



Video Fire Smoke Detection Using Motion and Color Features

Yu Chunyu, Fang Jun, Wang Jinjun and Zhang Yongming, State Key Laboratory of Fire Science, USTC, Number 96 Jin Zhai Road, Hefei, Anhui, China*
e-mail: ycyu@mail.ustc.edu.cn; fangjun@ustc.edu.cn; wangjinj@ustc.edu.cn

Received: 9 July 2009/**Accepted:** 29 September 2009

Abstract. A novel video smoke detection method using both color and motion features is presented. The result of optical flow is assumed to be an approximation of motion field. Background estimation and color-based decision rule are used to determine candidate smoke regions. The Lucas Kanade optical flow algorithm is proposed to calculate the optical flow of candidate regions. And the motion features are calculated from the optical flow results and use to differentiate smoke from some other moving objects. Finally, a back-propagation neural network is used to classify the smoke features from non-fire smoke features. Experiments show that the algorithm is significant for improving the accuracy of video smoke detection and reducing false alarms.

Keywords: Video smoke detection, Fire detection, Motion features, Optical flow, Neural network

1. Introduction

Conventional point-type thermal and smoke detectors are widely used nowadays, but they typically take charge of a limited area in space. In large rooms and high buildings, it may take a long time for smoke particles and heat to reach a detector.

Video-based fire detection (VFD) is a newly developed technique in the last few years, and it can greatly serve the fire detection requirement in large rooms and high buildings, and even outdoor environment.

Researchers all over the world have done a lot of work on this new technique. Up to now, most of methods make use of the visual features of fire including color, textures, geometry, flickering and motion. Early studies began with video flame detection using color information. Yamagishi and Yamaguchi [1, 2] presented a flame detection algorithm based on the spatio-temporal fluctuation data of the flame contour and used color information to segment flame regions. Noda et al. [3] developed a fire detection method based on gray scale images. They analyzed the relationship between temperature and RGB pixel channels, and used the gray level histogram features to recognize fire in tunnels. Phillips et al. [4]. used the Gaussian-smoothed color histogram to generate a color lookup table of fire flame pixel and then took advantage of temporal variation of pixel values to

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

determine whether it was a fire pixel or not. Shuenn-Jyi [5] used clustering algorithm to detect the fire flame. However, these methods using pixel information cannot segment fire pixels very well when there are objects having the same color distribution as fire. Although color features can be acquired by learning, false segmentation still exists inevitably. Ugur Toreyin et al. [6] synthetically utilized motion, flicker, edge blurring and color features for video flame detection. In their study temporal and spatial wavelet transform were performed to extract the characteristics of flicker and edge blurring.

But as we know, most fires start at the smouldering phase in which smoke usually appears before flame. And in these cases smoke detection gives an earlier fire alarm. But compared to flame, the visual characteristics of smoke such as color and grads are less trenchancy, so that smoke is harder to be differentiated from its disturbances. So the extraction of smoke's visual features becomes more complicated. Researchers began to use different features for the study of video smoke detection in recent years. Toreyin et al. [7] used partially transparent feature of smoke, and implemented by extracted the edge blurring value of the background object in wavelet domain. Vicente [8] clustered motions of smoke on a fractal curve, and presented an automatic system for early forest fire smoke detection. In Xiong's [9] study, they thought that smoke and flames were both turbulent phenomena, the shape complexity of turbulent phenomena might be characterized by a dimensionless edge/area or surface/volume measure. Yuan [10] used a accumulated model of block motion orientation to realize real-time smoke detection, and his model can mostly eliminate the disturbance of an artificial lights and non-smoke moving objects. Cui [11] combined tree-structured wavelet transform and gray level co-occurrence matrices to analyze the texture feature of fire smoke, but real-time detection was not considered.

The proposed algorithm uses both color and motion features, and the combination of the two features will greatly enhance the smoke detection reliability. Color features are extracted using a color-decision rule and motion features are extracted with optical flow which is an important technique in motion analysis for machine vision. Section 2 describes smoke feature extraction. In Sect. 3, a back-propagation neural network is used to learn and classify the statistic of the smoke features from non-fire smoke features. In Sect. 4, several experiments are described and the results are discussed. In the last section, this paper is concluded.

2. Smoke Feature Extraction

2.1. Background Estimation

First, moving pixels and regions are extracted from the image. They are determined by using a background estimation method developed by Collins et al. [12]. In this method, a background image B_{n+1} at time instant $n + 1$ is recursively estimated from the image frame I_n and the background image B_n of the video as follows:

$$B_{n+1}(x, y) = \begin{cases} aB_n(x, y) + (1 - a)I_n(x, y) & (x, y) \text{ stationary} \\ B_n(x, y) & (x, y) \text{ moving} \end{cases} \quad (1)$$

where $I_n(x, y)$ represents a pixel in the n th video frame I_n , and a is a parameter between 0 and 1. Moving pixels are determined by subtracting the current image from the background image.

$$X(x, y) = \begin{cases} 1 & \text{if } |I_n(x, y) - B_n(x, y)| > T \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

T is a threshold which is set according to the scene of the background.

2.2. Color-Based Decision Rule

The moving pixels $X(x, y)$ are further checked using a color based decision rule which is based on the studies developed by Chen [13]. He thinks smoke usually displays grayish colors, and the condition $R \pm a = G \pm a = B \pm a$ and with I_{int} (intensity) component of HIS color model $K_1 \leq I_{int} \leq K_2$. K_1 and K_2 are thresholds used to determine smoke pixel. The rule implies that three components R, G, and B of smoke pixels are equal or so.

Here we made a small modification. The decision function for smoke recognition is that for a pixel point (i, j) ,

$$m = \max\{R(i, j), G(i, j), B(i, j)\} \quad (3)$$

$$n = \min\{R(i, j), G(i, j), B(i, j)\} \quad (4)$$

$$I_{int} = \frac{1}{3}(R(i, j), G(i, j), B(i, j)) \quad (5)$$

If the pixel $X(x, y)$ satisfies both the conditions $m - n < a$ and $K_1 \leq I_{int} \leq K_2$ and at the same time, then $X(x, y)$ is considered as a smoke pixel, otherwise $X(x, y)$ is not a smoke pixel.

The typical value a ranges from 15 to 20 and dark-gray and light-gray smoke pixel threshold ranges from 80 (D_1) to 150 (D_2) and 150 (L_1) to 220 (L_2), respectively according to Thou-Ho Chen's study. D_1 , D_2 , L_1 and L_2 are special values of K_1 and K_2 . The pixels that pass the color decision rule are set as candidate regions and are further checked by calculating the motion features through the optical flow computation.

2.3. Lucas Kanade

2.3.1. Brightness Constancy Assumption. Horn and Schunck's algorithm [14] is the base of all various optical flow calculation methods. They defined a brightness

constancy assumption which describes as follows. If the motion is relatively small and illumination of the scene maintains uniform in space and steady over time, it can be assumed that the brightness of a particular point remains relatively constant during the motion. It is established as

$$I(x + u\delta t, y + v\delta t, t + \delta t) = I(x, y, t) \quad (6)$$

From a Taylor expansion of 4, the gradient constraint equation is derived:

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0, \quad (7)$$

which is the main optical flow constraint equation. $\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$ are quantities observed from the image sequence, and (u, v) are to be calculated. Apparently Eq. 7 is not sufficient to determine the two unknowns in (u, v) , which is known as the “aperture problem”, additional constraints are needed.

In order to solve the aperture problem, different optical flow techniques appeared, like differential, matching, energy-based and phase-based methods. Of these different techniques on the sequences Barron’s group [15] found that Lucas and Kanade [16] method was one of the most reliable.

2.3.2. Lucas Kanade Optical Flow Algorithm. Based upon the optical flow Eq. 7, Lucas and Kanade introduced an additional constraint needed for the optical flow estimation. Their solution assumes a locally constant flow that (u, v) is constant in a small neighbourhood Ω . Within this neighbourhood the following term is minimized

$$\sum_{(x,y) \in \Omega} W^2(\mathbf{x})(I_x u + I_y v + I_t)^2 \quad (8)$$

Here, $W(x)$ is a weighting function that favors the center part of Ω . The solution to (8) is given by

$$\mathbf{A}^T \mathbf{W}^2 \mathbf{A} \mathbf{v} = \mathbf{A}^T \mathbf{W}^2 \mathbf{b} \quad (9)$$

where, for n points $x_i \in \Omega$ at a single time t ,

$$\begin{aligned} \mathbf{A} &= [\nabla \mathbf{I}(\mathbf{x}_1), \dots, \nabla \mathbf{I}(\mathbf{x}_n)]^T, \\ \mathbf{W} &= \text{diag}[W(\mathbf{x}_1), \dots, W(\mathbf{x}_n)], \\ \mathbf{b} &= -(I_t(\mathbf{x}_1), \dots, I_t(\mathbf{x}_n))^T. \end{aligned} \quad (10)$$

And when $\mathbf{A}^T \mathbf{W}^2 \mathbf{A}$ is nonsingular, a closed form solution is obtained as:

$$\mathbf{v} = [\mathbf{A}^T \mathbf{W}^2 \mathbf{A}]^{-1} \mathbf{A}^T \mathbf{W}^2 \mathbf{b} \quad (11)$$

where

$$\mathbf{A}^T \mathbf{W}^2 \mathbf{A} = \begin{bmatrix} \sum W^2(\mathbf{x}) I_x^2(\mathbf{x}) & \sum W^2(\mathbf{x}) I_x(\mathbf{x}) I_y(\mathbf{x}) \\ \sum W^2(\mathbf{x}) I_y(\mathbf{x}) I_x(\mathbf{x}) & \sum W^2(\mathbf{x}) I_y^2(\mathbf{x}) \end{bmatrix} \quad (12)$$

All sums are taken over points in the neighbourhood Ω . After the calculation of the optical flow of each feature point, four statistical characteristics are considered: the averages and variations of the optical flow velocity and the orientation. The optical flow computation results are $\{d_i | d_i = [d_{xi}, d_{yi}]^T, i = 0, 1, \dots, n\}$, and n is the pixel number of the candidate regions that pass the color decision rule. The motion feature extraction is performed as follows.

$$\text{Average of velocity: } a_n = \frac{1}{n} \sum_{i=1}^n \sqrt{d_{xi}^2 + d_{yi}^2} \quad (13)$$

$$\text{Variation of velocity: } b_n = \frac{1}{n-1} \sum_{i=1}^n \left(\sqrt{d_{xi}^2 + d_{yi}^2} - a_n \right)^2 \quad (14)$$

$$\text{Average of orientation: } c_n = \frac{1}{n} \sum_{i=1}^n e_i \quad (15)$$

$$\text{Variation of orientation: } d_n = \frac{1}{n-1} \sum_{i=1}^n (e_i - c_n)^2 \quad (16)$$

while

$$e_i = \begin{cases} \arctan\left(\frac{d_{yi}}{d_{xi}}\right), & (d_{xi} > 0, d_{yi} > 0) \\ \pi - \arctan\left(\frac{d_{yi}}{d_{xi}}\right), & (d_{xi} > 0, d_{yi} < 0) \\ \pi + \arctan\left(\frac{d_{yi}}{d_{xi}}\right), & (d_{xi} < 0, d_{yi} < 0) \\ 2\pi - \arctan\left(\frac{d_{yi}}{d_{xi}}\right), & (d_{xi} < 0, d_{yi} > 0) \end{cases} \quad (17)$$

As an important physical phenomenon of fire, the smoke turbulent is a random motion which has abundant size and shape variation. If we consider the smoke is made up of lots of spots, as a result of the turbulent movement, the spots' velocity vectors will have irregular distributions. That's why the variations of the optical flow velocity field are used. Because of the buoyancy effect, the smoke plume movement is generally moving upward slowly if there is no large ventilation and airflow. So the average of the optical flow velocity field will fall into an interval in

a fixed scene. The four statistic values are further processed by using neural network to decide whether they are smoke features.

3. Smoke Feature Classification

A Back-Propagation Neural Network is used to discriminate the smoke features. In Lippmann [17], he represented steps of the back-propagation training algorithm and explanation. The back-propagation training algorithm is an iterative gradient designed to minimize the mean square error between the actual output of multi-layer feed forward perception and the desired output. It requires continuous differentiable non-linearity. The input of the network has three real units and the output has one ranging from 0 to 1. Set the input $x_i (i = 1, 2, 3, 4)$, the desired output is y . The derivative $F(x)$ is computed in two phases:

3.1. Feed Forward

This function updates the output value for each neuron. In this function, the first step is to initialize weights and biases. Set all weights and node offsets to small random values.

Next, starting with the first hidden layer, it takes the input to each neuron and finds the output by calculating the weighted sum of inputs. Consider the network with link weights w_{ij} , biases θ_i , and neuron activation functions $f_i(in_i)$:

$$u_i = f_i(in_i) \quad in_i = \sum_{j=1}^M w_{ij}u_j + \theta_i \quad (18)$$

in_i is the input into i -th neuron and M is the number of links connecting with i -th neuron.

Calculate actual output, use the sigmoid non linearity to calculate output y ,

$$u = \sum_{k=1}^N (w_k u_k + \theta) \quad y = f(u) \quad (19)$$

3.2. Back-Propagation

The network is run backwards. The function adapts weights. Calculate the error between output value from the calculation and the desired output.

$$E = \frac{1}{2} |y_o - y_i|^2 \quad (20)$$

Use a recursive algorithm to adjust weights. Starting at the output nodes and working back to the first hidden layer. Adjust weights by

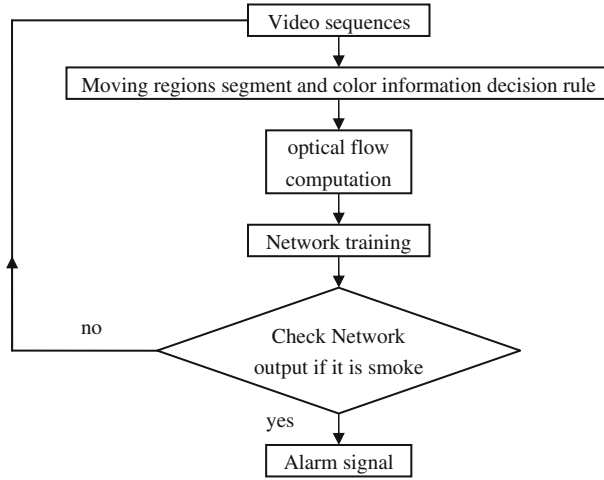


Figure 1. Flow chart of video smoke detection.

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_j x'_i \quad (21)$$

In this equation $w_{ij}(t)$ is the weight from hidden node i or from an input to node j at time t , w_j is either the output of node i or is an input, η is a gain term, and δ_j is an error term for node j , x'_i is the input value of node i . If node j is an internal hidden node, then

$$\delta_j = x'_j(1 - x'_j) \sum_k \delta_j^m w_{jk} \quad (22)$$

where k is over all nodes in the layers above node j . Internal node thresholds are adapted in a similar way by assuming they are connection weights on links from auxiliary constant-valued inputs. If node j is an output node, then calculate the error with function. If the error is smaller than a very small value like 0.00001, the recursive step is over, or adjust weights and back to the feed forward step.

The BP neural network consists of the input layer, output layer and one or more hidden layers. Each layer of BP includes one or more neurons that are directionally linked with the neurons from the previous and the next layer. In our work, we choose to use 4-5-5-1 layers, as there are four input feature values and only one output. The output layer uses a log-sigmoid transfer function, so the output of network is constrained between 0 and 1.

The sigmoid function is defined by the expression

$$f(x) = \frac{1}{1 + e^{-cx}} \quad (23)$$

The constant c in the paper is 1 arbitrarily. If the result is smoke, the y is set to be 1, or else is 0.

By using the Neural Network, the fire smoke can be recognized and detected in real-time. As shown in Figure 1, the video sequences which are collected from CCTV systems are processed as follows: First, background estimation and color-based decision rule are used to determine candidate smoke regions. Then, the Lucas Kanade optical flow computation is proposed to calculate the motion feature based on optical flow. At last, a back-propagation neural network is used to classify and recognize the smoke features and give out fire alarm signal accordingly.

4. Results and Discussion

The algorithm presented in this paper is implemented using Visual C++ and Open CV library. The algorithm can detect fire smoke in real time.

Figure 2 is a white smoke picture that has just passed the color-decision rule. Comparing (c) with (b), it can be seen that the color-based decision rule could remove most of none smoke pixel regions (a) walking person in Figure 2 and keep smoke regions at the same time.

Figure 3 shows results of a video sequence that a person wearing grayish color clothes. It can be clearly seen that some non-smoke regions with similar smoke pixels are wrongly extracted. The wrongly extracted regions here will be further checked by the optical flow features.

The optical flow computation is done to the candidate regions determined by color-decision rule. Some results are shown in Figure 4. For the increasing of vision effect, the results are indicated by arrows, the starting points of which are the corresponding points in the previous image. And the optical flow velocity direction is indicated by the direction of the arrows. The scalar value of the optical flow velocity is not expressed in the figure. For the best visual effect, the length of the arrows in the figure is normalized to 10 pixels.

The four statistic values computed from the optical flow results are further processed by neural network to classify and decide whether they are smoke features.

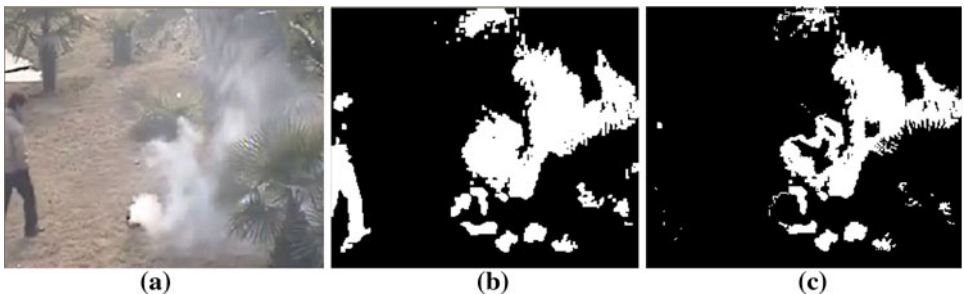


Figure 2. Results of color-decision rule. (a) Original image. (b) Background estimation results. (c) Color-decision rule results.

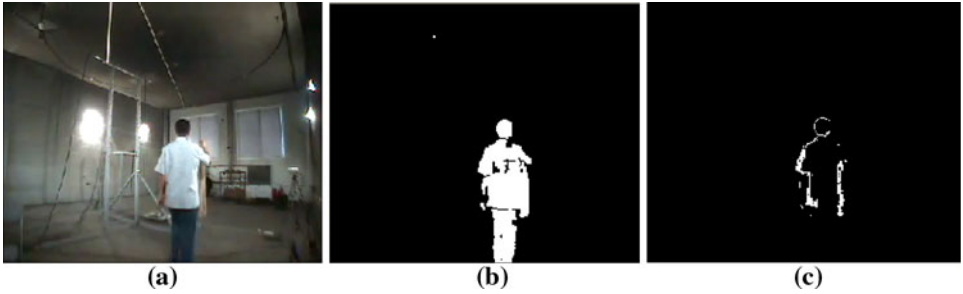


Figure 3. Results of color-decision rule. (a) Original image. (b) Background estimation results. (c) Color-decision rule results.

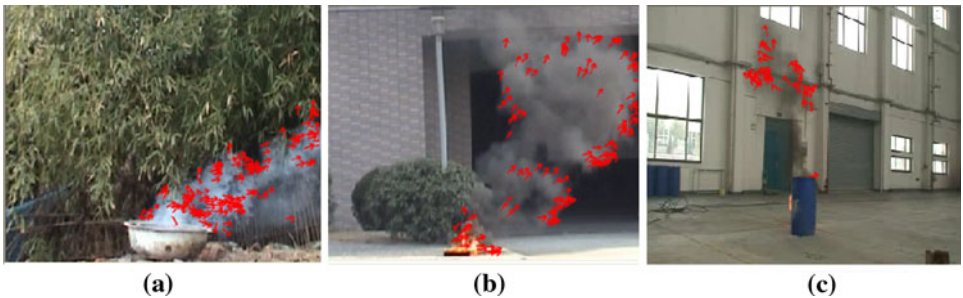


Figure 4. Extraction results of optical flow. (a) White smoke 1. (b) Black smoke 1. (c) Black smoke 2.

15 smoke and none smoke videos were used to train the neural network. The training process of neural network will not be discussed here. A discrimination example of smoke video and none smoke video (a walking person wearing grayish color clothes) is performed, and the output of the neural network is shown in Figure 5. For simplicity, a threshold is used to segment the output value to determine whether it is smoke or not. As shown Figure 5, most smoke output values are above 0.6, and at the same time none-smoke values are below 0.3. After a series of performance tests, it is found that the threshold is better to set as 0.5.

The performance of the proposed method is compared with that of Toreyin's [7] algorithm using eight fire smoke video sequences and seven non-smoke video sequences from <http://signal.ee.bilkent.edu.tr/VisiFire/Demo/SampleClips.html>, plus additional test data from our own video library. The scenes of the chosen video sequences are shown in Figures 6 and 7. All videos are normalized to 25 Hz and 320×240 pixels which are the same with the videos downloaded from the websites above.

As shown in Table 1, detection at frame# means that after the fire is started at 0th frame, the smoke is detected at frame#. The proposed method has a better performance than Toreyin's method when using movie 1 to 6 except movie 4. The smoke color in movie 4 is less trenchancy compared with background, and the

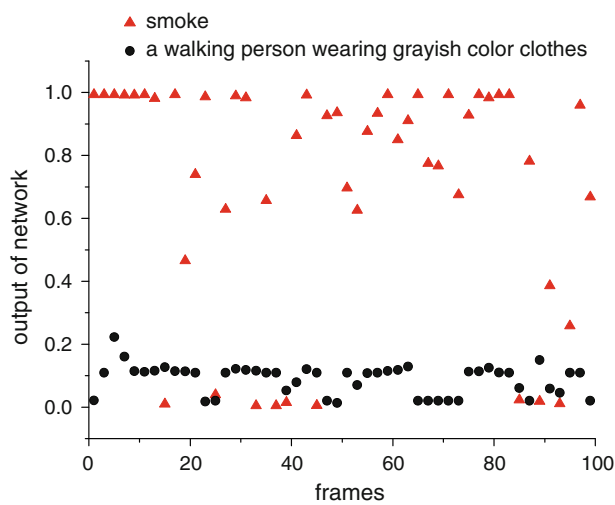


Figure 5. Output example of neural network.



Figure 6. Scenes of smoke videos.

blurring phenomena in the background edges which is described in Toreyin’s literature is especially obvious. So when the Toreyin’s method is applied to scenes like movie 5, it may have a better performance than the proposed method.

Typically, black smoke is produced by flaming fire and has a higher temperature than that of white smoke, so black smoke has stronger buoyancy and has an

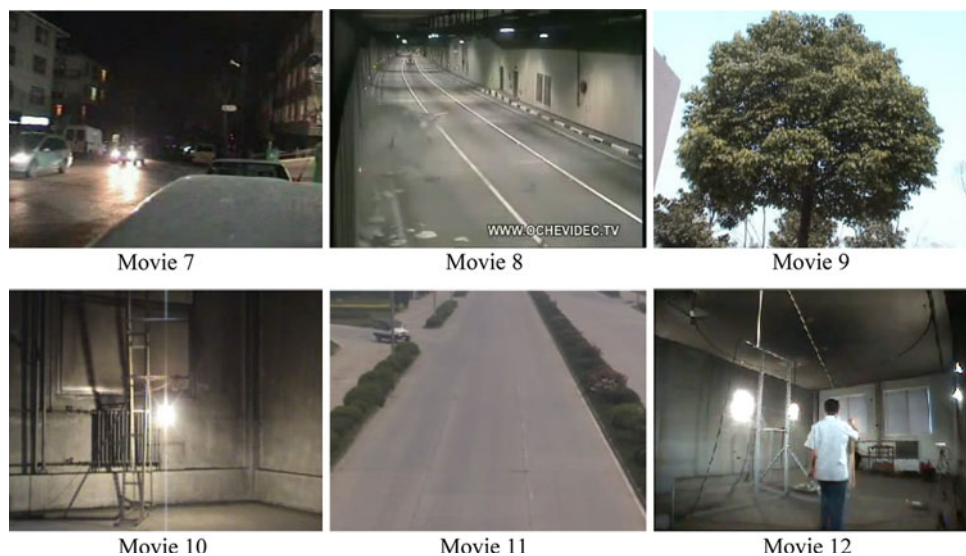


Figure 7. Scenes of non-smoke videos.

Table 1
Smoke Detection Performance Comparison of the Proposed Method and Toreyin's Method

Video sequences	Duration (frames)	Detection at frame#		Description
		Proposed method	Toreyin's method	
Movie 1	630	118	132	Smoke behind the fence
Movie 2	240	121	127	Smoke behind window
Movie 3	900	86	98	Smoke beside waste basket
Movie 4	2200	69	23	Cotton smoke beside wall
Movie 5	500	15	167	Black smoke of outdoor fire
Movie 6	1100	27	221	Black smoke in large room

obvious plume movement. So the proposed method using motion features has a much better performance when applied to black smoke scenes like movie 7 and 8.

Six non-smoke movies are used to test the false alarm rate of the proposed method and the results are compared with that of Toreyin's. Number of false alarms means that the method gives how many false alarms after tested all the frames of the movie. As shown in Table 2, the proposed method gives fewer false alarms and has a lower false alarm rate. In movie 8 and 12, there are some regions which have similar color distributions with smoke. But at the same time the velocity fields of the regions are different from that of smoke. So the proposed method does not give false alarms. In movie 10 and 11, reflection of the walls and moving cars cause disturbances to smoke velocity field.

Table 2
False Alarm Performance Comparison of the Proposed Method
and Toreyin's Method

Video sequences	Duration (frames)	Number of false alarms		Description
		Proposed method	Toreyin's method	
Movie 7	150	0	0	Car lights in the night
Movie 8	100	0	1	Tunnel accident 1
Movie 9	890	0	0	Waving leaves
Movie 10	1132	3	4	Moving lights
Movie 11	1550	2	1	Moving cars
Movie 12	1500	0	4	Grayish color clothes

Experiments show that the proposed method can extract motion features that distinguish smoke videos from none smoke videos. The algorithm is significant for improving the accuracy of smoke detection.

5. Conclusion and Outlook

In this paper, a smoke video detection method using optical flow computation and color-decision rule is proposed. This method can distinguish the disturbances which having the same color distribution as smoke, such as car lights, because of the using of motion features. Both motion and color information are involved which greatly improve the reliability of video smoke detection.

Experimental results show that the proposed method can distinguish smoke videos from none smoke videos and have a remarkable accuracy. And the proposed method has a better performance than Toreyin's method especially when applied to black smoke detection.

As shown in Table 2, the proposed method can give right alarm most of time, but there does exist some false alarm. This is because of the using of neural network. The results depend too much on the selected statistical values for training. If the selection of statistical values is not appropriate, the results' accuracy may be not very encouraging. This can be advanced by doing more work on training the neural network.

References

1. Yamagishi H, Yamaguchi J (1999) Fire flame detection algorithm using a color camera. MHS '99. In: Proceedings of 1999 international symposium on micromechatronics and human science, pp 255–260, 23–26, November 1999
2. Yamagishi H, Yamaguchi J (2000) A contour fluctuation data processing method for fire flame detection using a color camera. In: IEEE 26th annual conference on IECON of the industrial electronics society, vol 2, pp 824–829, 22–28, October 2000

3. Noda S, Ueda K (1994) Fire detection in tunnels using an image processing method. In: Proceedings of vehicle navigation and information systems conference, pp 57–62, 31 August–2, September 1994
4. Phillips W III, Shah M, Da Vitoria Lobo N (2000) Flame recognition in video. In: Fifth IEEE workshop on applications of computer vision, pp 224–229, 4–6, December 2000
5. Wang S-J, Tsai M-T, Ho Y-K, Chiang C-C (2006, 12) Video-based early flame detection for vessels by using the fuzzy color clustering algorithm. International computer symposium (ICS2006), III, 1179–1184
6. Ugur Toreyin B, Dedeoglu Y et al (2006) Computer vision based method for real-time fire, flame detection. *Pattern Recog Lett* 27(1):49–58
7. Ugur Toreyin B, Dedeoglu Y, Cetin EA (2005) Wavelet based real-time smoke detection in video. In: 13th European signal process conference EUSIPCO2005, Antalya, Turkey
8. Vicente J, Guillemant P (2002) An image processing technique for automatically detecting forest fire. *Internat J Therm Sci* 41:1113–1120
9. Xiong Z, Caballero R, Wang H, Alan MF, Muhidin AL, Peng P-Y (2007) Video-based smoke detection: possibilities, techniques, and challenges. In: IFPA, fire suppression and detection research and applications—a technical working conference (SUPDET), Orlando, FL
10. Yuan F (2008) A fast accumulative motion orientation model based on integral image for video smoke detection. *Pattern Recog Lett* 29(7):925–932
11. Cui Y, Dong H, Zhou E (2008) An early fire detection method based on smoke texture analysis and discrimination. In: Proceedings of the 2008 congress on image and signal processing, vol 3, CISP'08, pp 95–99
12. Collins RT, Lipton AJ, Kanade T (1999) A system for video surveillance and monitoring. In: The proceeding of American nuclear society (ANS) eighth international topical meeting on robotics and remote systems, Pittsburgh, PA
13. Chen T, Wu P, Chiou Y (2004) An early fire-detection method based on image processing. In Proceeding of IEEE ICIP '04, pp 1707–1710
14. Horn BKP, Schunck BG (1981) Determining optical flow. *AI* 17:185–304
15. Barron JL, Fleet DJ, Beauchemin S (1994) Performance of optical flow techniques. *Int J Comput Vis* 12(1):43–77
16. Lucas B, Kanade T (1981) An iterative image registration technique with an application to stereo vision. In: Proceeding of DARPA IU Workshop, pp 121–130
17. Lippmann RP (1987) An introduction to computing with neural network. *IEEE ASSP Mag* 4:4–22