Multimed Tools Appl (2018) 77:15075–15092
DOI 10.1007/s11042-017-5090-2

Fire smoke detection algorithm based on motion characteristic and convolutional neural networks

Yanmin Luo[1,2] · **Liang Zhao**[1] · **Peizhong Liu**[3] · **Detian Huang**[3]

Received: 6 January 2017 / Revised: 12 June 2017 / Accepted: 8 August 2017 /
Published online: 23 August 2017

Abstract It is a challenging task to recognize smoke from visual scenes due to large variations in the feature of color, texture, shapes, etc. The current detection algorithms are mainly based on single feature or fusion of multiple static features of smoke, which leads to low detection accuracy. To solve this problem, this paper proposes a smoke detection algorithm based on the motion characteristics of smoke and the convolutional neural networks (CNN). Firstly, a moving object detection algorithm based on background dynamic update and dark channel priori is proposed to detect the suspected smoke regions. Then, the features of suspected region is extracted automatically by CNN, on that the smoke identification is performed. Compared to previous work, our algorithm improves the detection accuracy, which can reach 99% in the testing sets. For the problem that the region of smoke is relatively small in the early stage of smoke generation, the strategy of implicit enlarging the suspected regions is proposed, which improves the timeliness of smoke detection. In addition a fine-tuning method is proposed to solve the problem of scarce of data in the training network. Also, the algorithm has good smoke detection performance by testing under various video scenes.

Keywords Background dynamic update · Dark channel prior · Convolutional neural networks · Smoke detection

1 Introduction

Thousands of fires occur every day around the world which resulting in hundreds of casualties and large areas of forest vegetation damage, and serious threat to human life and property

✉ Yanmin Luo
lym@hqu.edu.cn

[1] College of Computer Science & Technology, Huaqiao University, Xiamen 361021, China

[2] College of Mechanical Engineering & Automation, Huaqiao University, Xiamen 361021, China

[3] College of Engineering, Huaqiao University, Quanzhou, Fujian 362021, China

safety and natural ecological environment. Fire is often sudden and strong, affecting a wide range, and difficult to dispose. Therefore, the real-time monitoring of fire has become particularly important. The early detection of fire is the key to reducing the loss because once the fire spread it will be difficult to control. In general, the flame is small in the early, but the smoke is very obvious, therefore the detection of smoke is an important basis to determine whether the fire occurred timely.

The traditional fire smoke detection systems rely on sensors that work only when the smoke is close to the sensor, so they are not suited to open space. In addition for the reason that the sensors are susceptible to interference from dust, airflow, and human factors, those detection systems often has high false alarm rates. With the development of the high efficiency video processing technology [26–29], video-based fire smoke detection algorithm has great application prospects.

Smoke has rich features, such as color, texture and shape irregularity, fluttering, flicker, frequency, etc. At present, video-based fire smoke detection algorithm is usually based one or more features of smoke, and make decision directly or by classifiers. Toreyin etc. [23] used motion, flicker, edge blurring and color for smoke detection. Variance of edge magnitudes was extracted for smoke detection. The application scope of the algorithm is limited due to the need to analyze the background of the integrated scene. Chen et al. [2] used a color model to extract fire and smoke pixels, on that dynamical measures of growth and disorder were extracted for verification. The false detection rate of the algorithm will become very high if the scene has fire-colored moving objects. Fujiwara et al. [6] proposed a technique for extracting smoke regions from an image using fractal encoding concept. For the low contrast or fuzzy smoke image, the extracted fractal features are not stable enough. Ko et al. [10] proposed a feature extraction method based on the spatial-temporal Bag of Features (BoF) model, which combines static and dynamic features, and then classifies it through random forest. Chenebert et al. [3] proposed a non-temporal texture driven method for fire pixel detection in video or still imagery. But the method does not utilize any temporal information. Yuan [31] presented a video smoke detection method using an accumulative motion model, and a block-based algorithm is adopted to improve the detection efficiency but it cannot detect smoke drifting in any direction. Tian [21] extracted the suspected regions by Gaussian Mixture Model (GMM), and the texture of smoke was described by NR-LBP (Non-Redundant Local Binary Pattern) feature, and then classified by support vector machine (SVM). Genovese et al. [7] proposed a method for the detection of smoke on the basis of computational intelligence techniques. The detection process focuses on the extraction of features such as the movement, color, and edge of smoke. Then, two-layer feed forward neural networks are used to estimate the areas that describe smoke regions in the different frames. Yuan [32] extracted an effective feature vector by concatenating the histogram sequences of Local Binary Pattern (LBP) and Local Binary Pattern Variance (LBPV) pyramids, and a BP neural network was used for smoke detection. This method has a high accuracy in the test sets. Yu et al. [30] proposed a video smoke detection method using color and motion features. The method cannot obtain real time detection rates due to computationally expensive optical flow. Wang [25] used a sliding time window to detect fluttering regions, extracted these features such as the fluttering direction, the periodic fluttering magnitude, the positive periodic fluttering magnitude and the reverse periodic fluttering magnitude, and finally decided whether it was smoke by a trained fuzzy neural network. By condense video Luo [17] found that smoke trajectories have some special characteristics, such as right-leaning line, smooth streamline, low-frequency, fixed source and vertical–horizontal ratio. This method is very novel, but has a poor detection effect on slowly

diffusing smoke. [20] and [5] used CNN to achieve the detection of the fire smoke or flame, but the evaluation is image-level, and [5] does not take into account the time complexity of the sliding window. Up to now, it is still challenging to detect smoke in video due to the variability of smoke characteristics.

In conclusion, these algorithms rely on features that selected and processed manually which require certain expertise and experience, and selected features have a good performance in some scenes, but may invalid in other scenes due to a wide variety of smoke. In this paper, we apply high-capacity CNN to the video smoke detection. The algorithm can not only extract smoke features automatically, but also improve the detection accuracy and other criterion obviously. The contributions of this work are summarized as below:

1) We propose algorithm of two-step for video-based fire detection, in which we use the motion characteristic to extracts suspected smoke regions that not only reduces the number of candidate areas to meet the real-time requirements, but also reduces the possibility of false alarms.
2) At present, the datasets of smoke images are incomplete and difficult to obtain, resulting in inadequate of training model to produce over-fitting. We propose to speed up the training by fine-tuning, and can get a higher accuracy rate under the same number of iterations.
3) The direct reshape of suspected region may reduce the key features of the image when the region is particularly small. We propose the strategy to implicit enlarge the suspected regions to improve the accuracy and timeliness of the detection.

The rest of this paper is organized as follows. Section 2 introduces the advantages of convolutional neural network briefly. Section 3 details the proposed method and section 4 gives the experimental results. Section 5 concludes this work and discusses the application prospects.

2 Convolutional neural networks

CNN is a kind of special deep neural network architecture which can recognize the visual pattern directly from the original image. Recently, the deep learning model based on CNN has made a great breakthrough in the fields of computer vision, such as face recognition and target tracking. As shown in Fig. 1, CNN is a multi-layer neural network composed of convolutional layer and pooling layer. This architecture is very suitable for two-dimensional image recognition. In Fig. 1, *C* refers to the convolutional layer, to achieve feature extraction; *S* refers to the pooling layer, to achieve feature mapping. The input image is extracted by the convolutional layer, and the feature map is scaled by the pooling layer (The goal is to reduce the sensitivity of the output of the feature map to translation and other forms of distortion). And so on, all the eigenvalues are connected into a one-dimensional vector finally, and input to the traditional neural network to achieve classification. CNN has the following advantages in image processing compared with traditional neural network: (1) The input image and network topology can be a good match; (2) Feature extraction and pattern classification are performed simultaneously; (3) Weight-sharing technology greatly reduces the training parameters of the network and makes the architecture of the neural network more simple and adaptable.

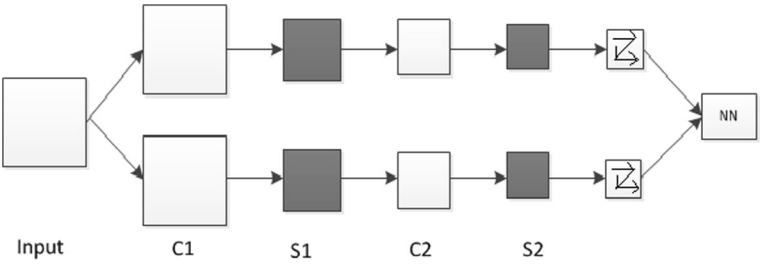


Fig. 1 The basic architecture of CNN

CNN can recognize the changeable pattern and have robustness to geometric deformation. At the same time, it can automatically extract the deeper features of image, and avoid the blindness and complexity of the traditional feature extraction algorithm. These advantages can be a good deal with the smoke large variations in the feature of color, texture, shapes, etc. Now we introduce how the algorithm uses CNN to achieve video-based fire smoke detection.

3 Smoke detection based on motion characteristic and CNN

3.1 Algorithm framework

The region of video surveillance is very wide. Smoke only cover small area of the video image in the early stage of generation, and the characteristics of smoke is not obvious enough at the same time, so the direct processing for the whole image is not conducive to smoke feature extraction, which will affecting the detection results. Therefore, we need to reduce the scope of the detection region what we called the suspected smoke regions extraction. After determining the suspected region, the feature vectors of suspected region is extracted by high-capacity CNN, and smoke detection is implemented finally. The algorithm is divided into two modules, and the overall framework of the algorithm is shown in Fig. 2. In Fig. 2, Conv1 represents the convolution layer, and the numbers represent the index of convolution layer; Pool1 represents the pooling layer, and the numbers represent the index of pooling layer; FC represents the fully connected layer, and the numbers represent the index of fully connected layer. And other symbols have similar meanings.

3.2 Suspected smoke regions extraction

A variety of recent papers offer methods for generating category-independent region in the field of target detection. For example: selective search [24], constrained parametric min-cuts

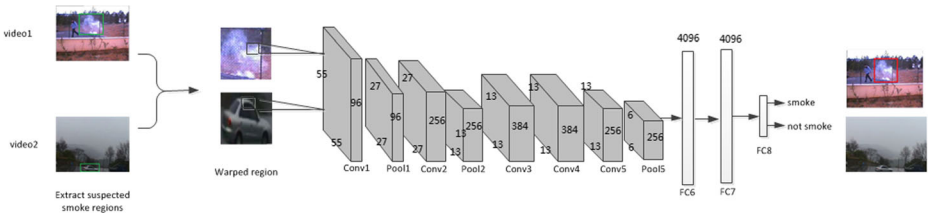


Fig. 2 Smoke detection framework

(CPMC) [1], etc. But the smoke is not the same as the general target, which does not have obvious color and contour features, and we do not need to extract the suspected regions for each frame because of the probability of fire is very small in the actual scene, so these algorithms are not suitable for the suspected regions extraction of smoke. Taking into account the motion characteristics of smoke, we propose a motion detection method to extract suspected smoke regions.

Under the conditions of static monitoring equipment, the common motion detection methods are frame difference method [16], Gaussian mixture method [22] and optical flow method [11]. These methods have a good effect on the detection of rigid objects, but for the detection of smoke which is non-rigid and it's movement pattern presents diffusion, it's easy to produce hollow phenomenon, therefore it is difficult to extract the complete smoke region. Moreover, the swinging branches, pedestrians, cars and other moving objects are very frequent in the natural scene. The foreground pixels generated by these moving objects may be extracted as suspected smoke regions if without special treatment, which not only affects the efficiency of the algorithm, but also improves the probability of false alarm to a certain extent.

In this paper, a moving object detection algorithm based on background dynamic update and dark channel priori [15] is used to extract suspected regions. The algorithm can not only extract the complete smoke region, but also filter some interference pixels generated by some common moving objects. The steps of the algorithm can be described as:

1) **Background update;** Background estimation is the core of background update. Smoke is diffusion to the periphery, so the gray value inside the smoke region of the adjacent frame changes little which leads to the hollow phenomenon of the traditional methods. So for the smoke detection, background updates would consider both the next frame and the current frame, while adding the original frame as an update reference. The background estimate is expressed as formula 1.

$$B_{n+1}(x,y)=\begin{cases} a^{*}B_n(x,y)+\beta^{*}F_{n+1}(x,y) \\ \quad +(1-\alpha-\beta)^{*}B_1(x,y) & if\,|F_{n+1}(x,y)-F_n(x,y)|>0 \\ B_n(x,y) & else \end{cases} \tag{1}$$

where n and $n+1$ represent the current frame index and the next frame index respectively, $B_n(x, y)$ and $B_{n+1}(x, y)$ represent the gray value of the current background image at (x, y) and the gray value of the estimated background image at (x, y) respectively, $F_{n+1}(x, y)$ is the gray value of the next frame at (x, y), $B_1(x, y)$ is the gray value of the original frame at (x, y), α and β are the weight coefficients, and $\alpha+\beta<1$.

2) **Motion foreground acquisition;** After the background update, we can get the foreground image by calculating the difference of the current frame and background. Formula 2 shows the calculation.

$$G_{n+1}(x,y)=\begin{cases} 255 & if\,|F_{n+1}(x,y)-B_{n+1}(x,y)|>T \\ 0 & else \end{cases} \tag{2}$$

where T and $G_{n+1}(x, y)$ represent the threshold value and the gray value of the foreground image at (x, y) respectively.

3) **Extracted suspected smoke regions;** First, the dark channel image of the current frame needs to be calculated. The dark channel image is defined as:

$$J^{dark}(x) = \min_{c\in(r,g,b)}\left(\min_{y\in\Omega(x)}(J^c(y))\right) \tag{3}$$

where J^c is the gray value of one of the color channels, and $\Omega(x)$ is a window centered on x. After obtaining the dark channel image of the current frame, we select the suitable threshold according to the dark channel characteristics of smoke, and calculate the dark channel threshold image by two valued processing. After that the foreground pixels of interference objects can be eliminate by comparing the foreground image and G(x, y). Finally we can get suspected smoke regions through a series of morphological transformation. Figure 3 shows the process of extracting suspected smoke regions. It can be seen that the algorithm Can eliminate the interference caused by general non-smoke object general objects well. But for the smoke-like moving objects such as pedestrian with white clothes, the silver car, lighting and etc., it needs further judgment.

3.3 Smoke recognition based on CNN

3.3.1 Network architecture

There are numerous variants of CNN architectures in the literature. However, their basic components are very similar. Our proposed network architecture is used for feature extraction and classification of smoke. As showed in Fig. 2, The network has eight layers in addition to the input layer, and without taking into account the strict distinction between the convolutional layers and the pooling layers, it consists of 5 convolutional layers and 3 fully connected layers. Our choice of a lean network design is motivated both from our desire to reduce the risk of overfitting as well as achieving high accuracy just by the simple network. The network is described in detail as follows:

- **Input layer;** In order to performing multi-layer operation for the input image, the size of input layer image for network architecture is fixed at 227 × 227 pixels. For the color image contains three color channels in usual, the total size of the input image is 227× 227 × 3.

Fig. 3 Extracted suspected regions (**a**. The current frame; **b**. The foreground image; **c**. The dark channel threshold image after a series of processing; **d**. Suspected smoke regions)

- **Convolutional layer;** The convolutional layer aims to learn feature representations of the inputs. This layer is composed of several convolution kernels which are used to compute different feature mappings. The size of the convolution kernel determines the size of the output feature map. After the convolution operation of the kernels, the size of feature map N can be calculated by formula 4.

$$\begin{cases} N_x^l = \dfrac{N_x^{l-1} - K_x^l + 2P_x^l}{S_x^l} \\ N_y^l = \dfrac{N_y^{l-1} - K_y^l + 2P_y^l}{S_y^l} \end{cases} \tag{4}$$

where l is the current layer index, K is the size of filter, P is the amount of fill pixels, and S is the step length. The convolution operation is followed by a non-linear transformation of the activation function. Here the *Relu* function [12] is the activation function, which is showed in formula 5.

$$Relu(x) = max(0, x) \tag{5}$$

when x is greater than 0, the derivative is equal to 1. Therefore it is very good to carry out the error back propagation, which can greatly reduce the training time. The error of the back propagation, greatly reducing the training time. Then, the computation expression for a neural of the convolutional layer can be expressed by formula 6.

$$x_j^l = Relu\left(\sum_{i \in M_j} x^{l-1} w_{ij}^l + b_j^l\right) \tag{6}$$

where M is the size of filter, w and b represent the connection weight vector and the bias term respectively.

- **Pooling layer;** The purpose of pooling is to reduce the number of neurons while maintaining the invariance of features for scale changes. Usually the pooling layer is placed between two convolutional layers. Each feature map of a pooling layer is connected to its corresponding feature map of the preceding convolutional layer. In this paper, the network conducts pooling operations after the first, third, and fifth convolutional layer. The pooling operation can divide the feature map into several rectangular regions, and then each region can be operated accordingly. The operation of pooling including maximum pooling and mean pooling, we choose the maximum pooling in our network, which can be expressed by formula 7.

$$\mathrm{y} = \max(x_i), x_i \in x \tag{7}$$

where x is an area of the feature map and x_i is the output of the neurons in the region.

- **Fully connected layer;** The fully connected layers aim to perform high level reasoning. They take all neurons in the previous layer and connect them to every single neuron of current layer to generate global semantic information. The network in this paper consists of three fully connected layers. The first fully-connected layer that receives the output of the

fifth convolutional layer and contains 4096 neurons, followed by a *Relu* and a dropout layer. The second fully connected layer that receives the 4096-dimensional output of the first fully connected layer and again contains 4096 neurons, followed by a *Relu* and a dropout layer. The last fully connected layer contains two neurons which maps to the final classes for smoke and non-smoke.

The network parameter setting is shown in Table 1.

Finally, the output of the last fully connected layer is fed to a softmax function [9] which assigns a probability for each class. The prediction itself is made by taking the class with the maximal probability for the given test image.

3.3.2 Network training

The process of training the network is to solve the parameters $\theta = \{W, B\}$. This is achieved through minimizing the loss of training set. Given a set of smoke and non-smoke images x_i and their actual category y_i (0/1), we use cross-entropy of softmax as the loss function [4] which is showed in formula 8.

$$L(\theta) = -\frac{1}{N}\sum_i \log[Softmax(\alpha_k)], i = 0, 1, \cdots, N-1 \tag{8}$$

where k is the actual label value, N is batch size.

The loss is minimized using stochastic gradient descent (SGD) with the standard back-propagation [13]. The other parameters required for training are shown in Table 2, and the corresponding weight matrices are updated by formula 9.

$$\begin{cases} v_{t+1} = 0.9^* v_t - 0.01^* \varepsilon^* \nabla L(W_t) \\ W_{t+1} = W_t + v_{t+1} \end{cases} \tag{9}$$

where t is the iteration index, v is the momentum variable, ε is the learning rate, $\nabla L(W_t)$ is the derivative.

Training deep convolutional neural networks needs to solve a large number of parameters, so the training requires a sufficient number of data set. The scarcity of the training set may lead to inadequate learning of the network and slow of convergence. The existing smoke data sets is small because it is difficult to obtain, so the network parameters initialized by drawing randomly from a Gaussian distribution with zero mean, it will perhaps lead to the above

Table 1 Network settings

Type	Feature map size	Kernel size	Stride
Conv1	55*55*96	11*11	4
Poling1	27*27*96	3*3	2
Conv2	27*27*256	5*5	1
Poling2	13*13*256	3*3	2
Conv3	13*13*384	3*3	1
Conv4	13*13*384	3*3	1
Conv5	13*13*256	3*3	1
Poling5	6*6*256	3*3	2
FC6	Neurons:4096		
FC7	Neurons:4096		
FC8	Neurons:2		

Table 2 Solver settings

Solver type	base_Ir	gamma	momentum	batch_size
SGD	0.01	0.1	0.9	256

problems. In this paper, we use fine-tuning method to alleviate these problems. Aside from replacing the CNN's ImageNet- specific 1000-way classification layer with a randomly initialized 2-way classification layer, the CNN architecture is unchanged. In this way, in addition to the last classification layer of the network parameters need to be initialized randomly, the other layer parameters are initialized by the corresponding pre-training model parameters. For the problem of scarce of smoke data, since the pre-training network model has certain feature extraction ability, so it can speed up convergence.

Aside from our use of lean network architecture, we apply two additional methods to further limit the risk of overfitting. First we apply dropout learning [14] (randomly setting the output value of network neurons to zero). The network uses the dropout method in FC6 and FC7 with a dropout ratio of 0.5 (50% chance of setting a neuron's output value to zero), thus reducing the dependency between fully connection, and improving the generalization ability of features. Second, we use data augmentation by taking a random crop of 227 × 227 pixels from the 256 × 256 input image and randomly mirror it in each forward-backward training pass. This, similarly to the multiple crop and mirror variations used by [18].

3.3.3 Smoke detection

Warped region Smoke generation is a continuous process of diffusion, so the size of smoke region will changed all the time. In order to compute CNN features for a suspected smoke regions, we must first convert the image data in that region into a form that is compatible with the CNN (its architecture requires inputs of a fixed 227 × 227 pixel size). The usual practice is to perform interpolation scaling [8]. Scaling can be divided into two forms, one is to keep the aspect ratio of the region, and then padding the pixels to the required size, as shown in Fig. 4(a). The other is regardless of the size or aspect ratio of the region, we warp all pixels in a tight bounding

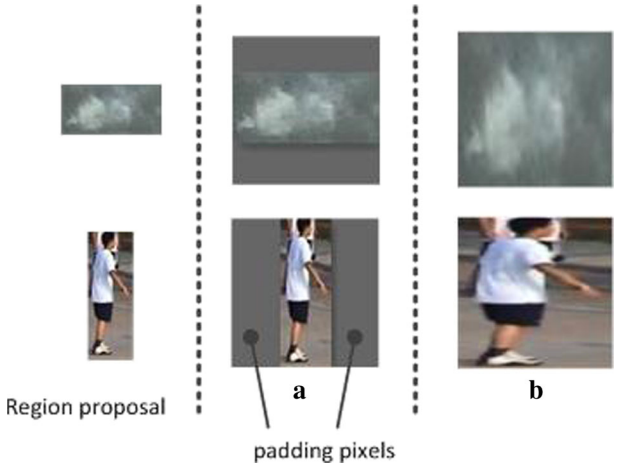


Fig. 4 Scaling forms

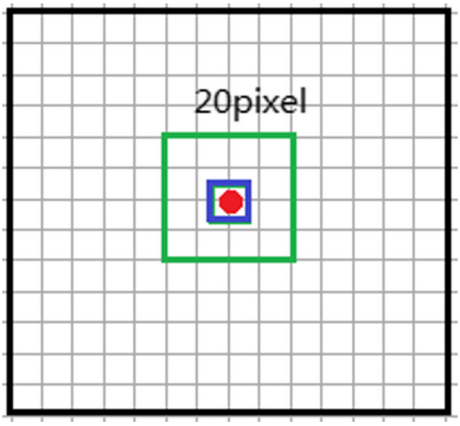


Fig. 5 Implicit enlarge the suspected region

box around it to the required size, as shown in Fig. 4(b). Because the smoke shape is not fixed and other features such as color, texture, is relatively uniform in the same image, and arbitrary scaling does not changes the inherent features of the smoke, so we opt the last simple scaling form.

Implicit enlarge the suspected regions When the early stage of smoke generation or other moving objects is particularly small, similar to the red area of Fig. 5 (the number of foreground pixels detected is less than 400 in our paper), and the suspected region of extraction will be small, similar to the blue box in Fig. 5. If we still opt to scaling directly in the suspected region, it will easy to cause false alarm because of too many pixels to be inserted. Of course, most other algorithms choose to ignore when the region is small, and wait until the smoke diffusions to a sufficient size for processing. Considering the importance of early detection of fire, we propose a method of implicit enlarging the suspected regions, which is centered on the original region, similar to the green box in Fig. 5. Through several experiments in the video scene, it is found that when the region of implicit is opted to 20 × 20 pixels for

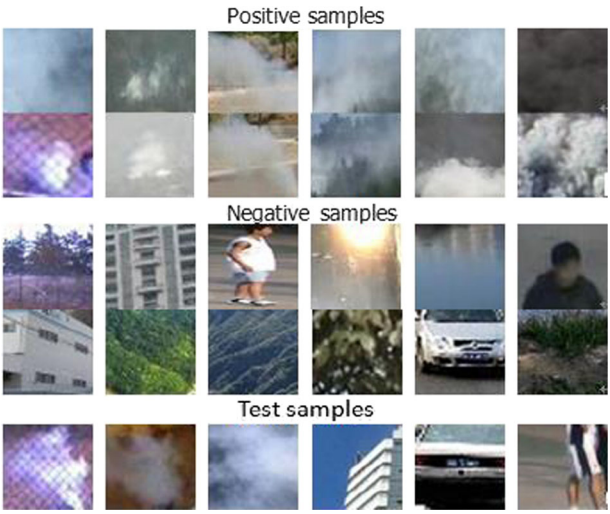


Fig. 6 Part of the samples in the datasets

Table 3 Experimental environment

Hardware environment	Software environment
GPU:NVIDIA Tesla K40C Memory:12GB	OS:Linux Platform:Caffe

scaling which is best for subsequent feature extraction and classification. This approach combines with high-capacity convolutional neural networks to ensure that the smoke can be detected in a timely and accurate.

Feature extraction and classification The suspected regions need to be filtered to reduce the effects of noise before pass through the input layer of the network. Then the multi-layer feature extraction and mapping are realized by the forward propagation of the convolutional layer and the fully connected layer of the network. Thus, our arbitrary-shaped suspected region can get a one-dimensional eigenvector of length 4096 through the network to realize the feature automatic extraction. The network uses the softmax function to realize feature classification after the last fully connected. The softmax function can be used to calculate the probability that the eigenvectors belong to each class. The probability vector $<P_0, P_1>$ can be obtained in our algorithm (P_0 represents the probability that the suspected region belongs to the non-smoke region, and P_1 represents the probability of belonging to the smoke region). The final classification results C can be obtained through the formula 10. And on that the detection of smoke can be done ultimately.

$$C = \underset{i}{argmax}(p_i), i = 0, 1 \tag{10}$$

4 Experimental evaluations

4.1 Experimental data

The datasets used in the experiment is partly from Yuan (http://staff.ustc.edu.cn/~yfn/vsd.html), and the rest from the network collection and the actual shooting. The datasets contain a training images and two testing images, and named Train1, Test1 and Test2 respectively. The size of the image is not fixed which the pixels of one side are between 30 ~ 150. Train1 contains 6776 smoke images and 12,352 non-

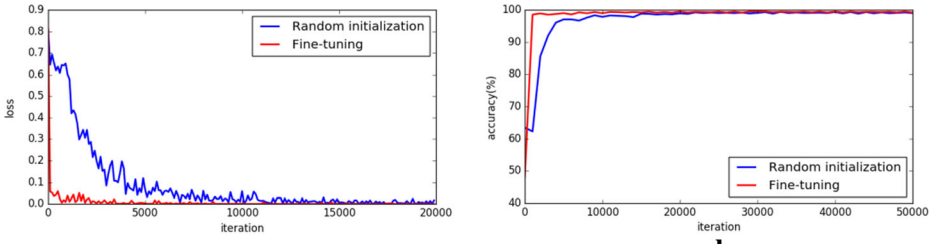


Fig. 7 Experimental comparison of Random Initialization and Fine-tuning (**a**. Loss curve; **b**. Accuracy curve)

Table 4 Experimental results on the testing sets

Testing sets	Detection rate		False alarm rate		Accuracy	
	Yuan	CNN	Yuan	CNN	Yuan	CNN
Test1(1548 smoke and 1880 non-smoke images)	0.882429 (1366 smoke images are detected)	0.998708 (1546 smoke images are detected)	0.013298 (25 non-smoke images are misclassified)	0.003191 (6 non-smoke images are misclassified)	0.939615	0.997666
Test2(1872 smoke and 3862 non-smoke images)	0.901175 (1687 smoke images are detected)	0.992521 (1858 smoke images are detected)	0.050492 (195 non-smoke images are misclassified)	0.010357 (40 non-smoke images are misclassified)	0.933729	0.990582

smoke images, totaling 19,128 positive and negative samples. Test1 contains 1548 smoke images and 1880 non-smoke images, totaling 3428 positive and negative samples. Test2 contains 1872 smoke images and 3962 non-smoke images, totaling 5834 positive and negative samples. The distribution of datasets is randomly selected, some of which are shown in Fig. 6.

4.2 Experimental of datasets

The software and hardware environment used in the experiment is shown in Table 3.The classification results of the model in the test set will directly affect the smoke detection performance under the video condition, so we use following criterion for evaluate a smoke detector in image-level:

$$Detection \quad rate = \frac{TP}{N_{pos}} \tag{11}$$

$$False \quad alarm \quad rate = \frac{FN}{N_{neg}} \tag{12}$$

$$Accuracy = \frac{TP + TN}{N_{pos} + N_{neg}} \tag{13}$$

where TP is the number of truth positives, i.e. the number of smoke images which are classified as non-smoke, TN is the number of truth negatives, i.e. the number of non-smoke images which are classified as non-smoke, FN is the number of false negatives, i.e.the number of non-smoke images which are classified as smoke. N_{pos} and N_{neg} are the number of positive samples and the number of negative samples in the set.

Experiments were conducted by random initialization and fine-tuning of parameters to train and test the model. Figure 7(a) shows the relationship between the number of iterations and the

Fig. 8 Detection results in video scenes

Table 5 Smoke detection performance comparisons on smoke videos

Video	Description	Duration	Detection at frame #	
			Yuan	CNN
Video 1	Rapid diffusion of smoke in field	626	53	43
Video 2	Slow diffusion of thin smoke	5956	552	531
Video 3	Slow diffusion of smoke and the distance is very far	6042	1120	976
Video 4	Rapid diffusion of smoke in the sunlight	240	107	71
Video 5	Rapid diffusion of dense smoke on low hill	2324	86	57

value of the loss during training, we can see that the fine-tuning method converges quickly which is stable after about 3000 iterations basically. Correspondingly, the random initialization method converges after 15,000 iterations. Figure 7(b) shows the relationship between the number of iterations and the accuracy of the model in the validation set, we can see that a high accuracy can be achieved with less number of iterations by fine-tuning which can over 98% after about 1000 iterations. This greatly alleviates the scarcity of smoke data when training network. Also, the accuracy of fine-tuning method is higher than the random initialization in the same iterations.

In order to compare the performance of convolutional neural networks and traditional method in feature extraction and classification, we apply Yuan's algorithm [32] to our datasets for comparison. Yuan used three-level pyramid texture model to extract region features, and a BP neural network was used for smoke detection. Table 4 lists the results of the two algorithms in Test1, Test2 (CNN: The test results for a network model that iterated 50,000 times). We can see that the CNN has obvious improvement on the performance criterions of the testing sets, which solves the problem that the detection accuracy is not high due to large variations in the feature of smoke for the traditional methods.

The experiment also tried the GoogleNet [19] under the same conditions for training and testing, which have achieved good results in various competitions of recent years. The average accuracy of GoogleNet can reach 99.24% in the testing sets, which is equivalent to the 8-layers network used in this paper. This result shows that the lean network architecture can already extract the smoke features well. Considering the network complexity, time efficiency and other factors, we adopt 8-layers CNN to realize smoke detection in this paper.

4.3 Experimental of videos

The algorithm is also tested under various video scenes. Videos from Korea CVPR Laboratory (http://cvpr.kmu.ac.kr/) and Bilkent University Laboratory (http://signal.ee.bilkent.edu.tr/VisiFire/Demo/SampleClips.html) respectively, where video 1 ~ 5 contains smoke, video

Table 6 Smoke detection performance comparisons on non-smoke videos

Video	Description	Duration	Number of false alarms	
			Yuan	CNN
Video 6	Rainy days,a white car passing by	7310	0	0
Video 7	Very strong car lights in the night	155	153	0
Video 8	The branches swing sharply in the strong wind	2807	0	0

6 ~ 8 is non-smoke video. The detection results obtained by our algorithm are shown in Fig. 8. It can be seen that the proposed algorithm has good performance under various scenes.

The timeliness of detection and the low false alarm rate are important criterion for evaluating video fire smoke detection. Table 5 and Table 6 describe the detection results of our algorithm and Yuan's algorithm in the video scene. From the results of video 1 ~ 5, it can be seen that the shape and density of smoke have little effect on our algorithm, which indicating that the extracted CNN features can describe the essential characteristics of smoke well. At the same time, it can be seen from video 3 that the extracted CNN features has scale invariance and the method of implicit enlarge the suspected region to ensure that the smoke can be detected in the early stage of generation. From the results of video 7, we can see that Yuan's algorithm is poor for recognition the light of driving car, which will continue to produce false alarm. Correspondingly, our algorithm does not appear. From Video 6 and Video 8, we can see that rainy or windy weather have little effect on our algorithm and Yuan's algorithm. To sum up, our algorithm gets better performance than Yuan's algorithm in actual smoke detection. However, our algorithm is slower than traditional algorithms in the speed of processing. The average frame rate can reach 6–7 fps for video with the pixel size of 320 × 240.

5 Conclusions

Smoke has very large variations in color, texture and shapes, so it is still a challenging task to accurately recognize smoke from visual scenes. In this paper, we successfully applies convolutional neural networks to fire smoke detection, which solves the limitation of smoke feature extraction by manual design and greatly improve the accuracy of detection. The strategy of implicit enlarges the suspected smoke regions, which improve the timeliness of the detection. In addition, we show that it is highly effective to fine-tune the network for the smoke detection where data is scarce. At the same time, we have established a larger datasets on an existing basis. With the development of the deep learning and hardware, the network architecture tends to be miniaturized while assuring the classification accuracy, our algorithm has a good application prospects.

Acknowledegments This work was supported by the Talent project of Huaqiao University (No. 14BS215) and Quanzhou scientific and technological planning projects of Fujian, China (2015Z40, 2015Z120).

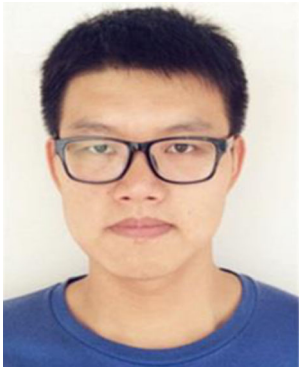
References

1. Carreira J, Sminchisescu C (2012) CPMC: automatic object segmentation using constrained parametric min-cuts. IEEE Trans Pattern Anal Mach Intell 34(7):1312–1328
2. Chen T, Wu P, Chiou Y (2004) An early fire-detection method based on image processing. Proc Int Conf on Image Proc 4:1707–1710
3. Chenebert A, Breckon T P, Gaszczak A (2011) A non-temporal texture driven approach to real-time fire detection. In: 2011 18th IEEE International Conference on Image Processing, 263(4). IEEE, Brussels, pp 1741–1744
4. Dong C, Loy CC, He K, Tang X (2016) Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell 38(2):295–307

5. Frizzi S, Kaabi R, Bouchouicha M, Ginoux JM, Moreau E, Fnaiech F (2016) Convolutional neural network for video fire and smoke detection. In: Proceedings of the IEEE conference on Industrial Electronics Society. IECON, Florence. pp 877–882
6. Fujiwara N, Terada K (2004) Extraction of a smoke region using fractal coding. IEEE International Symposium on Communications and Information Technology, 2004(2). IEEE, Sapporo, pp 659-662
7. Genovese A, Labati RD, Piuri V, Scotti F (2011) Wildfire smoke detection using computational intelligence techniques. In: IEEE International Conference on Computational Intelligence for Measurement Systems & Applications. IEEE, Ottawa. 43(4), pp 1–6
8. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE Computer Society, Washington DC, pp 580–587
9. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: Convolutional architecture for fast feature embedding. In: proceedings of the 22nd ACM international conference on Multimedia. ACM, New York, pp 675–678
10. Ko BC, Park JO, Nam JY (2013) Spatiotemporal bag-of-features for early wildfire smoke detection. Image Vis Comput 31(10):786–795
11. Kolesov I, Karasev P, Tannenbaum A, Haber E (2010) Fire and smoke detection in video with optimal mass transport based optical flow andneural networks. In: Proceedings of International Conference on Image Processing, 119(5). IEEE Computer Society Press, Hong Kong, pp 761-764
12. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Proces Syst 25(2):1097–1105
13. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
14. Levi G, Hassner T (2015) Age and gender classification using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015. IEEE, Boston, pp 34–42
15. Liang Zhao, Yan-Min Luo, Xiang-Yu Luo (2017) Based on dynamic backgroundupdate and dark channel prior offire smoke detection algorithm. Application Research of Computes 34(2):957–960
16. Luo S (2013) State-of-art of video based smoke detection algorithms. J Image Graph 18(10):1225–1236
17. Luo S, Yan C, Wu K, Zheng J (2015) Smoke detection based on condensed image. Fire Saf J 75(2015):23–35
18. Sun Y, Wang X, Tang X (2014) Deep learning face representation from predicting 10,000 classes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014. IEEE, Columbus, pp 1891–1898
19. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S , Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference oncomputer vision and pattern recognition, 2015. IEEE, Boston, pp 1–9
20. Tao C, Zhang J, Wang P (2016) Smoke detection based on deep convolutional neural networks. In: Proceedings of the IEEE conference on Industrial Informatics-Computing Technology, Intelligent Technology (ICIICII), 2016. IEEE, Wuhan, pp 150–153
21. Tian H, Li W, Ogunbona P, Nguyen D T (2011) Smoke detection in videos using non-redundant local binary pattern-based features. In: IEEE International Workshop on Multimedia Signal Processing, 2011. IEEE, Hangzhou, pp 1–4
22. Tian H, Li W, Wang L, Ogunbona P (2012) A novel video-based smoke detection method using image separation. In: Proceedings of International Conference on Multimedia and Expo. IEEE Press, Melbourne. pp 532–537
23. Toreyin BU, Dedeoglu Y, Cetin AE (2015) Wavelet based real-time smoke detection in video. In: Proceedings of 13th European Signal Processing Conference, Antalya, 2015. pp 1-4
24. Uijlings JR, Sande KE, Gevers T, Smeulders AW (2013) Selective search for object recognition. Int J Comput Vis 104(2):154–171
25. Wang T, Liu Y, Xie ZP (2011) Flutter analysis based video smoke detection. J Electron Inf Technol 35(5): 1024–1029
26. Yan C, Zhang Y, Xu J, Dai F, Li L, Dai Q, Wu F (2014) A highly parallel framework for HEVC coding unit partitioning tree decision on many-core processors. IEEE Signal Process Lett 21(5):573–576
27. Yan C, Zhang Y, Xu J, Dai F, Zhang J, Dai Q, Wu F (2014) Efficient parallel framework for HEVC motion estimation on many-core processors. IEEE Trans Circuits Syst Video Technol 24(12):2077–2089
28. Yan C, Zhang Y, Dai F, Wang X (2014) Parallel deblocking filter for HEVC on many-core processor. Electronics Letters 50(5):367–368
29. Yan C, Zhang Y, Dai F, Zhang J, Li L, Dai Q (2014) Efficient parallel HEVC intra-prediction on many-core processor. Electron Lett 50(11):805–806

30. Yu C, Fang J, Wang J, Wang Y (2010) Video fire Smokedetection using motion and color features. Fire Technol 46(3):651–663
31. Yuan FN (2008) A fast accumulative motion orientation model based on integral image for video smoke detection. Pattern Recogn Lett 29(7):925–932
32. Yuan FN (2011) Video-based smoke detection with histogram sequence of LBP and LBPV pyramids. Fire Saf J 46(3):132–139

Yanmin Luo (1974.11-) received PhD Degree from the Cognitive Science Department of Xiamen University. Now, he is an associate professor of College of Computer Science & Technology, Huaqiao University, China. His research interests are machine learning, computing intelligent and pattern recognition. Email: lym@hqu.edu.cn



Liang Zhao is studying for M.S. Degree in computer science and technology institute, Huaqiao University, Xiamen, China. His main research interests include image processing, computer vision and deep learning, etc.

Peizhong Liu he was born in 1976.He received the P.H. degree from school of information science and engineering, Xiamen University, Xiamen, Fujian, China. Now he is a IOT Experimentalist, and his research interests include multi-dimensional space biomimetic informatics, visual media retrieval, network model, information security.

Detian Huang (1985.02-) received B. Sc. Degree from Xiamen University in 2008, received PhD Degree from University of Chinese Academy of Sciences in 2013. Now, he is a teacher in Huaqiao University. His research interests are image enhancement, image restoration, and machine learning. Email: huangdetian@sina.com