



Fire Detection in Video Using LMS Based Active Learning

Osman Günay*, Kasım Taşdemir, B. Uğur Töreysin and A. Enis Çetin, Department of Electrical and Electronics Engineering, Bilkent University, 06800 Bilkent, Ankara, Turkey
e-mail: tasdemir@ee.bilkent.edu.tr; bugur@ee.bilkent.edu.tr; enis@ee.bilkent.edu.tr

Received: 6 February 2009/Accepted: 21 August 2009

Abstract. In this paper, a video based algorithm for fire and flame detection is developed. In addition to ordinary motion and color clues, flame flicker is distinguished from motion of flame colored moving objects using Markov models. Irregular nature of flame boundaries is detected by performing temporal wavelet analysis using Hidden Markov Models as well. Color variations in fire is detected by computing the spatial wavelet transform of moving fire-colored regions. Boundary of flames are represented in wavelet domain and irregular nature of the boundaries of fire regions is also used as an indication of the flame flicker. Decisions from sub-algorithms are linearly combined using an adaptive active fusion method. The main detection algorithm is composed of four sub-algorithms (i) detection of fire colored moving objects, (ii) temporal, and (iii) spatial wavelet analysis for flicker detection and (iv) contour analysis of fire colored region boundaries. Each algorithm yields a continuous decision value as a real number in the range $[-1, 1]$ at every image frame of a video sequence. Decision values from sub-algorithms are fused using an adaptive algorithm in which weights are updated using the least mean square (LMS) method in the training (learning) stage.

Keywords: Fire detection, Least-mean-square methods, Active learning, Decision Fusion, On-line learning

1. Introduction

Conventional chemical point fire sensors typically detect the presence of certain particles generated by smoke and fire by ionisation or photometry. An important weakness of point detectors is that they are distance limited and fail in open or large spaces because it may take a long time for particles to reach them.

The main advantage of using video in fire detection is the ability to monitor large and open spaces. Fire and flame detection algorithm in [1] is based on the use of color and motion information in video. In this paper, we not only detect fire and flame colored moving regions but also analyze the motion. It is well-known that turbulent flames flicker. Therefore, fire detection scheme can be made

* Correspondence should be addressed to: Osman Günay, E-mail: osman@ee.bilkent.edu.tr

more robust by detecting irregular rapid movements in flame colored moving pixels compared to existing fire detection systems described in [1].

High frequency motion around the boundary of an object with frequency greater than 0.5 Hz is an important sign of presence of flames in the scene. In this paper, high-frequency analysis of moving pixels is carried out in wavelet domain. Wavelet transform is a time-frequency analysis tool and one can examine an entire frequency band in wavelet domain [2, 4, 5]. Hence, it is ideally suited to determine an increase in high-frequency activity in some fire and flame colored objects. In addition, turbulent high-frequency behavior exist not only on the boundary but also inside a fire region. Spatial wavelet analysis makes is used to detect high-frequency behavior inside a flame region.

Two similar methods for identifying flame in video are described in [1, 6]. The method in [6] only makes use of the color information. On the other hand, the scheme in [1] is based on detecting the fire colored regions in the current video first. If these fire colored regions move then they are marked as possible regions of fire in the scene monitored by a camera. In [7] another real-time fire-detection system is proposed that combines foreground object information with color pixel statistics of flames. Some of the previous works [8–11] include fire detection algorithms that use temporal and spatial wavelet analysis of the video in a Hidden Markov Models framework to determine the existence of fire. In this paper, we use an LMS based on-line learning algorithm to combine the decisions of sub-algorithms, obtained using wavelet analysis and markov models, in an efficient manner.

Moving objects are determined using a background subtraction algorithm and fire colored moving objects are determined using Hidden Markov Models. Temporal and spatial wavelet analysis are carried out on flame boundaries and inside the fire region. An increase in energy of wavelet coefficients indicate an increase in high frequency activity. Contours of moving objects are also analyzed by estimating the boundaries of moving fire colored regions in each image frame. This spatial domain clue is also combined with temporal clues to reach a final decision.

The proposed automatic video based fire detection algorithm is based on four sub-algorithms: (i) detection of fire colored moving objects, (ii) temporal and (iii) spatial wavelet analysis for flicker detection, and (iv) contour analysis of flame boundaries. Each sub-algorithm separately decides on the existence of fire in the viewing range of the camera. Decisions from sub-algorithms are linearly combined using an adaptive active fusion method. Initial weights of the sub-algorithms are determined from actual fire videos. They are updated using the least-mean-square (LMS) algorithm during initial installation [12]. The error function in the LMS adaptation is defined as the difference between the overall decision of the compound algorithm and the decision of an oracle. The oracle can be the user of the program or the security guard monitoring the computer screen in the case of a surveillance application. The system asks the oracle to verify its decision whenever an alarm occurs. In this way, the user actively participates in the learning process.

The paper is organized as follows: Sect. 2 describes each one of the four sub-algorithms which make up the compound (main) fire detection algorithm. Adaptive active fusion method is described in Sect. 3. In Sect. 4, experimental results based on test fires are presented.

2. Sub-Algorithms of Flame Detection Algorithm

Flame detection algorithm is developed to locate flame regions within the viewing range of visible range camera. Four sub-algorithms that make up the composite detection algorithm are: (i) detection of fire colored moving objects, performing (ii) temporal wavelet analysis, and (iii) spatial wavelet analysis, and (iv) contour analysis of flame boundaries. The respective decision functions, $D_1(x, n)$, $D_2(x, n)$, $D_3(x, n)$ and $D_4(x, n)$, are defined, for each pixel at location x of every incoming image frame at time step n .

The outputs of decision functions D_i , $i = 1, \dots, M$ of sub-algorithms are zero-mean real numbers for each incoming sample x . When the number is positive (negative), it means that the individual algorithm has decided that there is (not) fire in the viewing range of the camera. The confidence level of each sub-algorithm is determined by the amplitude of the decision value. Higher the value, the more confident the algorithm.

2.1. Detection of Flame Colored Moving Objects

2.1.1. Moving Region Detection. Moving object segmentation for surveillance applications can be achieved using background subtraction. There are several methods in the literature [13–15]. In Video Surveillance and Monitoring (VSAM) Project at Carnegie Mellon University [13] a recursive background estimation method was developed from the actual image data using ℓ_1 -norm based calculations. In this method background is obtained by updating each pixel independently using a simple IIR filter and an adaptive threshold update algorithm is used to determine foreground and background pixels.

Background can be defined as temporally stationary part of the video, therefore stationary pixels in the video are the pixels of the background scene. If the scene is observed for some time, then pixels forming the entire background scene can be estimated because moving regions and objects occupy only some parts of the scene in a typical image of a video. A simple approach to estimate the background is to average the observed image frames of the video. Since moving objects and regions occupy only a part of the image, they conceal a part of the background scene and their effect is canceled over time by averaging. Our main concern is real-time performance of the system.

Let $I(x, n)$ represent the intensity value of the pixel at location x in the n -th video frame I . Estimated background intensity value, $B(x, n + 1)$, at the same pixel position is calculated as follows:

$$B(x, n + 1) = \begin{cases} aB(x, n) + (1 - a)I(x, n) & \text{if } x \text{ is stationary} \\ B(x, n) & \text{if } x \text{ is a moving pixel} \end{cases} \quad (1)$$

where $B(x, n)$ is the previous estimate of the background intensity value at the same pixel position. The update parameter a is a positive real number close to one. Initially, $B(x, 0)$ is set to the first image frame $I(x, 0)$. A pixel positioned at x is assumed to be moving if:

$$|I(x, n) - I(x, n - 1)| > T(x, n) \quad (2)$$

where $I(x, n - 1)$ is the intensity value of the pixel at location x in the $(n - 1)$ -th video frame I and $T(x, n)$ is a recursively updated threshold at each frame n , describing a statistically significant intensity change at pixel position x :

$$T(x, n + 1) = \begin{cases} aT(x, n) + (1 - a)(c|I(x, n) - B(x, n)|) & \text{if } x \text{ is stationary} \\ T(x, n) & \text{if } x \text{ is a moving pixel} \end{cases} \quad (3)$$

where c is a real number greater than one and the update parameter a is a positive number close to one. Initial threshold values are set to a pre-determined non-zero value.

Both the background image $B(x, n)$ and the threshold image $T(x, n)$ are statistical blue prints of the pixel intensities observed from the sequence of images $\{I(x, k)\}$ for $k < n$. The background image $B(x, n)$ is analogous to a local temporal average of intensity values, and $T(x, n)$ is analogous to c times the local temporal standard deviation of intensity in ℓ_1 -norm [13].

As it can be seen from Equation 3, the higher the parameter c , higher the threshold or lower the sensitivity of detection scheme. It is assumed that regions significantly different from the background are moving regions. Estimated background image is subtracted from the current image to detect moving regions which corresponds to the set of pixels satisfying:

$$|I(x, n) - B(x, n)| > T(x, n) \quad (4)$$

are determined. These pixels are grouped into connected regions (blobs) and labeled by using a two-level connected component labeling algorithm [16]. The output of the first step of the algorithm is a binary pixel map that indicates whether or not the pixel at location x in n th frame is moving.

2.1.2. Detection of Flame Colored Pixels. Markov models shown in Figure 1 are used to detect flame in color video. Two models are trained off-line for both flame and non-flame pixels. States of the Markov models are determined according to color information. The fire and flame color model of [11] is used for defining the flame-pixels. Although there are various types of fires, fire flames especially in the initial stages of the fire exhibit a color range of red to yellow. In terms of RGB values, this fact corresponds to the following inter-relation between R, G and B color channels: $R > G$ and $G > B$. The combined condition for the fire region in the captured image is $R > G > B$. Besides, R should be more stressed than the other components, because R becomes the dominating color channel in an RGB image of flames. This imposes another condition for R as to be over some pre-determined threshold, R_T . However, lighting conditions in the background may adversely affect the saturation values of flames resulting in similar R, G and B

values which may cause non-flame pixels to be considered as flame colored. Therefore, saturation values of the pixels under consideration should also be over some threshold value. All of these conditions are summarized in the following composite condition:

- Condition1: $R > R_T$
- Condition2: $R > G > B$
- Condition3: $S > (255 - R) * S_T/R_T$

where S_T is the value of saturation when the value of R channel is R_T . If both of the three conditions are satisfied for a pixel, then that pixel is considered as fire colored. As it is known, the saturation will decrease with increasing R value. This is formulated in the term $(255 - R) * S_T/R_T$. In fire color classification, both values of R_T and S_T are defined according to various experimental results, and typical values range from 40 to 60 and 170 to 190, for S_T and R_T , respectively.

The three-state Markov model used for flame detection is presented in Figure 1. The state $F1$ corresponds to a pixel having a fire color. The state $F2$ also corresponds to a pixel having a fire color but the fire color range of $F2$ is different from $F1$. The state called as Out is reserved for non-fire colored pixels. Let $S_x(i)$ be the state of the pixel x at frame i . The conditions for transition between states for x is shown as a flow chart in Figure 2. In this diagram, $R_x(i)$ corresponds to the R channel value of the pixel x at frame i , and $S'_x(i)$ is $F2$ if $S_x(i)$ is $F1$ and vice versa.

The thresholds should satisfy $T1 < T2$ and they are set to 10 and 40, respectively, in our implementation. Transition between states $F1$ and $F2$ occurs when there is a relatively large variation in the R channel of the fire colored pixel. When this variation is above the larger threshold $T2$, a transition to state Out takes place. The state of the pixel is preserved whenever a variation in the R channel of the fire colored pixel, which is smaller than $T1$, is observed.

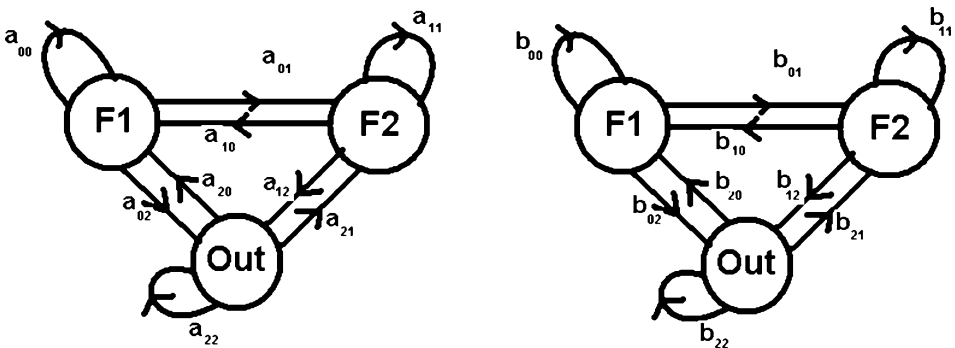


Figure 1. Three-state Markov models for flame (left) and non-flame moving pixels.

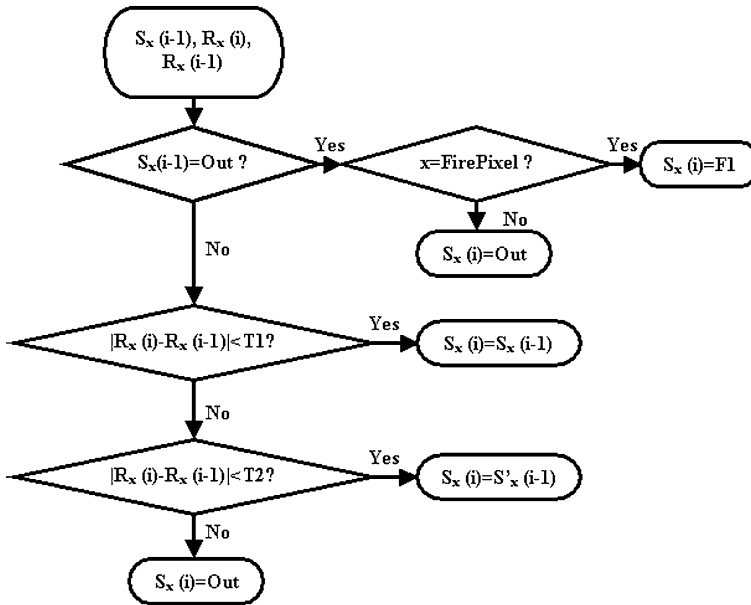


Figure 2. State transition flow-chart of the Markov chain.

Temporal variation in RGB values of each pixel belonging to a moving region is used as a one dimensional (1-D) feature signal, $F = f(n)$, and fed to the Markov models shown in Figure 1.

A moving pixel is classified as a fire pixel when the probability of obtaining the observed feature signal $F = f(n)$ given the probability model λ_1 is greater than the probability of obtaining the observed feature signal $F = f(n)$ given the probability model λ_2 , i.e., when the pixel has fire color characteristics :

$$p_1 = P(F|\lambda_1) > p_2 = P(F|\lambda_2) \quad (5)$$

where F is the observed feature signal, λ_1 and λ_2 represent the Markov models for fire and ordinary moving objects, respectively.

As the probability $p_1(p_2)$ gets a larger value than $p_2(p_1)$, the confidence level of this sub-algorithm increases (decreases). A decision function, $D_1(x, n)$, is defined describing the Markov model based flame colored region detection sub-algorithm. The zero-mean decision function $D_1(x, n)$ is determined by the normalized difference of Markov model probabilities:

$$D_1(x, n) = \begin{cases} \frac{p_1 - p_2}{p_1 + p_2}, & \text{if } x \text{ is a moving pixel} \\ -1, & \text{otherwise} \end{cases} \quad (6)$$

When a moving pixel is classified as a fire colored pixel, i.e., $p_1 \gg p_2$, $D_1(x, n)$ is close to 1. Otherwise, the decision function $D_1(x, n)$ is close to -1.

2.2. Temporal Wavelet Analysis for Flicker Detection

The second sub-algorithm analyzes the frequency history of pixels in flame colored moving regions. In order to detect flicker or oscillations in pixels due to fire in a reliable manner, the video capture rate should be high enough to capture high-frequency flicker in flames. To capture 10 Hz flicker, the video should capture at least 20 frames per second (fps). However, in some surveillance systems, the video capture rate is below 20 Hz. If the video is available at a lower capture rate, aliasing occurs but flicker due to flames can still be observed in the video. For example, 8 Hz sinusoid appears as 2 Hz sinusoid in a 10 fps video.

Each pixel $I(x, n)$ at location x belonging a fire colored moving object in the image frame at time step n is fed to a two stage-filter bank as shown in Figure 3. The signal $\tilde{I}_n(x)$ is a one-dimensional signal representing the temporal variations in color values of the pixel $I(x, n)$ at location x in the n th image frame. Temporal wavelet analysis can be carried out using either the luminance (Y component) in YUV color representation or the red component in RGB color representation. In our implementation the red channel values of the pixels are used. The two-channel subband decomposition filter bank is composed of half-band high-pass and low-pass filters with filter coefficients $\{-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}\}$ and $\{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$, respectively, as shown in Figure 3. The filter bank produces wavelet subsignals $d_n(x)$ and $e_n(x)$. If there is high frequency activity at pixel location x , high-band subsignals d and e get non-zero values. However, in a stationary pixel, the values of these two subsignals should be equal to zero or very close to zero because of high-pass filters used in subband analysis. If the pixel is part of a flame boundary at some time (see Figure 4), then there will be several spikes in one second due to transitions from background colors to flame colors and vice versa. If there is an ordinary fire-colored moving object going through pixel at location x , then there will be a single spike in one of these wavelet subsignals because of the transition from the background pixel to the object pixel as shown in Figure 5. The number of zero crossings of the subband signals d_n and e_n in a few seconds can be used to discriminate between a flame pixel and an ordinary fire colored object pixel. If this number is above some threshold, then an alarm can be issued for this pixel.

The temporal history of the red channel of a pixel at location $x = (111, 34)$ which is part of a flame, and the corresponding wavelet signals are shown in

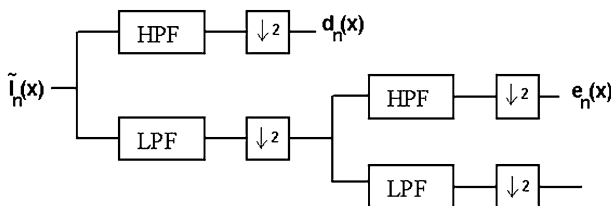


Figure 3. A two-stage filter bank. HPF and LPF represent half-band high-pass and low-pass filters, with filter coefficients $\{-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}\}$ and $\{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$, respectively. This filter bank is used for wavelet analysis.

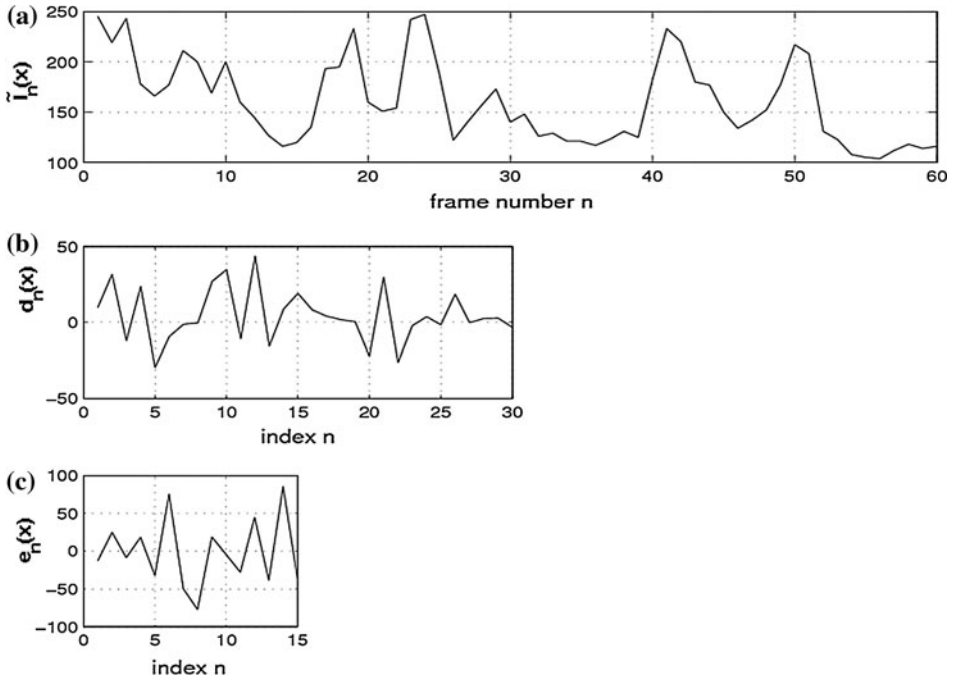


Fig. 4. (a) Temporal variation of image pixel at location $x = (111, 34)$, $\tilde{I}_n(x)$. The pixel at $x = (111, 34)$ is part of a flame for image frames $I(x, n)$, $n = 1, 2, 3, 19, 23, 24, 41$ and 50 . It becomes part of the background for $n = 12, \dots, 17, 20, 21, 26, 27, 31, \dots, 39, 45, 52, \dots$, and 60 . Wavelet domain subsignals (b) d_n and (c) e_n reveal the fluctuations of the pixel at location $x = (111, 34)$.

Figure 4. A flicker in the red channel values of this flame pixel is obvious from the figure. The pixel is part of a flame for image frames with time steps $n = 1, 2, 3, 19, 23, 24, 41$ and 50 . It becomes part of the background for $n = 12, \dots, 17, 20, 21, 26, 27, 31, \dots, 39, 45, 52, \dots$, and 60 . Wavelet domain subsignals d_n and e_n reveal the fluctuations of the pixel at location $x = (111, 34)$ with several zero crossings. Due to a down-sampling operation during wavelet computation, the length of wavelet signals are halved after each stage of subband filtering. As a result, the value of a sample in a subband signal corresponds to several samples in the original signal, e.g., the value of $d_5(111, 34)$ corresponds to the values of $\tilde{I}_{10}(111, 34)$ and $\tilde{I}_{11}(111, 34)$, and the value of $e_4(111, 34)$ corresponds to the values of $\tilde{I}_{12}(111, 34)$, $\tilde{I}_{13}(111, 34)$, $\tilde{I}_{14}(111, 34)$ and $\tilde{I}_{15}(111, 34)$, in the original signal.

The temporal history of the red channel of a pixel at location $x = (18, 34)$, which is part of a fire colored object, and the corresponding wavelet signals are shown in Figure 5. As shown in this figure, neither the original nor the wavelet signals exhibit oscillatory behavior. The pixel is part of a white-colored background for $n = 1, 2$, and 3 , becomes part of a fire colored object for $n = 4, 5, 6$,

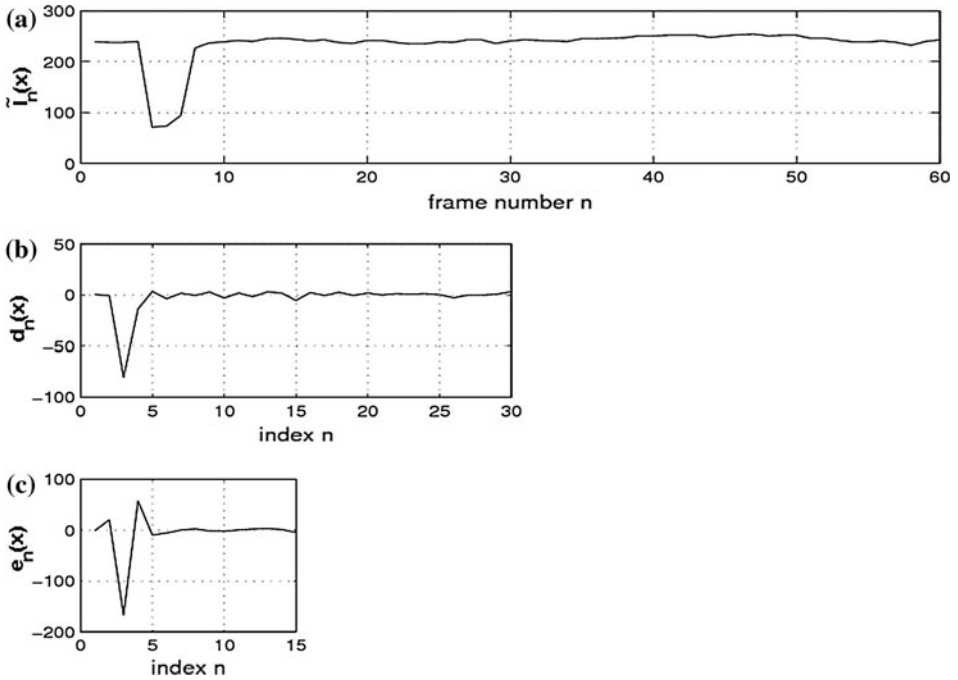


Figure 5. (a) Temporal history of the pixel at location $x = (18, 34)$. It is part of a fire-colored object for $n = 4, 5, 6, 7,$ and 8 , and it becomes part of the background afterwards. Corresponding subsignals (b) d_n and (c) e_n exhibit stationary behavior for $n > 8$.

7, and 8, then it becomes part of the background again for $n > 8$. Corresponding wavelet signals d_n and e_n do not exhibit oscillatory behavior as shown in Figure 5. Small variations due to noise around zero after the 10-th frame are ignored by setting up a threshold.

The number of wavelet stages needed for used in flame flicker analysis is determined by the video capture rate. In the first stage of dyadic wavelet decomposition, the low-band subsignal and the high-band wavelet subsignal $d_n(x)$ of the signal $\tilde{I}_n(x)$ are obtained. The subsignal $d_n(x)$ contains [2.5 Hz, 5 Hz] frequency band information of the original signal $\tilde{I}_n(x)$ in 10 Hz video frame rate. In the second stage the low-band subsignal is processed once again using a dyadic filter bank, and the wavelet subsignal $e_n(x)$ covering the frequency band [1.25 Hz, 2.5 Hz] is obtained. Thus by monitoring wavelet subsignals $e_n(x)$ and $d_n(x)$, one can detect fluctuations between 1.25 to 5 Hz in the pixel at location x .

Flame flicker can be detected in low-rate image sequences obtained with a rate of less than 20 Hz as well in spite of the aliasing phenomenon. To capture 10 Hz flicker, the video should capture at least 20 frames per second (fps). However, in some surveillance systems, the video capture rate is below 20 Hz. If the video is available at a lower capture rate, aliasing occurs but flicker due to flames can still

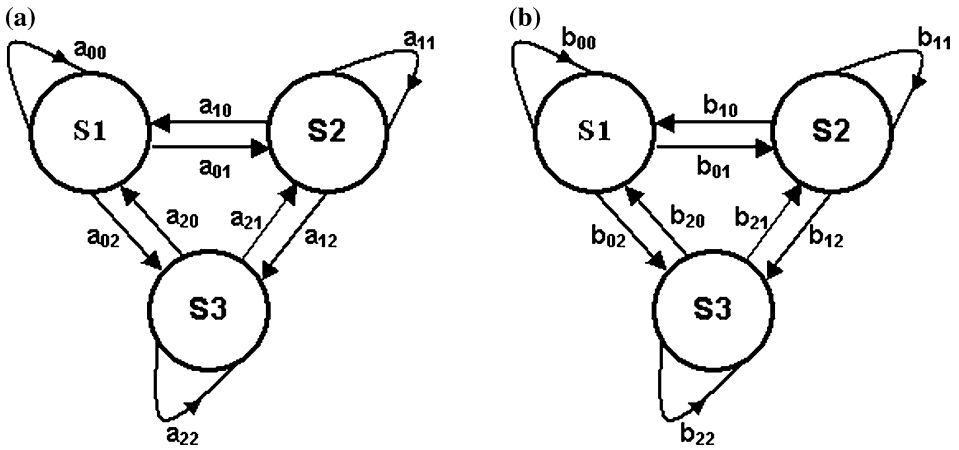


Figure 6. Three-state Markov models for (a) flame and (b) non-flame moving flame-colored pixels.

be observed in the video. For example, 8 Hz sinusoid appears as 2 Hz sinusoid in a 10 fps video [9]. Aliased version of flame flicker signal is also a wide-band signal in discrete-time Fourier Transform domain. This characteristic flicker behavior is very well suited to be modeled as a random Markov model which is extensively used in speech recognition systems and recently these models have been used in computer vision applications [17].

Three-state Markov models are trained off-line for both flame and non-flame pixels to represent the temporal behavior (Figure 6). These models are trained by using first-level wavelet coefficients $d_n(x)$ corresponding to the intensity values $\tilde{I}_n(x)$ of the flame-colored moving pixel at location x as the feature signal. A single-level decomposition of the intensity variation signal $\tilde{I}_n(x)$ is sufficient to characterize the turbulent nature of flame flicker. One may use higher-order wavelet coefficients such as, $e_n(x)$, for flicker characterization, as well. However, this may incur additional delays in detection.

Wavelet signals can easily reveal the random characteristic of a given signal which is an intrinsic nature of flame pixels. That is why the use of wavelets instead of actual pixel values lead to more robust detection of flames in video. Since, wavelet signals are high-pass filtered signals, slow variations in the original signal lead to zero-valued wavelet coefficients. Hence it is easier to set thresholds in the wavelet domain to distinguish slow varying signals from rapidly changing signals. Non-negative thresholds $T_1 < T_2$ are introduced in wavelet domain to define the three states of the Hidden Markov Models for flame and non-flame moving objects. For the pixels of regular flame colored objects no rapid changes take place in the pixel values. The lower threshold T_1 basically determines a given wavelet coefficient being close to zero. The second threshold T_2 indicates that the wavelet coefficient is significantly higher than zero. When the wavelet coefficients fluctuate between values above the higher threshold T_2 and below the lower threshold T_1 in a frequent manner this indicates the existence of flames in the viewing range of the camera.

The states of HMMs are defined as follows: at time n , if $|w(n)|_1$, the state is in $S1$; if $T_1 < |w(n)|_2$, the state is $S2$; else if $|w(n)|_2$, the state $S3$ is attained. Here $|w(n)|$ denotes the absolute value of the wavelet coefficient corresponding to the currently analyzed pixel. For the pixels of regular flame-colored moving objects, like walking people in red shirts, no rapid changes take place in the pixel values. Therefore, the temporal wavelet coefficients ideally should be zero but due to thermal noise of the camera the wavelet coefficients wiggle around zero. The lower threshold T_1 basically determines a given wavelet coefficient being close to zero. The state defined for the wavelet coefficients below T_1 is $S1$. The second threshold T_2 indicates that the wavelet coefficient is significantly higher than zero. The state defined for the wavelet coefficients above this second threshold T_2 is $S3$. The values between T_1 and T_2 define $S2$. The state $S2$ provides hysteresis and it prevents sudden transitions from $S1$ to $S3$ or vice versa. When the wavelet coefficients fluctuate between values above the higher threshold T_2 and below the lower threshold T_1 in a frequent manner this indicates the existence of flames in the viewing range of the camera.

In flame pixels, the transition probabilities a 's should be high and close to each other due to random nature of uncontrolled fire. On the other hand, transition probabilities should be small in constant temperature moving bodies because there is no change or little change in pixel values. Hence we expect a higher probability for b_{00} than any other b value in the non-flame moving pixels model (cf. Figure 6), which corresponds to higher probability of being in $S1$. The state $S2$ provides hysteresis and it prevents sudden transitions from $S1$ to $S3$ or vice versa.

The transition probabilities between states for a pixel are estimated during a pre-determined period of time around flame boundaries. In this way, the model not only learns the way flame boundaries flicker during a period of time, but also it tailors its parameters to mimic the spatial characteristics of flame regions. The way the model is trained as such, drastically reduces the false alarm rates.

During the recognition phase, the HMM based analysis is carried out in pixels near the contour boundaries of flame-colored moving regions. The state sequence of length 20 image frames is determined for these candidate pixels and fed to the flame and non-flame pixel models.

Let $p1$ and $p2$ denote the probabilities obtained from the models for flame and non-flame pixels respectively. As the probability $p_1(p_2)$ gets a larger value than $p_2(p_1)$, the confidence level of this sub-algorithm increases (decreases). Therefore, the zero-mean decision function $D_2(x, n)$ is determined by the normalized difference of these probabilities:

$$D_2(x, n) = \frac{p_1 - p_2}{p_1 + p_2} \quad (7)$$

When a fire colored moving region is classified as a fire pixels according to frequency history, i.e., $p_1 \gg p_2$, $D_2(x, n)$ is close to 1. Otherwise, the decision function $D_2(x, n)$ is close to -1 .

The probability of a Markov model producing a given sequence of wavelet coefficients is determined by the sequence of state transition probabilities. Therefore,

the flame decision process is insensitive to the choice of thresholds T_1 and T_2 , which basically determine if a given wavelet coefficient is close to zero or not. Still, thresholds can be determined using a k-means type algorithm, as well.

2.3. Spatial Wavelet Analysis

The third sub-algorithm is the spatial wavelet analysis of moving regions containing fire colored pixels to capture color variations in pixel values. In an ordinary fire-colored object there will be little spatial variations in the moving region as shown in Figure 7a. On the other hand, there will be significant spatial variations in a fire region as shown in Figure 8a. The spatial wavelet analysis of a rectangular frame containing the pixels of fire-colored moving regions is performed. The images in Figures 7b and 8b are obtained after a single stage two-dimensional wavelet transform that is implemented in a separable manner using the same filters explained in Subsect. 2. Absolute values of low-high, high-low and high-high wavelet subimages are added to obtain these images. A decision parameter describing spatial variance is defined for this step, according to the energy of the wavelet subimages:

$$\xi = \frac{1}{M \times N} \sum_{k,l} |I_{lh}(k, l)| + |I_{hl}(k, l)| + |I_{hh}(k, l)|, \quad (8)$$

where $I_{lh}(k, l)$ is the low-high subimage, $I_{hl}(k, l)$ is the high-low subimage, and $I_{hh}(k, l)$ is the high-high subimage of the wavelet transform, respectively, and $M \times N$ is the number of pixels in the fire-colored moving region. If the decision parameter of the fourth step of the algorithm, ξ , exceeds a threshold, then it is likely that this moving and fire-colored region under investigation is a fire region. The decision function for this sub-algorithm is determined as follows:

$$D_3(x, n) = \begin{cases} 2 \frac{\xi}{\xi_{\max}} - 1, & \text{if } \xi \geq \xi_T \\ -1, & \text{otherwise} \end{cases} \quad (9)$$

where ξ_{\max} and ξ_T are experimentally determined parameters from videos containing flames. ξ_{\max} is the largest value that ξ can take and ξ_T is a predefined threshold. The threshold determines the definite non-fire cases. The decision function is not sensitive to this threshold. One can also use $D_3(x, n) = 2 \frac{\xi}{\xi_{\max}} - 1$ as the decision without the dependence on the threshold.

2.4. Wavelet Domain Analysis of Object Contours

The fourth sub-algorithm of the proposed method analyzes the contours of flame colored objects. A one-dimensional (1-D) signal $x(\theta)$ is obtained by computing the distance from the center of mass of the object to the object boundary for $0 \leq \theta < 2\pi$. In Figure 9, two image frames are shown. Example feature functions for fire colored shirt and the fire region in Figure 9 are shown in Figure 10 for 64 equally spaced angles $x[l] = x(l\theta_s)$, $\theta_s = \frac{2\pi}{64}$. To determine the high-frequency content of a curve, we use a single scale wavelet transform shown in Figure 11. The feature signal $x[l]$ is fed to a filterbank shown in Figure 11 and the low-band signal

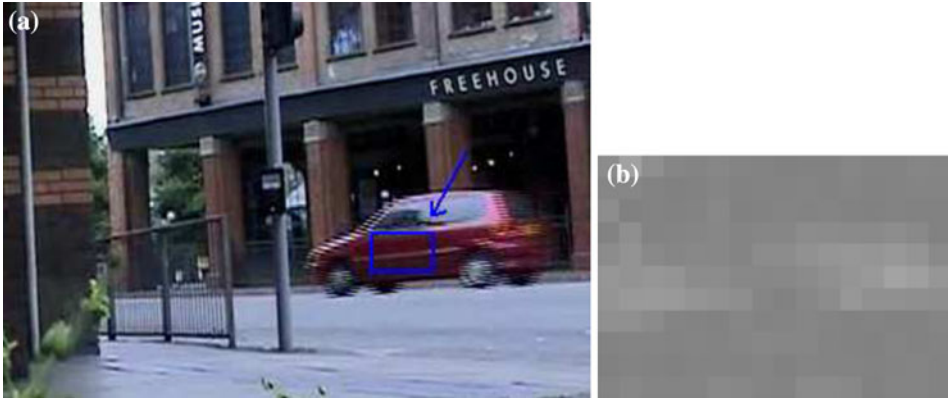


Fig. 7. (a) A fire colored moving car, and (b) the absolute sum of spatial wavelet transform coefficients, $|I_{lh}(k,l)| + |I_{hl}(k,-l)| + |I_{hh}(k,l)|$, of the region bounded by the indicated rectangle. (Color figure online).



Fig. 8. (a) Fire, and (b) the absolute sum of spatial wavelet transform coefficients, $|I_{lh}(k,l)| + |I_{hl}(k,l)| + |I_{hh}(k,l)|$, of the region bounded by the indicated rectangle.

$$c[l] = \sum_m h[2l - m]x[m] \tag{10}$$

and the high-band subsignal

$$w[l] = \sum_m g[2l - m]x[m] \tag{11}$$

are obtained. Coefficients of the lowpass and the highpass filters are $h[l] = \{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$ and $g[l] = \{-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}\}$, respectively [3, 18].

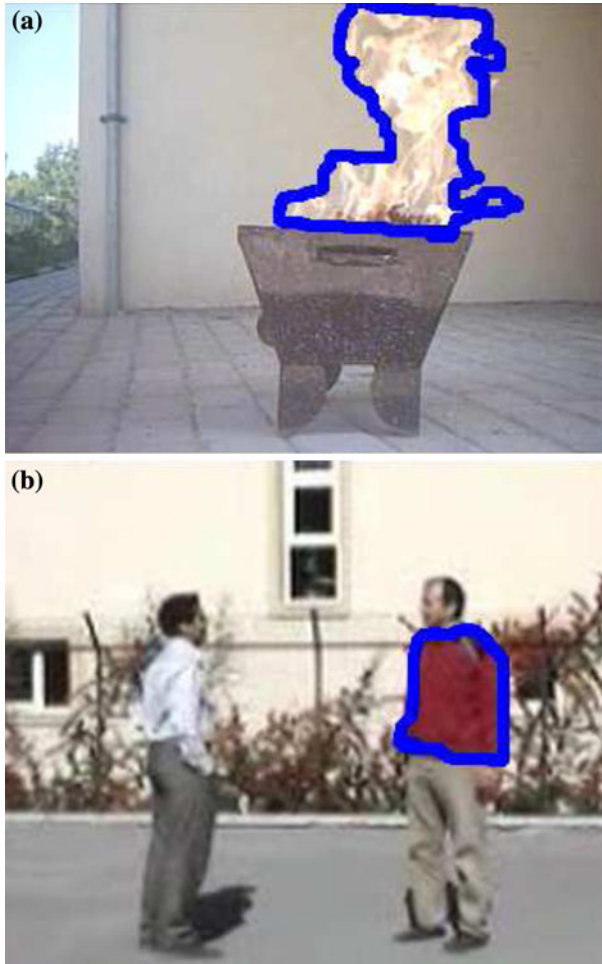


Figure 9. Two fire colored objects in video: (a) fire image, and (b) fire colored shirt. (Color figure online).

The absolute values of wavelet, $w[l]$ and low-band $c[l]$ coefficients of the fire region and the fire colored shirt are shown in Figures 12 and 13, respectively. The high-frequency variations of the feature signal of the fire region is clearly distinct from that of the shirt. Since regular objects have relatively smooth boundaries compared to flames, the high-frequency wavelet coefficients of flame boundary feature signals have more energy than regular objects. Therefore, the ratio of the wavelet domain energy to the energy of the low-band signal is a good indicator of a fire region. This ratio is defined as

$$\rho = \frac{\sum_l |w[l]|}{\sum_l |c[l]|} \quad (12)$$

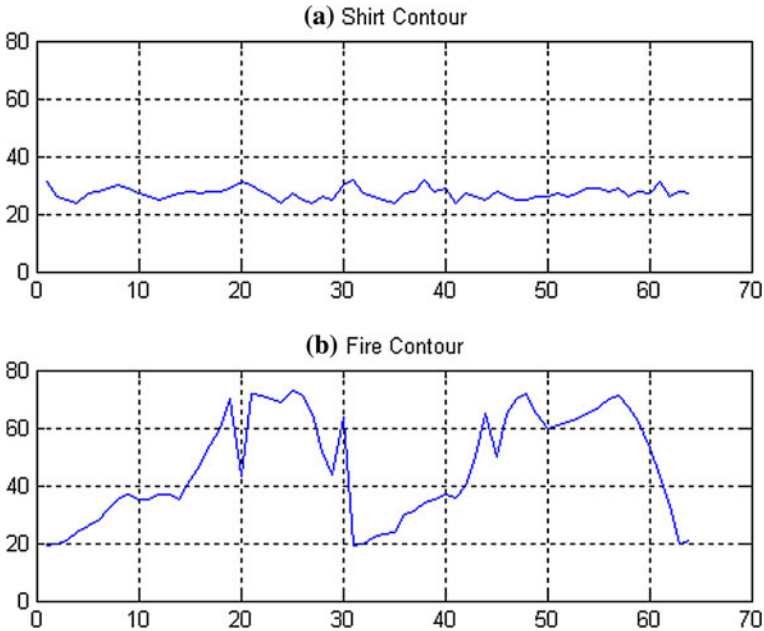


Figure 10. Equally spaced 64 contour points of the (a) fire colored shirt, and (b) the fire region.

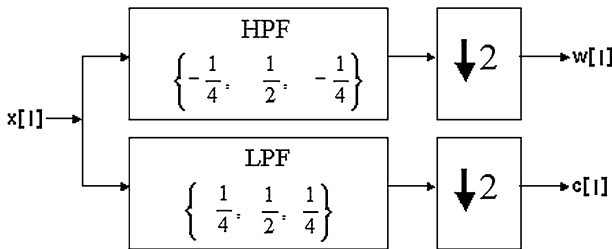


Figure 11. Single-stage wavelet filter bank. The high-pass and the low-pass filter coefficients are $\{-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}\}$ and $\{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$, respectively.

The likelihood of the moving region to be a fire region is highly correlated with the parameter ρ . Higher the value of ρ , higher the probability of the region belonging to flame regions. The decision function for this sub-algorithm is defined as follows:

$$D_4(x, n) = \begin{cases} 2\frac{\rho}{\rho_{max}} - 1, & \text{if } \rho \geq \rho_T \\ -1, & \text{otherwise} \end{cases} \quad (13)$$

where ρ_{max} is the maximum value of ρ and ρ_T is an experimentally determined threshold. The threshold determines the definite non-fire cases. The decision function is not sensitive to this threshold. One can also use $D_4(x, n) = 2\frac{\rho}{\rho_{max}} - 1$ as the decision without the dependence on the threshold.

3. Sub-Algorithm Weight Update

As described in Sect. 2, the main fire detection algorithm is composed of four sub-algorithms. Each algorithm has its own decision function. Weights of sub-algorithms are adaptively updated by linearly combining decision values from sub-algorithms. The adaptive update of sub-algorithm weights is performed according to the Least-Mean-Square (LMS) algorithm which is the most widely used adaptive filtering method [19, 20].

Most adaptive decision methods use binary values 1(correct) or -1 (false) as the outputs of decision methods, in this paper we implement a more versatile approach by designing decision functions in such a way that they produce a zero-mean real numbers in the range $[-1, 1]$ as decision values. If the number is positive (negative), then the individual algorithm decides that there is (not) fire in the viewing range of the camera. Higher the absolute value, the more confident the sub-algorithm.

When the composite algorithm is composed of M -many detection algorithms: D_1, \dots, D_M each algorithm yields a zero-mean decision value $D_i(x) \in R$, upon receiving a sample input x . The type of the sample input x may vary depending on the algorithm. It may be an individual pixel, or an image region, or the entire image depending on the sub-algorithm of the computer vision problem. In the fire detection problem the number of sub-algorithms, $M = 4$ and each pixel at the location x of incoming image frame is considered as a sample input for every detection algorithm.

Let $\mathbf{D}(x, n) = [D_1(x, n) \dots D_M(x, n)]^T$, be the vector of confidence values of the sub-algorithms for the pixel at location x of input image frame at time step n , and $\mathbf{w}(n) = [w_1(n) \dots w_M(n)]^T$ be the current weight vector.

We define

$$\hat{y}(x, n) = \mathbf{D}^T(x, n)\mathbf{w}(n) = \sum_i w_i(n)D_i(x, n) \quad (14)$$

as an estimate of the correct classification result $y(x, n)$ of the oracle for the pixel at location x of input image frame at time step n , and the error $e(x, n)$ as $e(x, n) = y(x, n) - \hat{y}(x, n)$. Weights are updated by minimizing the mean-square-error (MSE):

$$\min_{w_i} E[(y(x, n) - \hat{y}(x, n))^2], \quad i = 1, \dots, M \quad (15)$$

where E represents the expectation operator. Taking the derivative with respect to weights:

$$\frac{\partial E}{\partial w_i} = -2E[(y(x, n) - \hat{y}(x, n))D_i(x, n)] = -2E[e(x, n)D_i(x, n)], \quad i = 1, \dots, M \quad (16)$$

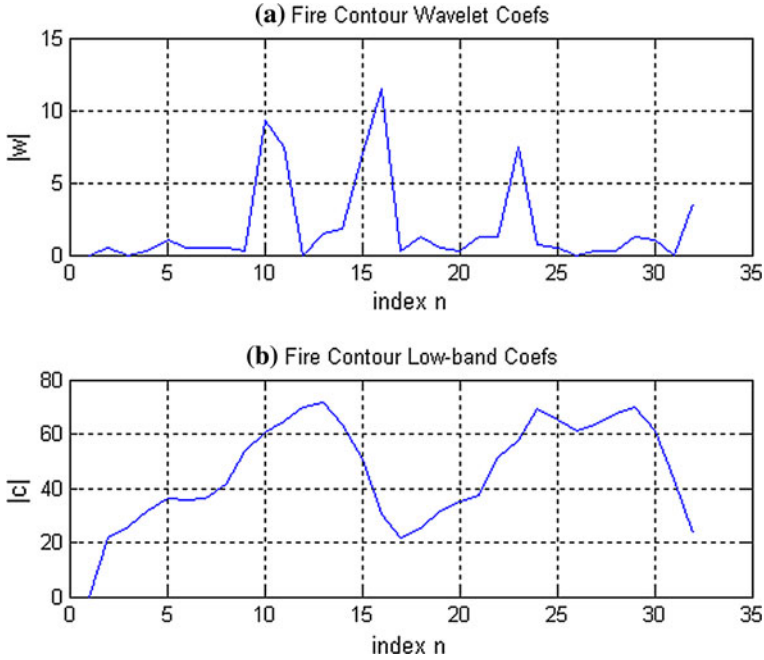


Figure 12. The absolute values of (a) wavelet and (b) low-band coefficients for the fire region.

and setting the result to zero:

$$-2E[e(x, n)D_i(x, n)] = 0, \quad i = 1, \dots, M \tag{17}$$

a set of M equations is obtained. The solution of this set of equations is called the Wiener solution [19, 20]. Unfortunately, the solution requires the computation of cross-correlation terms in Equation 17. The gradient in Equation 16 can be used in a steepest descent algorithm to obtain an iterative solution to the minimization problem in Equation 15 as follows:

$$\mathbf{w}(n + 1) = \mathbf{w}(n) + \lambda E[e(x, n)\mathbf{D}(x, n)] \tag{18}$$

where λ is a step size. In the well-known LMS algorithm, the ensemble average $E[e(x, n)\mathbf{D}(x, n)]$ is estimated using the instantaneous value $e(x, n)\mathbf{D}(x, n)$ or it can be estimated from previously processed pixels as follows:

$$\hat{e}(x, n)\hat{\mathbf{D}}(x, n) = \frac{1}{L} \sum_{x, n} e(x, n)\mathbf{D}(x, n) \tag{19}$$

where L is the number of previously processed pixels. The LMS algorithm is derived by noting that the quantity in Equation 18 is not available but its

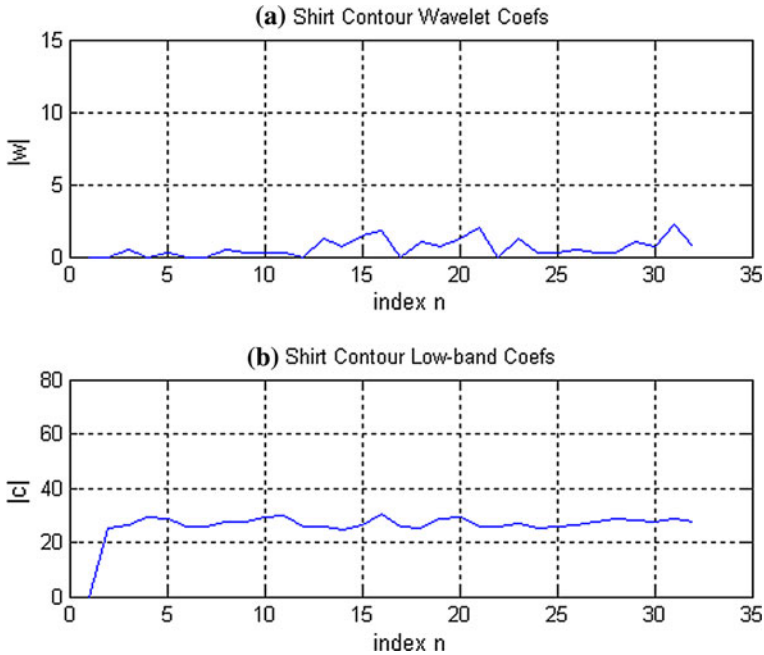


Figure 13. The absolute (a) wavelet and (b) low-band coefficients for the shirt.

instantaneous value is easily computable, and hence the expectation is simply replaced by its instantaneous value [21]:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \lambda e(x,n)\mathbf{D}(x,n) \quad (20)$$

Equation 20 is a computable weight-update equation. Whenever the oracle provides a decision, the error $e(x,n)$ is computed and the weights are updated according to Equation 20. Note that, oracle does not assign her/his decision to each and every pixel one by one. She/he actually selects a window on the image frame and assigns a “1” or “-1” to the selected window.

Convergence of the LMS algorithm can be analyzed based on the MSE surface:

$$E[e^2(x,n)] = P_y(x,n) - 2\mathbf{w}^T \mathbf{p} - \mathbf{w}^T \mathbf{R} \mathbf{w} \quad (21)$$

where $P_y = E[y^2(x,n)]$, $\mathbf{p} = E[y(x,n)\mathbf{D}(x,n)]$, and $\mathbf{R} = E[\mathbf{D}(x,n)\mathbf{D}^T(x,n)]$, with the assumption that $y(x,n)$ and $\mathbf{D}(x,n)$ are wide-sense-stationary random processes. The MSE surface is a function of the weight vector \mathbf{w} . Since $E[e^2(x,n)]$ is a quadratic function of \mathbf{w} , it has a single global minimum and no local minima. Therefore, the steepest descent algorithm of Equations 18 and 20 is guaranteed to converge to the Wiener solution, \mathbf{w}^* [21] with the following condition on the step size λ [20]:

$$0 < \lambda < \frac{1}{\alpha_{\max}} \quad (22)$$

where α_{\max} is the largest eigenvalue of \mathbf{R} .

In Equation 20, the step size λ can be replaced by

$$\frac{\mu}{\|\mathbf{D}(x, n)\|^2} \quad (23)$$

as in the normalized LMS algorithm, which leads to:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \frac{e(x, n)}{\|\mathbf{D}(x, n)\|^2} \mathbf{D}(x, n) \quad (24)$$

where the μ is an update parameter and the normalized LMS algorithm converges for $0 < \mu < 2$ to the Wiener solution, w^* with the wide-sense-stationarity assumption. Initially the weights can be selected as $\frac{1}{M}$. The adaptive algorithm converges, if $y(x, n)$ and $D_i(x, n)$ are wide-sense stationary random processes and when the update parameter μ lies between 0 and 2 [22]. Unfortunately, the wide-sense-stationarity assumption is not a valid assumption in natural images, as in many signal processing applications. Nevertheless, the LMS algorithm is successfully used in many telecommunication and signal processing problems. Wide-sense-stationarity assumption may be valid in some parts of a sequence in which there are no spatial edges and temporal changes.

In all the tests that we carried out, sub-algorithms described in the previous section yield non-negative decision values, D_i 's, for pixels inside fire regions. The final decision which is nothing but the weighted sum of individual decisions must also take a non-negative value when the decision functions yield non-negative values. This implies that, in the weight update step of the active decision fusion method, weights should also be non-negative: $w(n) \geq 0$. In the proposed method, the weights are updated according to Equation 24 and negative weights are reset to zero complying with the non-negative weight constraint.

The main advantage of the LMS algorithm compared to other related methods, such as the weighted majority algorithm [23], is the controlled feedback mechanism based on the error term. Weights of the algorithms producing incorrect (correct) decision is reduced (increased) according to Equation 24 in a controlled and fast manner. In weighted majority algorithm, conflicting weights with the oracle are simply reduced by a factor of two [23, 24]. Another advantage of the LMS algorithm is that it does not assume any specific probability distribution about the data.

4. Experimental Results

Three approaches are compared with each other in the experiments: (a) LMS based method, (b) Weighted majority algorithm (WMA) based method, and (c) non-adaptive method. The method with no adaptive learning simply issues an

```

Weighted Majority(x,n)
for  $i = 1$  to  $M$  do
     $w_i(0) = \frac{1}{M}$ , Initialization
end for
if  $\sum_{i:d_i(x,n)=1} w_i(n) \geq \sum_{i:d_i(x,n)=-1} w_i(n)$  then
    return 1
else
    return -1
end if
for  $i = 1$  to  $M$  do
    if  $d_i(x,n) \neq y$  then
         $w_i(n+1) \leftarrow \frac{w_i(n)}{2}$ 
    end if
end for

```

Figure 14. The pseudo-code for the weighted majority algorithm.

alarm if all of the decision functions are 1 for the case of binary decision functions producing outputs 1 and -1 for fire and non-fire regions. Comparative tests are carried out with recordings containing actual fire and test sequences with no fires. Fire alarms are issued by all three methods at about the same time after fire becomes visible. However, there are some performance differences among the schemes in terms of false alarm rates.

The WMA is summarized in Figure 14 [24]. In WMA, as opposed to our method, individual decision values from sub-algorithms are binary, i.e., $d_i(x, n) \in \{-1, 1\}$, which are simply the quantized version of real valued $D_i(x, n)$ defined in Sect. 2. In the WMA, the weights of sub-algorithms yielding contradictory decisions with that of the oracle are reduced by a factor of two in an uncontrolled manner, unlike the proposed LMS based algorithm [23]. Initial weights for WMA are taken as $\frac{1}{M}$, as in the proposed LMS based scheme.

The LMS based scheme, the WMA based scheme and the non-adaptive approach are compared with each other in the following experiments. In Tables 1 and 3, video recordings containing actual fires and video sequences with no fires are used. The durations of the fire videos, the burning material in the video, and the explanation of the fire source for the fire videos in Table 1 are given in Table 2. The snapshots of the fire videos are also given in Figures 15 and 16. The fire videos are mostly outdoor recordings or forest fires, burning paper, oil, gasoline, and petroleum. The snapshots from the videos that do not contain fire are shown in Figure 17. The videos V18 and V19 which are used for false alarm tests contain car headlights that show oscillatory behavior, V20 has a fire colored moving car, V21 has a man with fire colored shirt, V22 has yellow reeds moving in the wind, V23 has moving tree leaves against a fire-colored background which gives the effect of moving flames, V24 has red flowers moving in the wind.

LMS and WMA based decision fusion methods detect fires within 12 seconds but the method with no learning capability failed to produce alarms for 3 of the video sequences, as shown in Table 1. The LMS based method issues a correct

Table 1
Three Different Methods (Non-Adaptive, LMS Based, WMA Based) are Compared in Terms of Frame Numbers at Which an Alarm is Issued for Fire Captured at Various Ranges and fps

Video seq.	Range (m)	Frame rate (fps)	Frame number/time (sec) of first alarm					
			LMS based		WMA based		Non-adaptive	
V1	2	10	44	4.4 s	39	3.9 s	51	5.1 s
V2	50	30	48	1.6 s	43	1.4 s	62	2.0 s
V3	50	30	49	1.6 s	35	1.2 s	37	1.2 s
V4	30	30	106	3.5 s	44	1.5 s	No alarm	
V5	1	30	64	2.1 s	42	1.4 s	18	0.6 s
V6	5	10	43	4.3 s	140	14 s	38	3.8 s
V7	60	30	334	11.1 s	320	10.7 s	349	11.6 s
V8	80	30	73	2.4 s	78	2.6 s	No alarm	
V9	100	30	41	1.4 s	36	1.2 s	12	0.4 s
V10	10	15	48	3.2 s	44	2.9 s	56	3.7 s
V11	20	15	49	3.3 s	33	2.2 s	48	3.2 s
V12	50	15	46	3.0 s	41	2.7 s	80	5.3 s
V13	40	15	47	3.1 s	43	2.9 s	18	1.2 s
V14	30	15	72	4.8 s	68	4.5 s	20	1.3 s
V15	70	30	212	7.0 s	216	7.2 s	156	5.2 s
V16	15	30	259	8.6 s	249	8.3 s	163	5.4 s
V17	20	30	68	2.3 s	47	1.6 s	No alarm	
Average			94.3	4.0 s	89.3	4.1 s	79.1	3.6 s

It is assumed that the fire starts at frame 0

Table 2
The Durations, the Burning Material and Explanations for the Fire Videos in Table 1

Video seq.	Duration (frames)	Material	Explanation
V1	430	Pine cones	Burning pine cones
V2	330	Wood	Forest fire
V3	210	Wood	Forest fire
V4	1200	Wood + diesel oil	A burning boat
V5	900	Gasoline	A burning car
V6	1200	Cardboard + paper	Burning paper and cardboard
V7	1200	Wood	Forest fire
V8	210	Wood	Forest fire
V9	240	Wood	Forest fire
V10	750	Dry grass	Dry grass burning in a bin
V11	225	Wood	Forest fire
V12	225	Wood	Forest fire
V13	225	Wood	Forest fire
V14	220	Wood	Forest fire
V15	210	Petroleum	Fire at a refinery
V16	210	Wood + fuel	Fire at a ranch
V17	270	Wood + gasoline	A burning RV



Figure 15. Frames from the fire videos in Table 1.

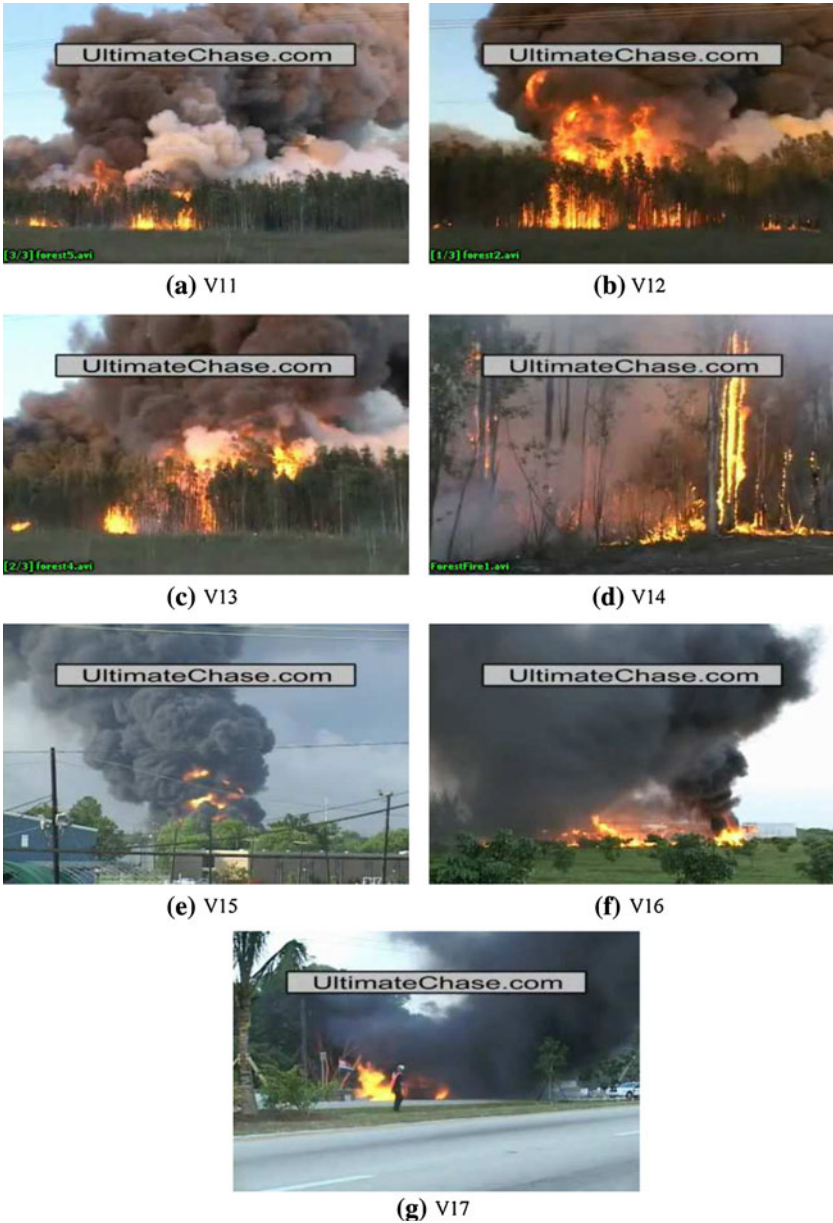


Figure 16. Frames from the fire videos in Table 1.

alarm within 4 seconds on the average. The detection rates of the methods are comparable to each other. On the other hand, the proposed adaptive fusion strategy reduces the false alarm rate of the system by integrating the feedback from the guard (oracle) into the decision mechanism within the active learning



(a) V18



(b) V19



(c) V20



(d) V21



(e) V22



(f) V23



(g) V24

Figure 17. Frames from the nuisance sources in Table 3.

Table 3
Three Different Methods (Non-Adaptive, LMS Based, WMA Based) are Compared in Terms of the Number of False Alarms Issued for Fire Video Sequences that Do Not Contain Fire

Video seq.	Range (m)	Frame rate (fps)	Duration (frames)	Number of false alarms		
				LMS based	WMA based	Non-adaptive
V18	20	25	1500	0	0	1
V19	25	25	2000	2	1	6
V20	50	25	2000	0	0	2
V21	2	25	150	0	2	1
V22	1	25	500	3	5	7
V23	5	25	1000	0	5	2
V24	1	25	150	0	0	2

framework described in Sect. 3. A set of video clips that do not contain fire is used to generate Table 3. These video clips are especially selected from recordings that contain fire colored moving objects. Number of false alarms issued by different methods are presented. The adaptive algorithms produce lower number of false alarms and LMS based scheme is better than WMA except for one video sequence. Total number of false alarms for the clips in Table 3 issued by the methods (a) the LMS based scheme, (b) the WMA based scheme, (c) the non-adaptive approach are 5, 13 and 21, respectively.

5. Conclusion

A video based fire detection algorithm with LMS based active learning capability is developed. The main algorithm comprises four sub-algorithms which produce their individual decision values for fires. Each algorithm is designed to characterize an aspect of fire. The decision functions of sub-algorithms yield their own decisions as confidence values in the range $[-1, 1] \in R$. Computationally efficient sub-algorithms are selected in order to realize a real-time fire detection system working on a standard PC. The LMS based adaptive decision fusion strategy takes into account the feedback from the user of the application. Experimental results show that the learning duration is decreased with the proposed active learning scheme. It is also observed that false alarm rate of the proposed LMS based algorithm is the lowest in our data set, compared to non-adaptive and WMA based methods.

Acknowledgment

This work was supported in part by the Scientific and Technical Research Council of Turkey, TUBITAK, with grant no. 106G126 and 105E191, and in part by European Commission 6th Framework Program with grant number FP6-507752 (MUSCLE Network of Excellence Project).

References

1. Phillips W, Shah M, Lobo NV (2002) Flame recognition in video. *Pattern Recognit Lett* 23:319–327
2. Mallat S, Zhong S (1992) Characterization of signals from multiscale edges. *IEEE Trans Pattern Anal Mach Intell* 14(7):710–732
3. Cetin AE, Ansari R (1994) Signal recovery from wavelet transform maxima. *IEEE Trans Sig Process* 42:194–196
4. Quatieri TF (2001) *Discrete-time speech signal processing: principles and practice*. Prentice-Hall, Indiana
5. Cetin AE, Jabloun F, Erzin E (1999) Teager energy based feature parameters for speech recognition in car noise. *IEEE Sig Pcess Lett* 6(10):259–261
6. Healey G, Slater D, Lin T, Drda B, and Goedeke AD (1993) A system for real-time fire detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp 15–17
7. Celik T, Demirel H, Ozkaramanli H, Uyguroglu M (2007) Fire detection using statistical color model in video sequences. *J Vis Commun Image Represent* 18(2):176–185
8. Töreyn BU, Çetin AE (2007) Online detection of fire in video. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp 1–5
9. Töreyn BU, Dedeoglu Y, Gudukbay U, and Cetin AE (2006) Computer vision based system for real-time fire and flame detection. *Pattern Recognit Lett* 27:49–58
10. Dedeoglu Y, Töreyn BU, Gudukbay U, Cetin AE (2005) Real-time fire and flame detection in video. In: *Proceedings of the IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp 669–672
11. Töreyn BU, Dedeoglu Y, Cetin AE (2005) Flame detection in video using hidden markov models. In: *Proceedings of the IEEE international conference on image processing (ICIP)*, pp 1230–1233
12. Widrow B, Hoff ME (1960) Adaptive switching circuits. In: *Proceedings of the IRE WESCON (New York Convention Record)*, vol 4, pp 96–104
13. Collins RT, Lipton AJ, Kanade T (1999) A system for video surveillance and monitoring. In: *Proceedings of the 8-th international topical meeting on robotics and remote systems*. April 1999, American Nuclear Society
14. Bagci M, Yardimci Y, Cetin AE (2002) Moving object detection using adaptive subband decomposition and fractional lower order statistics in video sequences. *Signal processing*, pp 1941–1947
15. Stauffer C, Grimson WEL (1999) Adaptive background mixture models for real-time tracking. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, vol 2
16. Heijden F (1996) *Image based measurement systems: object recognition and parameter estimation*. Wiley, New York
17. Bunke H, Caelli T (Eds) (2001) *HMMs applications in computer vision*. World Scientific, Singapore
18. Gerek ÖN, Cetin AE (2000) Adaptive polyphase subband decomposition structures for image compression. *IEEE Trans Image Process* 9:1649–1659
19. Haykin S (2002) *Adaptive filter theory*. Prentice Hall, London
20. Widrow B, Stearns SD (1985) *Adaptive signal processing*. Prentice Hall, NJ
21. Schnaufer BA, Jenkins WK (1993) New data-reusing LMS algorithms for improved convergence. In: *Proceedings of the Asilomar conference*, Pacific Groves, CA pp 1584–1588
22. Widrow B, McCool JM, Larimore MG, Johnson CR (1976) Stationary and nonstationary learning characteristics of the LMS adaptive filter. *Proc IEEE* 64(8):1151–1162

23. Littlestone N, Warmuth MK (1994) The weighted majority algorithm. *Inf Comput* 108:212261
24. Oza NC (2001) Online ensemble learning. Ph.D. thesis, Electrical Engineering and Computer Sciences, University of California September