive to temperature, smoke particles and gas, video fire detection (VFD) systems have many advantages [1]. Image sensors (CCD and CMOS) do not need to touch smoke particles or be close to the flame. Therefore, they can be used in tall spaces, such as shopping atriums, and outdoor open spaces, such as forest reserves. VFD systems are easy to embed in traditional surveillance systems, which can save cost.

* Correspondence should be addressed to: Qixing Zhang, E-mail: qixing@ustc.edu.cn



Many algorithms attempt to realize accurate fire detection based on images or video sequences. They can be divided into two types, flame detection and smoke detection, according to the objects of detection. Flames of high temperature are relatively easy to detect, particularly with the use of infrared camera [2]. In most fire accidents, smoke is produced in the smoldering phase, and flames are more likely to be covered; thus, smoke emerges before flames. Therefore, smoke video detection is a more efficient approach to early fire detection. A variety of algorithms have been proposed for video smoke detection in the past decades. They are centered around smoke motion, texture, color, geometric statistical features and so on [3–5]; these are known as traditional features or handcrafted features. Generally, approaches using handcrafted features contain three steps. First, use the foreground extraction method to generate the suspected smoke areas (or blocks); second, use handcrafted features to extract feature vectors; finally, train the classifier or classify objects in the train and test scenes. However, although smoke detection has been developed over a long period of time and much encouraging progress has been made, many problems persist, and it is still difficult to apply video smoke detection in real scenarios. Compared to images of faces, pedestrians and vehicles, smoke is flexible in shape. Additionally, motion, illumination, and the distance between smoke and camera also have very large variation range. Handcrafted features are often effective for only one scene with very weak robustness.

Convolutional neural networks (CNN) is successfully used to recognize handwritten characters in 1998 [6]. A breakthrough in computer visions based on CNN is presented at the ImageNet Challenge in 2012. In general, smoke detection can be considered a special type of object detection task in computer vision. Researchers have therefore attempted to address smoke detection by exploring successful deep learning techniques for generic object detection tasks. Many CNN-based methods were proposed for smoke detection and fire detection.

Yuan et al. [7] proposed a deep normalization and convolutional neural network (DNCNN) for smoke detection. Mao [8] proposed a novel fire detection method based on multi-channel convolutional neural network. Sharma [9] used an imbalanced fire images dataset to train two pretrained state-of-the-art Deep CNNs, VGG16 and Resnet50, for fire detection. Muhammad [10] proposed a cost-effective fire detection CNN architecture for surveillance videos focusing on computational complexity and detection accuracy. Xu [11] proposed using synthetic smoke images and domain adaptation to train CNN structures as the lack of data.

Previous studies have focused on using convolutional neural networks as feature extraction tool. The fire detection pipeline generally consists of three steps. Firstly, flame or smoke suspected regions are extracted based on traditional features, such as color [12], motion [13] and Haar [14]. The suspected region can be treated as a whole object and resized to a uniform size to accommodate CNN's input requirement. Some fire detection pipelines directly divide the whole image into blocks of fixed size to generate suspected region [15]. In the second step, CNN is used to extract features of suspected regions. Finally, flame or smoke recognition is achieved with softmax function or support vector machine. In these pipelines,

CNN only completed the classification task. Lin [16] detects wildland forest fire smoke with synthetic smoke images using faster RCNN, which is a framework for CNN to achieve end-to-end object detection.

On the other hand, previous studies are mainly based on images for flame and smoke detection. However, video sequences contain a wealth of temporal information. The occurrence of smoke can be considered a kind of action. Convolutional Neural Networks (CNN) have also been successfully used for video analysis, such as 3D CNN [17], which learns to capture spatiotemporal features. Inspired by action recognition with 3D convolutional neural networks, we develop a joint detecting framework based on faster RCNN [18] and 3D CNN. The former realizes the smoke target location, and the latter realizes smoke recognition.

The rest of the paper is organized as follows: in Sect. 2 we review convolutional neural network structures used in our work. Section 3 give a description of our proposed work. In Sect. 4 illustration of the dataset and experimental results are given. Finally, Sect. 5 summarizes the paper.

2. Related Work

2.1. Convolutional Neural Networks

Convolutional neural networks are inspired by the seminal work of Hubel and Wiesel [19] arguing that the visual cortex contains neurons that individually respond to small regions of the visual field. They are a special type of neural network architecture and are especially well adapted to computer vision applications, as they can be trained end to end from raw pixel values to classifier outputs. Figure 1 depicts the structure of AlexNet, a significant CNN architecture that revived interest in Convolutional neural networks in 2012 [6].

The core concepts of CNNs are receptive field and down sampling. First, because pixels within a neighborhood are usually highly correlated, CNNs use grouped local connections and feature sharing to reduce network parameters, unlike standard neural networks, which use one-to-one connections. Second, CNNs introduce a pooling step to reduce the resolution of convolutional feature maps and increase the receptive field size that allows the network to represent more abstract characteristics of the input as the network's depth increases.

2.2. Object Detection with Faster RCNN

The region-based CNN (RCNN) method [20] is a kind of CNN extension for solving object detection tasks that combine a region proposal method with a CNN architecture. Faster R-CNN [18] is an improved variant that pushes the use of CNN further by adding region proposal network after the last convolutional layer of a CNN to propose candidate regions, as shown in Fig. 2. RPN returns the potential object positions and score, which represents the probability of the object belonging to a class.

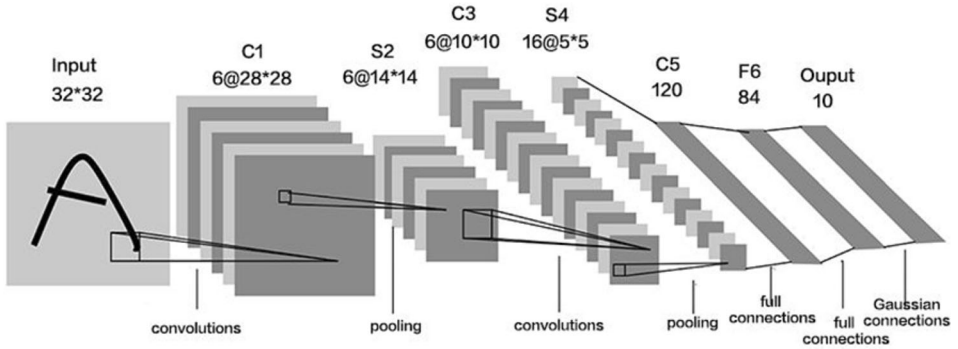


Figure 1. Structure of LeNet-5 [6]. C represents convolution layer, S represents sub-sampling, and F for fully connection.

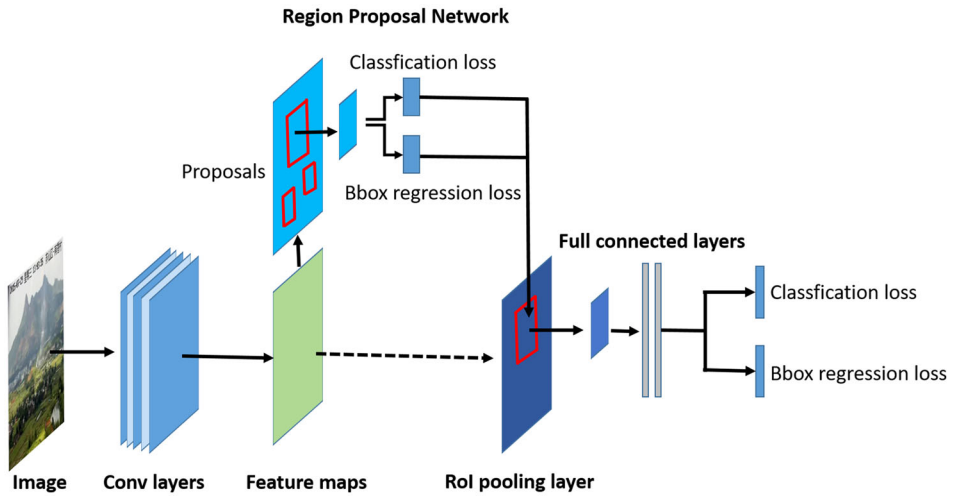


Figure 2. Structure of faster RCNN. Region proposal network proposes propose candidate regions.

2.3. Spatiotemporal Convolutional Networks

Convolutional networks can be applied to data of arbitrary dimensions in theory; this has inspired researchers to extend 2D spatial CNN to 3D spatiotemporal CNN for video analysis. Generally, there are three architectural design decision-based spatiotemporal CNNs: Two-Stream CNN [21], 3D CNN [17] and LSTM [22]. 3D CNNs are the most straightforward spatiotemporal networks that use 3D convolutional filters to operate video sequences. Figure 3 illustrates the difference in 2D convolutional filter and 3D convolutional filter.

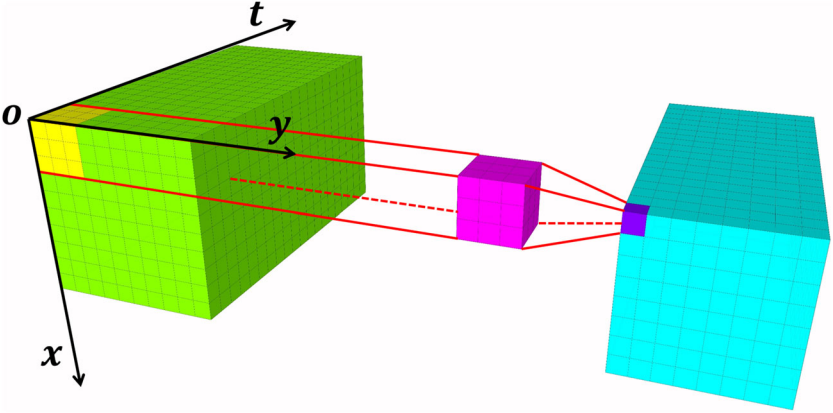


Figure 3. 3D convolution operation in video sequences.

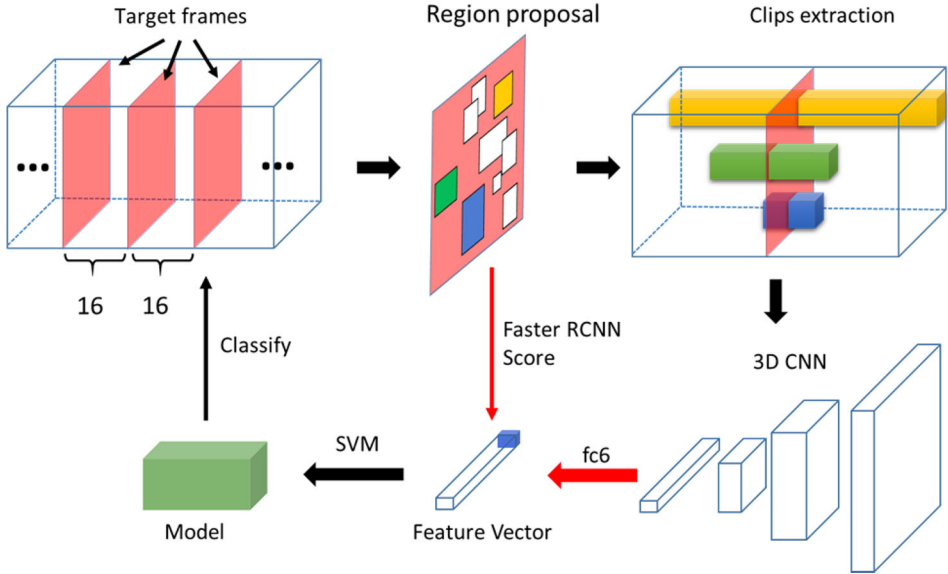


Figure 4. Overview of our joint detection framework for video smoke detection. Using faster RCNN to initially smoke detection on target frame.

3. The Proposed Approach

3.1. Overview of Methodology

Our joint detecting framework for video smoke detection is outlined in Fig. 4. First, faster RCNN is employed to generate suspected smoke boxes in target frames that are picked from video sequences at a fixed interval. When a suspected smoke box is detected in the target frame, a clip for the box is extracted by crop-

ping continuous frames around the target frame. Then, a designed 3D convolutional network extracts the spatio-temporal features of the clip. Finally, softmax or svm is used to train the video smoke detection model.

Fire is a small probability event, which means that most video sequences recorded by monitoring systems do not contain smoke. However, there are some CNN frameworks that can locate and classify tasks together in untrimmed videos, such as R-C3D [23] and S-CNN [24]. Preliminary detection based on frames is a sensible way to reduce computational complexity.

For an untrimmed video, we extract temporal sliding windows of lengths 16, 32 and 64 frames with a sliding stride of 16.

3.2. Smoke Clips Proposal with Faster RCNN

In [16], Faster R-CNN is used to detect wildland forest fire smoke based on synthetic smoke images. Synthetic smoke images are used to address the lack of smoke image data and eliminate sample annotation. Test results in [16] show that the faster RCNN model trained by synthetic smoke images is sensitive to columnar smoke plumes, but it is insensitive to thin smoke due to the lack of similar samples in the training data. Further testing results show that the false alarm rate is also relatively high. Figure 5 shows the box detection results of the model tested on Test Dataset, which will be introduced in detail in Sect. 4.1. Obviously, too many non-smoke objects were given a high score, indicating that this faster RCNN model has weak classification ability.

To improve the detection rate, we consider using 3D CNNs for further identification and regard the faster RCNN as a proposal method. In general video smoke detection frameworks, motion and color extraction are often used as suspected smoke region proposal methods whose purpose is to reduce the feature extraction region and locate the object.

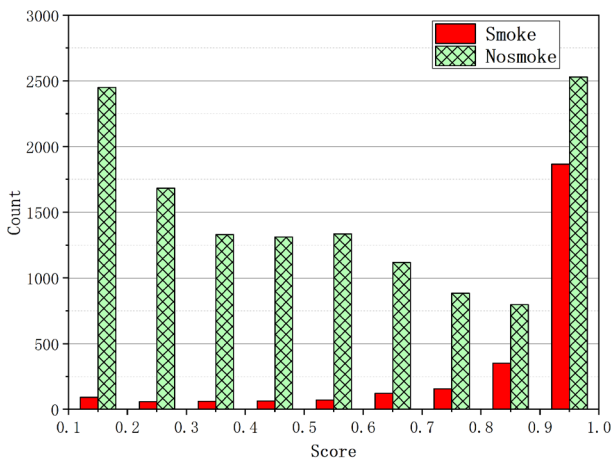


Figure 5. Box detection results of faster RCNN trained by synthetic smoke images.

3.3. Non-maximum Annexation

RPN in faster RCNN generates 300 proposals for each target frame, and some proposals overlap with each other. To reduce redundancy, faster RCNN adopts non-maximum suppression (NMS) on the proposed boxes based on their scores. However, NMS substantially reduces the number of proposals, but it is not suitable for suspected smoke proposal. First, a large amount of overlap still exists, as two proposals are reserved together when their intersection-over-union (IoU) is under 0.3. Additionally, the box proposed by NMS is too small to cover the whole smoke object.

Unlike rigid objects like faces and cars, smoke with a semitransparent property has a blurred boundary, which confuses RPN with the proposed precise boxes. The range of smoke expands gradually in the clip as smoke diffuses over time. Much temporal information is contained in the boundary of smoke, and this should be input into 3D CNN for feature extraction. With this aim, we propose Non-maximum annexation (NMA) to integrate the 300 boxes proposed by RPN. NMS and NMA algorithms are shown in Table 1. The bounding boxes generated by NMA do not overlap with each other, and each bounding box covers a whole smoke object. The different results of NMS and NMA are shown in Fig. 6.

3.4. Data Augmentation

Plentiful high-quality data are the key to great machine learning models. Models trained with small datasets often suffer from the problem of overfitting, as they do not adequately generalize data from the validation and test dataset. Data augmentation is a way to avoid the problem of lack of data. There are many approaches to augmenting data, such as adding noise and applying transformations on existing data or the simulation of data as in [16].

In this paper, we use equipped data augmentation methods to enrich the train dataset. In the data layer of the 3D convolutional neural network, horizontal flip

Table 1
NMS and Non-maximum Annexation

NMS	Rank 300 boxes by score If $\text{IoU}(\text{boundingbox}_i, \text{box}_j) > 0.3$, delete box_j Else if $\text{IoU}(\text{boundingbox}_i, \text{box}_j) < 0.3$, save box_j as boundingbox_{i+1} If score of $\text{boundingbox}_i > 0.8$ alarm Rank boxes by score
NMA	If score of $\text{box}_j < 0.01$, delete box_j Else if $\sum_{i=1}^I \text{IoU}(\text{boundingbox}_i, \text{box}_j) == 0$, save box_j as boundingbox_{i+1} Else if $\text{IoU}(\text{boundingbox}_i, \text{box}_j) < 0.6$ && $\sum_{k \neq i}^I \text{IoU}(\text{boundingbox}_k, \text{boundingbox}_i \cup \text{box}_j) == 0$, save $\text{boundingbox}_i \cup \text{box}_j$ as boundingbox_i Take $\text{boundingbox}_i (i = 1, 2 \dots I)$ as suspected smoke boxes

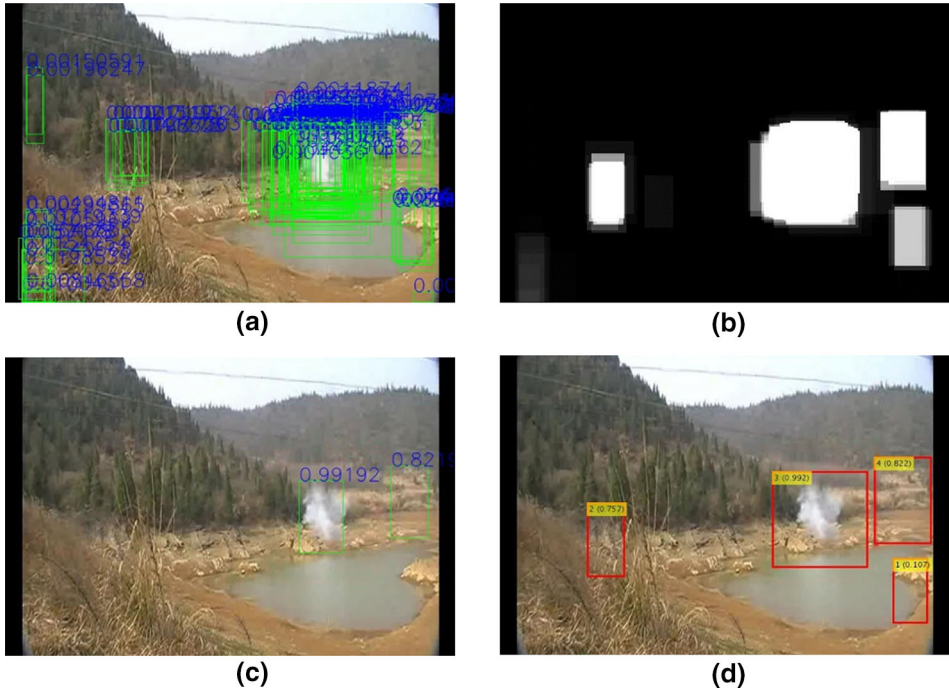


Figure 6. Difference between NMS and NMA. (a) 300 boxes proposed by RPN, (b) superposition of 300 boxes' scores, (c) bounding boxes generated by NMS, (d) bounding boxes generated by NMA.

and random crop are applied. Considering the upward movement of smoke, vertical flip has not been applied. The clips fed into the network are cropped randomly into 112×112 after being resized to 128×171 .

Brightness variation in videos taken by surveillance systems is inevitable given the influences of diurnal variation and weather changes. We vary the brightness of clips in the train dataset with increases and decreases in intensity to simulate these environmental changes. The train data are increased two times through these operations. The increase and decrease in brightness of each clip is controlled by a random number, as follows:

$$I_{increase} = I + (0.2 + 0.6 \times r) \times (255 - I)$$

$$I_{decrease} = (0.4 + 0.4 \times r) \times I$$

where r is a random number between 0 and 1 and I is the initial brightness of the clip.

In digital imaging equipment, Gaussian white noise and salt-pepper noise are both common sources of noise. To further enrich the train data, we increase the

data by two times the size of the original train data by adding Gaussian white noise and salt-pepper noise to the clips, as follows:

$$F_{\text{gaussian}} = F + \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

$$F_{s\&p} = F + s\&p(SNR)$$

where the mean value μ is equal to 0, the value of variance is 0.001 to 0.01, and SNR is the signal-to-noise ratio with a value of 0.01 to 0.1. Figure 7 shows an example of the new images generated from original images through data augmentation techniques.

Although convolutional neural networks can achieve end-to-end learning, it works like a black box, as what precisely has been learned is vague. Preprocessing on input data may show better performance than raw input with a clearer objective. In two stream convolutional neural networks [21], optical flow is used as an input in the temporal recognition stream and is significantly better than the train on raw stacked frames. As we expect 3D CNN to extract temporal information from the video sequences, preprocessing methods for motion information extraction are used in our experiments, such as optical flow and background subtraction. A sample of preprocessing results is illustrated in Fig. 7.

3.5. Spatial–Temporal Features Extracted by 3D CNN

The 3DCNN for Spatial–temporal features extraction used in our framework is based on C3D-v1.0 [17]. Sports-1M dataset [25], which is currently the largest

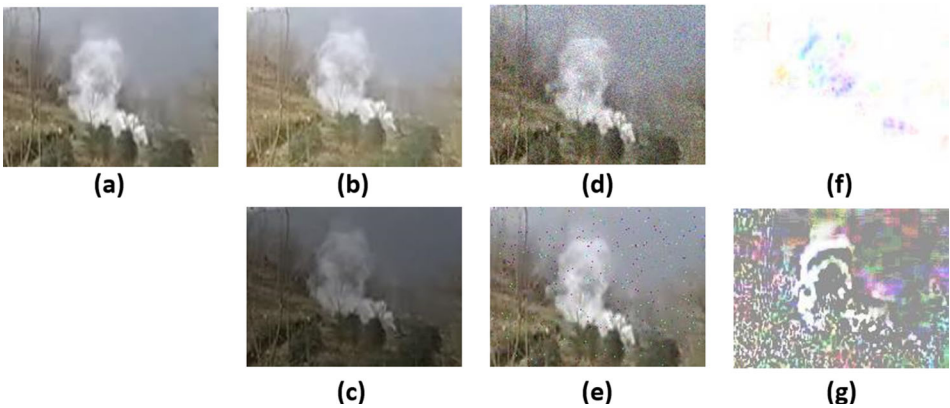


Figure 7. Example of data augmentation. (a) Original frame, (b) brightness increase, (c) brightness decrease, (d) Gaussian white noise, (e) salt-pepper noise, (f) optical flow frame, (g) background subtraction frame.

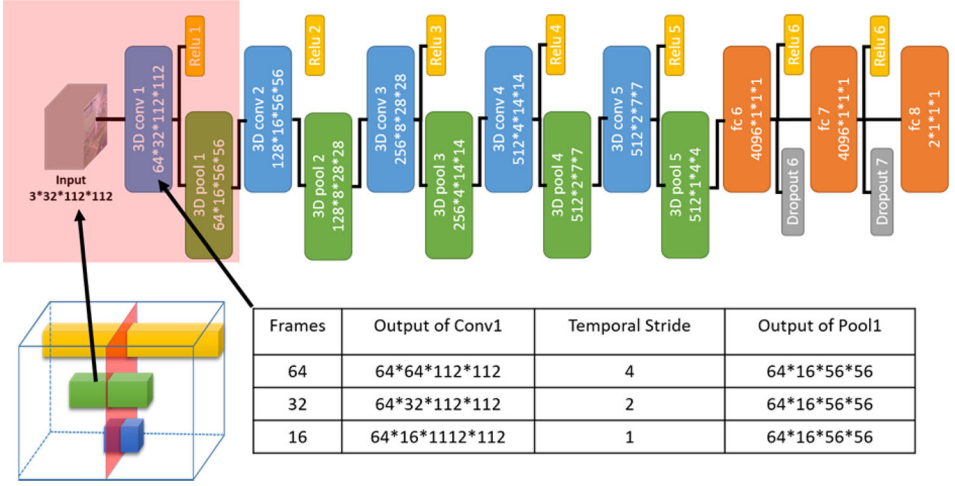


Figure 8. Structure of 3D convolutional neural network. The input can be selected in three clip lengths.

video classification benchmark, consists of 1.1 million sports videos and 487 sports categories. Even a smaller dataset UCF101 consists of 13,320 videos of a few seconds and 101 human action categories. Overfitting is highly possible, as our dataset is much smaller than Sports-1M and UCF101. We use a smaller network to prevent overfitting as shown in Fig. 8. The network consists of five 3D convolutional (3D conv) layers, five 3D max pooling (3D pool) layers, and three fully connected (fc) layers and a softmax loss layer to predict smoke labels. As the activation function, a rectified linear unit (ReLU) is placed at the output of each 3D convolutional layer and the first two fully connected (fc) layers. The first two fully connected layers are followed by dropout layers [26] to avoid overfitting.

In traditional uses of C3D, the input is a short 16-frame video clip. However, the diffusion of smoke is very slow, particularly in long distance monitoring surroundings such as forests. The motion is barely observable over time frames of less than a second. In addition to 16-frame clips, we also use 32-frame and 64-frame clips as inputs for 3D CNN. All clip frames are resized to 128×171 . The input dimensions are $3 \times \text{length} \times 128 \times 171$, and random crops of size $3 \times \text{length} \times 112 \times 112$ are used during training.

4. Experimental Evaluation

4.1. Experimental Data

The training of deep convolutional networks requires a substantial amount of data to fully optimize the network's parameters and weight values, such as ImageNet for image classification and Sport1M for action recognition. Insufficient training data lead to overfitting with poor classification performance on test datasets. However, the data for smoke video detection research are scarcer than those

available for video surveillance systems that record fire accidents with small probability. Obtaining data through experiments must also take into consideration security and economic problems.

The videos used in the experiment are collected from different publicly available datasets [27–29] and shooting experiments. For convenience of evaluation, all raw videos are cropped in the time dimension based on whether sequences contain smoke. The dataset consists of 38 smoke videos and 20 non-smoke videos. Although the smoke is relatively slight at the beginning of some smoke videos, we still mark the frames as smoke frames with the label of 1. Smoke videos are divided into a training dataset and test dataset, containing 20 and 18 videos, respectively. The duration, height and width of each video are shown in Figs. 9, 10 and 11. The scenes of these videos are mainly large outdoor space and forest. The distance of the smoke is between tens and several hundred meters.

Target frames are extracted from each video with a sliding stride of 16. Then, the faster RCNN model trained by synthetic smoke images and NMA are applied to generate bounding boxes. To evaluate the recognition frameworks, we annotate all bounding boxes with smoke or non-smoke. Table 2 shows the details of our dataset. The non-smoke boxes in the train dataset are cropped randomly and do not overlap with any smoke boxes.

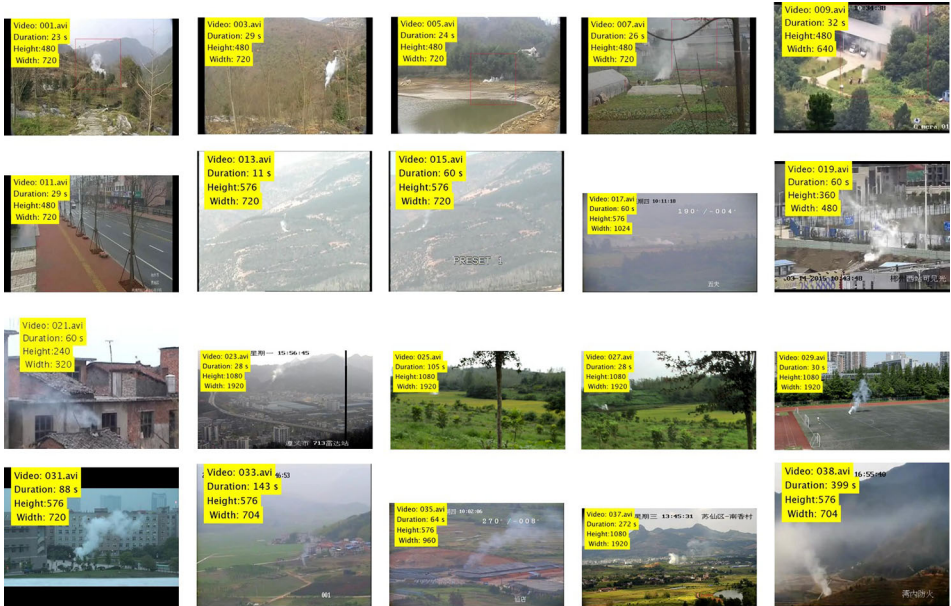


Figure 9. Smoke videos for training.



Figure 10. Smoke videos for testing.

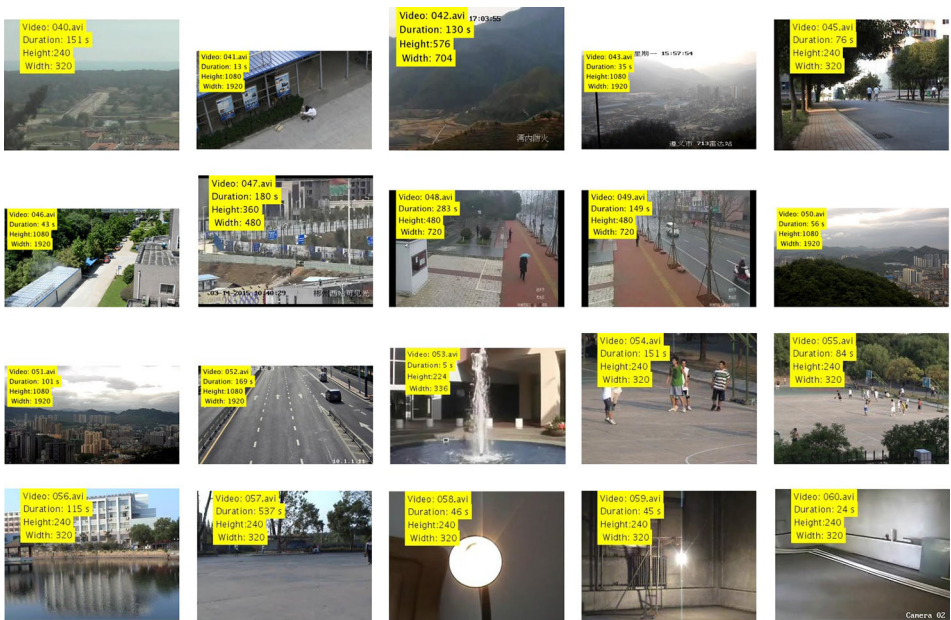


Figure 11. Non-smoke videos.

Table 2
Statistics of the Experimental Dataset

Dataset	Videos	Durations (s)	Target Frames	Bounding boxes	Smoke boxes	Non-smoke boxes
Train	20	1471	2436	6568	2168	4400
Test smoke	18	1549	2402	4619	2858	1761
Test non-smoke	20	2393	3953	11,682	0	11,682

4.2. Evaluation Methods

To test the performance of the proposed method, we introduce Accuracy Rate (AR), Detection Rate (DR), Precision Rate, and False Alarm Rate (FAR), as follows:

$$\text{AR} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \times 100\%$$

where T_p and T_n are the number of true positives and true negatives, respectively, F_p are F_n the number of false positives and false negatives, respectively. Detection Rate is also called the recall rate, and it is defined as the number of true positives divided by the total number of samples that are accurately labeled as smoke. It reflects the ability of the detection system to find the smoke targets:

$$\text{DR} = \frac{T_p}{T_p + F_n} \times 100\%$$

Precision Rate is also called positive predictive value. It reflects the credibility of the fire alarm:

$$\text{PR} = \frac{T_p}{T_p + F_p} \times 100\%$$

In fire detection, the False Alarm Rate is an important indicator for the detection system, as in most cases there is no fire in various monitoring environments. According to experience, excessive false alarms are prominent problems in system operation tests. FAR is defined as:

$$\text{FAR} = \frac{F_p}{F_p + T_n} \times 100\%$$

4.3. Effects of Spatial–Temporal Information

First, we investigate the improvement in the smoke detection framework with spatial–temporal information extracted by 3D CNN. Table 3 shows the detection

Table 3
Results of Faster RCNN and 3D CNN

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
Faster RCNN threshold = 0.8				
Bounding box	75.71	77.96	40.09	24.76
Frame	59.34	85.97	47.89	56.84
Faster RCNN + C3D				
Bounding box	93.74	74.00	88.42	2.06
Frame	91.88	84.72	93.18	3.77

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

Table 4
Results of Training Model with Different Input Frames

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
AlexNet	80.10	89.05	46.48	21.80
GoogleNet	85.72	86.14	56.03	14.37
3DCNN	93.74	74.00	88.42	2.06

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

result of faster RCNN with threshold 0.8. Obviously, the faster RCNN model trained using a small amount of simulated smoke pictures does not perform well in our database. The AR of bounding boxes is only 75.71%, and the FAR is as high as 24.76%. If we use a simple frame alarm rule according to which one smoke box occurs in a target frame, the AR of the target frame is only 59.34%. After joining 3D CNN for clip classification, the performance of the smoke detection framework has been greatly improved.

Generally, images from different scenes need different preprocessing to generate suspected regions. We focus on the comparison of classification performance between different fire detection methods based on CNNs. Faster RCNN can be regarded as the first step in general fire detection pipeline mentioned in Sect. 1. AlexNet and GoogleNet are classic networks that commonly used in the second step for classification of suspected fire region. The detection results shown in Table 4 show more clearly that 3D CNN improves the smoke detection performance greatly with the ability to extract spatio-temporal information.

4.4. Structure of 3DCNN

As described in Sect. 3.5, smoke diffuses slowly compared with general action recognition such as playing golf and archery in the UCF101 dataset. Longer duration clips are used as network inputs to integrate more smoke motion information and ensure the detection precision rate. Tables 5 and 6 show the results of input

Table 5
Results of Fine-Tuning Model with Different Input Frames

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
Finetuning_16	92.89	75.33	82.59	3.38
Finetuning_32	92.49	64.77	89.51	1.61
Finetuning_64	93.31	67.63	92.09	1.23

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

Table 6
Results of Training Model with Different Input Frames

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
training_16	93.87	74.81	88.46	2.08
training_32	93.74	74.00	88.42	2.06
training_64	93.56	72.36	88.87	1.93

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

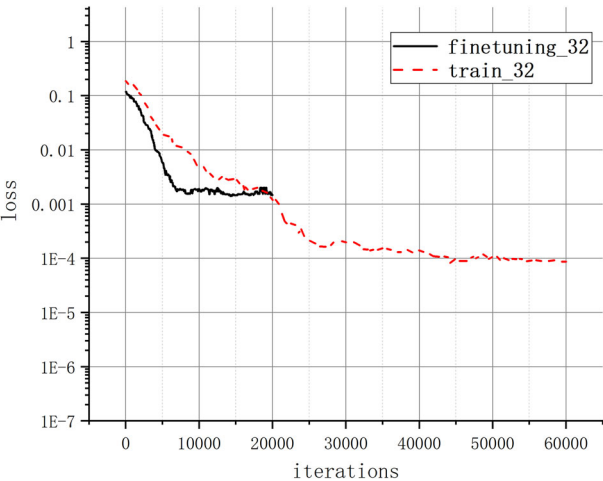


Figure 12. Losses of finetuning_32 and train_32.

clips of different lengths with fine-tuning and training. It is obvious that inputs with more frames can improve the PR and reduce the FAR prominently.

Fine-tuning refers to the process of using a network that has already been trained for a given task with large-scale training data to initialize the convolutional neural network and making small modifications to improve the convolutional neural network with training data. The pre-trained network used in this paper is trained with sportM1 [17]. Figure 12 shows the loss curves of fine-tuning

and training. It shows that fine-tuning needs 20,000 iterations only and training sustains a lower loss than fine-tuning. Fine-tuning can save training time and prevent overfitting for a small training dataset. However, it is not convenient to compare convolutional neural networks with different structures as the structure of the pre-trained network is fixed. To unify the standards, training is used in the follow-up experiments.

In addition to the adjustment of the data layer, the conv layers also have been analyzed. Actually, 8 conv layers have been used in the C3D network [17]. However, for a small training dataset, a smaller network helps to prevent overfitting with fewer parameters. The results of a large network with 8 conv layers and a small network with 5 conv layers are shown in Table 7. The input clips of both networks contain 32 frames. It is obvious that small networks perform better than large ones as our train dataset is small compared with UCF101 and sportM1. Because of the fact there is no standard dataset for video fire detection and the collected video data are far from meeting the requirement, a network with a small structure is more appropriate. With the development of AI, a VFD system should be applied more and more widely, resulting in better performance, with increasing video data and a deeper network.

In the previous section, we discussed network inputs with different numbers of frames. For input clips with 32 frames and 64 frames, we use different temporal strides to ensure the same output of pool1 layer and the network scale. Actually, there are multiple options for connectivity like Early Fusion and Slow Fusion described in [25]. We realize Slow Fusion by controlling the temporal strides of pool layers as shown in Fig. 13, in which differences from the network in Fig. 8

Table 7
Results of Networks with Different Structures

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
Large net_32	92.70	74.42	82.25	3.41
Small net_32	93.74	74.00	88.42	2.06

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

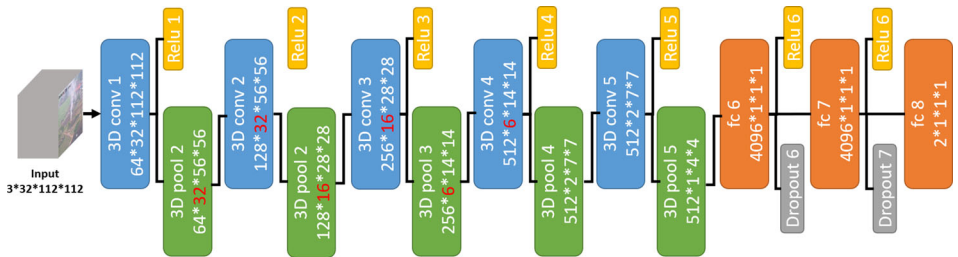


Figure 13. Realization of Slow Fusion by reducing the temporal the strides of pool layers and increasing the number of feature maps.

Table 8
Results of Early Fusion and Slow Fusion

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
Small net_32 early fusion	93.74	74.00	88.42	2.06
Small net_32 Slow Fusion	94.28	75.54	90.26	1.73

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

Table 9
Results of Different Training Data

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
Synthetic dataset	79.56	42.30	41.80	12.52
Train_32	93.74	74.00	88.42	2.06
Data augmentation	93.91	70.96	92.60	1.21
Optical flow	94.64	73.97	94.21	0.97
Background subtraction	88.16	37.96	87.36	1.17

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

are given in red numbers. The results given in Table 8 show that Slow Fusion performs better than Early Fusion. However, Slow Fusion requires a larger memory as it needs about 10G memory with a batch size of 10, and Early Fusion only needs 5G with the same batch size.

4.5. Effects of Data Process

We attempt to increase training data with synthetic smoke videos inspired by works in [16]. The synthetic dataset contains 7850 smoke clips and 6300 non-smoke clips. However, with the exception of the static shape of smoke in synthetic smoke images, synthetic smoke videos in this work have to imitate the motion of smoke. The results listed in Table 9 show that the performance of the synthetic dataset is not satisfactory. This approach requires more realistic simulation techniques.

We used optical flow and background subtraction to extract the motion of each input clip; the objective was to explicitly describe the motion between each clip frame and make the recognition easier, as the network does not need to estimate motion implicitly. The results listed in Table 8 show that 3D CNN trained on optical flow clips achieves very good performance. However, the method using background subtraction seems not to work, as the results are worse compared with the raw input.

4.6. Feature Vectors Extracted from the Middle Layer

Softmax is the most common classifier in convolutional neural network. It is a generalization of multiple classes of the binary Logistic Regression classifier and

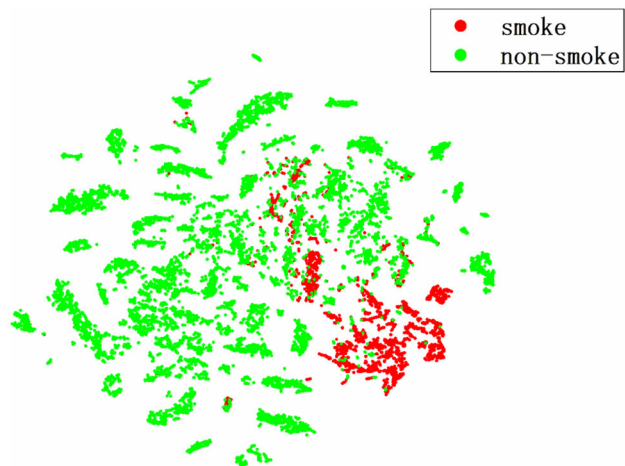


Figure 14. Visualizations of features extracted from fc7 using t-SNE [31].

**Table 10
Results of Feature Vector Trained with SVM**

Methods	AR (%)	DR (%)	PR (%)	FAR (%)
Train_32	93.74	74.00	88.42	2.06
Svm raw_fc7	93.07	65.40	93.03	1.04
Svm raw_fc7 + score	93.15	65.36	93.68	0.94
Svm raw_fc7 + opticalflow_fc7 + score	95.23	74.67	97.58	0.39

Bold values indicate the optimal value in the column, that is, the highest AR, DR, PR, or the lowest FAR

provides a probability for each class. Another popular choice is the SVM classifier, which has the small but consistent advantage of replacing the softmax layer on the same datasets as described in [30]. On the other hand, compared to the probability of each class, we are more concerned about whether it is smoke. We take the output of a late layer as the feature vector and train a separate SVM on that to replace the softmax layer. The latter layer is fc7 in this work; its feature embedding visualization is shown in Fig. 14 using t-SNE [31]. It is obvious that most smoke clip features extracted by our network are semantically separable from those of the non-smoke clips.

Further, as illustrated in Fig. 4, the result of Faster RCNN recognition on the frame can be utilized to improve the performance by combining the score and features extracted with 3D CNN. Because optical flow exhibits good performance as demonstrated in Sect. 4.4, inspired by two-stream [21], we combine the faster RCNN score and fc7 feature vectors extracted from raw and optical flow 3D CNN, respectively. This method significantly improves the AR and PR and lowers the FAR simultaneously. The results are shown in Table 10.

5. Conclusions

To utilize temporal information between video sequences to improve the performance of video smoke detection methods using convolutional neural networks, we propose a joint detection framework based on faster RCNN and 3D CNN. The faster RCNN is mainly used as a suspected smoke region proposal method that realizes smoke location and preliminary recognition. 3D CNN are used to extract temporal information. To execute the video smoke recognition task, we propose NMA to substitute for NMS in the proposal network of faster RCNN and obtain larger, non-overlapping bounding boxes. Experimental results show that 3D CNNs improves the smoke detection task significantly compared with image-based methods. Considering the structure design, the networks should not be too deep as the dataset is small to prevent overfitting. Slow Fusion is a better choice compared with Early Fusion without considering computational consumption.

Lack of training data is a limit on the development of video smoke detection, which can lead to overfitting and poor generalization. Synthetic smoke video is a novel solution, but the imitation of smoke motion is difficult to realize and leads to poor performance. Data augmentation techniques are effective methods and include horizontal flip, random crop, brightness variation and noise addition. On the other hand, preprocessing data using the optical flow method extrudes the temporal information and weakens the interference of the background. Using SVM to replace softmax and combining a faster RCNN score and feature vector extracted from the middle layer significantly improves the performance, in particular in the case of both raw and optical flow clips. Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Acknowledgements

This work was supported by the National Key Research and Development Plan under Grant No. 2016YFC0800100, Anhui Provincial Key Research and Development Plan under Grant No. 1704a0902030, and the Fundamental Research Funds for the Central Universities under Grant No. WK2320000035. The authors gratefully acknowledge all of these supports.

References

1. Çetin AE, Dimitropoulos K, Gouverneur B et al (2013) Video fire detection—review. *Digit Signal Process* 23(6):1827–1843
2. Ugur Töreyn B, Enis Cetin A (2007) Fire detection in infrared video using wavelet analysis. *Opt Eng* 46(6):7204
3. Toreyn BU, Dedeoglu Y, Cetin AE (2006) Contour based smoke detection in video using wavelets. In: *European signal processing conference 2006*. IEEE, pp 1–5
4. Yu C, Fang J, Wang J et al (2010) Video fire smoke detection using motion and color features. *Fire Technol* 46(3):651–666

5. Jia Y, Yuan J, Wang J et al (2016) A saliency-based method for early smoke detection in video sequences. *Fire Technol* 52(5):1–2
6. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324
7. Yin Z, Wan B, Yuan F et al (2017) A deep normalization and convolutional neural network for image smoke detection. *IEEE Access* 5(99):18429–18438
8. Mao W, Wang W, Dou Z et al (2018) Fire recognition based on multi-channel convolutional neural network. *Fire Technol* 54(2):531–554
9. Sharma J, Granmo OC, Goodwin M et al (2017) Deep convolutional neural networks for fire detection in images. In: Boracchi G, Iliadis L, Jayne C, Likas A (eds) *International conference on engineering applications of neural networks* Springer, Cham, pp 183–193
10. Muhammad K, Ahmad J, Mehmood I et al (2018) Convolutional neural networks based fire detection in surveillance videos. *IEEE Access* 6:18174–18183
11. Xu G, Zhang Y, Zhang Q, Lin G et al (2017) Domain adaptation from synthesis to reality in single-model detector for video smoke detection. [arXiv:1709.08142](https://arxiv.org/abs/1709.08142)
12. Dung NM, Ro S (2018) Algorithm for fire detection using a camera surveillance system. In: *Proceedings of the 2018 international conference on image and graphics processing*. ACM, pp 38–42
13. Luo Y, Zhao L, Liu P et al (2018) Fire smoke detection algorithm based on motion characteristic and convolutional neural networks. *Multimed Tools Appl* 77:15075–15092
14. Wang Z, Wang Z, Zhang H et al (2017) A novel fire detection approach based on CNN-SVM using Tensorflow. In: *International conference on intelligent computing*. Springer, Cham, pp 682–693
15. Zhang Q, Xu J, Xu L et al (2016) Deep convolutional neural networks for forest fire detection. In: *Proceedings of the 2016 international forum on management, education and information technology application*. Atlantis Press
16. Zhang QX, Lin GH, Zhang YM et al (2018) Wildland forest fire smoke detection based on faster R-CNN using synthetic smoke images. *Procedia Eng* 211:441–446
17. Du T, Bourdev L, Fergus R et al (2015) Learning spatiotemporal features with 3D convolutional networks. In: *IEEE International conference on computer vision*. IEEE, pp 4489–4497
18. Ren S, Girshick R, Girshick R et al (2015) Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell* 39(6):1137–1149
19. Hubel DH, Wiesel TN (1962) Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J Physiol* 160(1):106–154
20. Girshick R, Donahue J, Darrell T et al (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 580–587
21. Simonyan K, Zisserman A (2014) Two-stream convolutional networks for action recognition in videos. *Comput Linguist* 1(4):568–576
22. Donahue J, Hendricks LA, Rohrbach M et al (2017) Long-term recurrent convolutional networks for visual recognition and description. *IEEE Trans Pattern Anal Mach Intell* 39(4):677–691
23. Xu H, Das A, Saenko K (2017) R-C3D: region convolutional 3D network for temporal activity detection. In: *Proceedings of the IEEE international conference on computer vision*, pp 5783–5792

24. Shou Z, Wang D, Chang SF (2016) Temporal action localization in untrimmed videos via multi-stage CNNs. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1049–1058
25. Karpathy A, Toderici G, Shetty S et al (2014) Large-scale video classification with convolutional neural networks. In: IEEE conference on computer vision and pattern recognition. IEEE Computer Society, pp 1725–1732
26. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov R (2012) Improving neural networks by preventing coadaptation of feature detectors. Computing Research Repository. <http://arxiv.org/abs/1207.0580>
27. <http://signal.ee.bilkent.edu.tr/VisiFire/Demo>. Accessed 5 May 2018
28. <http://cvpr.kmu.ac.kr/>. Accessed 5 May 2018
29. Li S, Wang B, Dong R et al (2016) A novel smoke detection algorithm based on fast self-tuning background subtraction. In: Control and decision conference. IEEE, pp 3539–3543
30. Tang Y (2013) Deep learning using linear support vector machines. [arXiv:1306.0239](https://arxiv.org/abs/1306.0239)
31. van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. JMLR 9:2579–2605

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.