



Fire flame detection in video sequences using multi-stage pattern recognition techniques

Tung Xuan Truong, Jong-Myon Kim*

School of Computer Engineering & Information Technology, University of Ulsan, 680-749, South Korea

ARTICLE INFO

Available online 26 May 2012

Keywords:

Fire detection
Mixture of Gaussian model
Fuzzy c-means
Support vector machines

ABSTRACT

In this paper, we propose an effective technique that is used to automatically detect fire in video images. The proposed algorithm is composed of four stages: (1) an adaptive Gaussian mixture model to detect moving regions, (2) a fuzzy c-means (FCM) algorithm to segment the candidate fire regions from these moving regions based on the color of fire, (3) special parameters extracted based on the temporal-spatial characteristics of fire regions, and (4) a support vector machine (SVM) algorithm using these special parameters to distinguish between fire and non-fire. Experimental results indicate that the proposed method outperforms other fire detection algorithms, providing high reliability and a low false alarm rate.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Fire detection has become very appealing for both personal security and commercial applications, and several conventional methods have been proposed to detect fire. However, most of these methods require a close proximity to the source of the fire and are based on particle sensors (Jones, 2004). Therefore, they cannot detect fire in open or large spaces and cannot provide additional information regarding the burning process. To overcome these weaknesses, video fire detection is a suitable candidate.

Several video-based fire detection algorithms have been proposed (Chen et al., 2004; Toreyin and Centin, 2007; Celik and Demirel, 2009; Ko et al., 2009; Toreyin et al., 2006; Borges and Izquierdo, 2010; Han and Lee, 2006; Celik et al., 2007). Most of these algorithms are based on color pixel recognition, motion detection, or both. In (Chen et al., 2004), a dynamic analysis of flames using an RGB/HIS color model was used to determine the existence of fire. However, the decision rule of this method is not good at distinguishing real fire regions from regions of movement or noise since the flame difference is measured only between two consecutive frames. In (Toreyin and Centin, 2007; Toreyin et al., 2006), the boundary of the flames was represented in the wavelet domain, and the high frequency natures of the boundaries of fire regions were used to spatially model flame flicker, yielding good results. In (Celik and Demirel, 2009), a new method of flame detection that used flame pixel color properties was proposed. This method used the YCbCr color space because the RGB color

space has illumination dependence disadvantages. This means that if the illumination in the image changes, the fire pixel classification rules will not perform well. In (Ko et al., 2009), fire detection based on vision sensors and support vector machines was proposed. In the study, a non-linear classification method using support vector machines and luminescence maps was proposed, showing robust results for flame detection. In (Borges and Izquierdo, 2010), a probabilistic model for color-based fire detection was utilized to extract candidate fire regions. In addition, four parameters were extracted from the features of candidate fire regions, such as area size, surface coarseness, boundary roughness, and skewness. Moreover, a Bayes classifier was used to distinguish between fire and non-fire. In (Han and Lee, 2006), the authors proposed the fire and smoke detection system in the tunnel environment. This algorithm detects the fire by comparing image of normal state with input image using color information. In addition, it automatically detects smoke using motion direction, edge detection, and comparison of color information of input images. However, this algorithm uses many ad-hoc parameters, prohibiting for applying to dynamic fire situations. A real-time fire detection method using the statistical color model and foreground object information was proposed in (Celik et al., 2007). The authors introduced the statistic color model for generic fire model. Some of the above algorithms were applied to real systems with considerable success. However, each of these methods still had limited application and lacked enough robustness. In order to enhance the performance of fire detection, we propose an efficient four-stage fire detection approach. In the first stage, movement-containing regions are detected using an adaptive Gaussian mixture model. In the second stage, fire and non-fire regions are segmented using fuzzy c-means (FCM) clustering. In the third stage, additional features are extracted from

* Corresponding author. Tel.: +82 052 259 2217; fax: +82 052 259 1687.

E-mail addresses: jmkim07@ulsan.ac.kr, jongmyon.kim@gmail.com (J.-M. Kim).

the tempo-spatial characteristics of fire regions. In the final stage, fire is classified using support vector machines (SVM). Experimental results indicate that the proposed approach outperforms existing fire detection algorithms in terms of accuracy of fire detection, providing a low false alarm rate and high reliability in both indoor and outdoor test videos.

The rest of this paper is organized as follows. Section 2 introduces the features of fire. Section 3 describes the proposed four-stage fire detection algorithm. Section 4 discusses experimental results of the proposed method and compares the performance of the proposed method with those of other fire detection algorithms, and Section 5 presents conclusions of the study.

2. Features of fire

The features of fire play an important role in the development of fire detection systems since these features are used to distinguish between fire and non-fire. In this study, important characteristics of fire such as color, region, time, and space were analyzed. In practice, most fuels will burn under appropriate conditions, reacting with oxygen from the air, generating combustion products, emitting light, and releasing heat. When fire is produced, it usually arises from a stable location, drifting upward in a diffuse manner. The color usually ranges from red to yellow and may turn white when the temperature is very high. The size, area, shape and number of fire regions in an image vary from frame to frame. The surfaces and contours of fire regions are usually rough.

3. The proposed fire detection algorithm

A flowchart of the proposed fire detection approach is depicted in Fig. 1. The proposed algorithm is comprised of four stages: (1) moving region detection using an adaptive mixture Gaussian model, (2) fire color segmentation using FCM clustering, (3) parameter extraction from the tempo-spatial characteristics of fire regions and (4) fire identification using support vector machines. The approach is also composed of the two phases of training and classification. In the training phase, the training data sets are extracted from the training videos and used to train the SVM. In the classification phase, the trained SVM is utilized to distinguish between fire and non-fire. In the following sections, the proposed fire detection algorithm is analyzed in detail.

3.1. Moving region detection

The detection of moving regions is a fundamental key in video fire detection, which is the first stage of the proposed method. Several methods have been proposed to detect moving regions, with background subtraction being the most typical. As the name suggests, background subtraction is the process of separating out foreground objects from the background in a sequence of video frames. Many different techniques have been proposed with different strengths and weaknesses in terms of performance and computational requirements (Piccardi, 2004). Among these methods, the Gaussian mixture model (GMM) that was first introduced by Stauffer and Grimson in 1999 is the most widely used method for background subtraction due to its speed, simplicity and ease of implementation (Stauffer and Grimson, 1999; Stauffer and Grimson, 2000). In this method, each pixel is modeled as a mixture of Gaussian distributions, and any pixel intensity value that does not fit into one of the modeled Gaussian distributions is marked as a foreground pixel. In order to achieve high performance, we utilize an adaptive Gaussian mixture model (KaewTraKulPong and Bowden, 2002) because it can successfully manage illumination

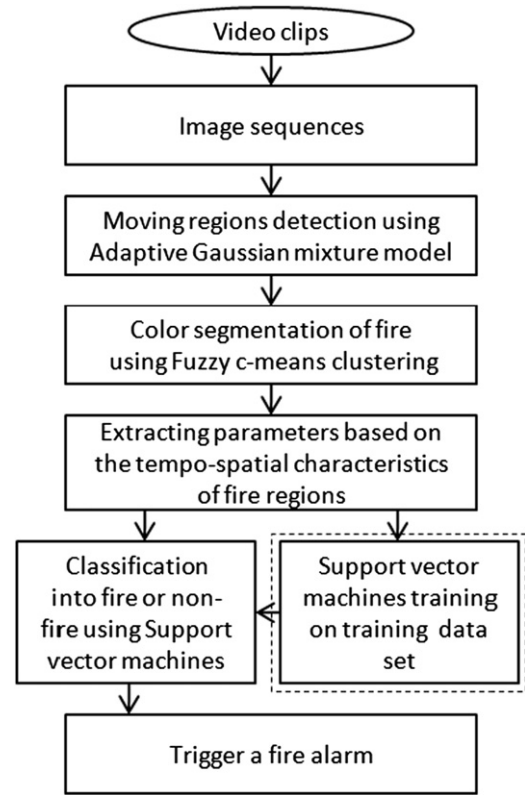


Fig. 1. A flowchart of the proposed video-based fire detection approach.

changes and reduces the effects of small repetitive motions such as moving vegetation and small camera displacement.

In GMM, each pixel location in RGB color space is modeled with a set of K Gaussian distributions. The probability of observing the current pixel value X_t at the time t is defined as follows:

$$P(X_t) = \sum_{k=1}^K \omega_{k,t} \eta(X_t, \mu_{k,t}, \Sigma_{k,t}), \quad (1)$$

where K is the number of Gaussian components per pixel, $\omega_{k,t}$ is the estimate of the weight parameter of the k th Gaussian component at the time t , $\mu_{k,t}$ and $\Sigma_{k,t}$ are the mean and the covariance matrix of the k th Gaussian component in the mixture at time t , and $\eta(X_t, \mu_{k,t}, \Sigma_{k,t})$ is the Gaussian probability density function which is defined as follows:

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} e^{-(1/2)(X_t - \mu)^T \Sigma^{-1} (X_t - \mu)}, \quad (2)$$

where $d=3$ because X_t is in the RGB color space, and $\Sigma_{k,t} = \sigma_k^2 I$. This assumes that the red, green and blue pixel values in RGB color space are independent and have the same variances.

The mixture of Gaussians actually models both foreground pixels and background pixels without distinction; that is, some mixture components model foreground pixels, while others model background pixels. In order to identify the foreground pixels, the background pixels are determined as follows: K distributions are sorted based on the fitness value ω_k/σ_k , and then the first B Gaussian distributions are used as a model of the image background, where B is estimated as follows:

$$B = \arg \min_b \left(\sum_{j=1}^b \omega_j > T \right), \quad (3)$$

where T is the threshold. The other distributions are considered to represent a foreground pixel. The values of the weight ω_k , the mean μ_k , and the covariance σ_k are updated using online Expectation Maximization (EM) algorithms. This information is represented in the detail in (KaewTraKulPong and Bowden, 2002).

Fig. 2 shows the results of the adaptive Gaussian mixture model with and without morphological operations for detecting moving regions. Fig. 2A shows the original fire image. In general, the boundary of flame tends to fluctuate continuously and the inner region of flame tends to static. Therefore, the foreground flame region can be detected easily in the initial stage of fire. However, as times go on, the inner regions of flame are merged into backgrounds. Thus, the holes (or 'ghost') might exist in the foreground as shown in Fig. 2B. In order to improve the quality of the movement regions, adaptive GMM method was utilized as shown in Fig. 2C. In addition, morphological operations (dilation and erosion) were used to remove some inherent noise and smooth the result as shown in Fig. 2D. However, the moving regions may consist of fire or non-fire objects, such as smoke, people or objects. FCM clustering is utilized to cluster only candidate fire regions from these moving regions.

3.2. Color segmentation of fire using the FCM algorithm

In practice, there are several objects in videos which move in addition to fire, such as people, vehicles, birds, clouds, and smoke. However, the colors of these objects generally will differ from the color of fire. Thus, we use color segmentation of fire in this study. The basic idea is composed of two steps: (1) the pixels in the moving regions are distributed into groups, and then (2) groups having colors similar to fire are selected. To accomplish this, the well-known FCM algorithm is employed (Bezdek, 1981; Bezdek et al., 2005). In addition, we use the CIE LAB color space to construct a generic chrominance model for fire pixel segmentation instead of red-green-blue (RGB) color space. In contrast to RGB color space, CIE LAB color space is completely device-independent and makes it possible to separate luminance/illumination from chrominance information. The CIE LAB color components can be

obtained by converting from RGB to CIE LAB (Bhaskaran and Konstantinides, 1997) and include the three components L , A and B , where L indicates the lightness of pixels, A and B indicate the colors of the pixels. The chrominance components A and B are inputs for the FCM algorithm, while the output is the clusters of pixels in the moving regions. The steps used in the FCM algorithm are as follows:

1. Compute the number of groups c and initialize the centroid $V^{(0)} = \{v_1^{(0)}, v_2^{(0)}, \dots, v_c^{(0)}\}$.
2. Compute the membership values u_{ij} for each data element using the following equation:

$$u_{ij} = \left[\sum_{k=1}^c \left(\frac{d^2(x_j, v_i)}{d^2(x_j, v_k)} \right)^{1/(m-1)} \right]^{-1}, \quad (4)$$

where m represents the degree of fuzziness.

3. Update the centroid value v_i as follows:

$$v_i = \frac{\sum_{j=1}^n u_{ij}^m x_j}{\sum_{j=1}^n u_{ij}^m}, \quad 1 \leq i \leq c \quad (5)$$

4. Evaluate the terminating condition $\max_{1 \leq i \leq c} \|v_i^{(t)} - v_i^{(t-1)}\| < \varepsilon$ (where $\|\cdot\|$ is the Euclidean norm). The iteration stops when this is satisfied; otherwise go to step 2.
5. Assign all pixels to a cluster according to the corresponding maximum membership values.

In order to improve the accuracy of clustering, it is necessary to compute the precise number of clusters and the initial values of the centroids in the clusters. To compute the initialization of the aforementioned parameters, we utilized the empirical method of (Tan and Isa, 2010). The value of the centroid of each cluster is compared with the color of fire. The clusters with centroid values close to the color of fire are selected for processing in the next step. If there is no suitable centroid, it can be concluded that the objects in the moving regions are not fire.

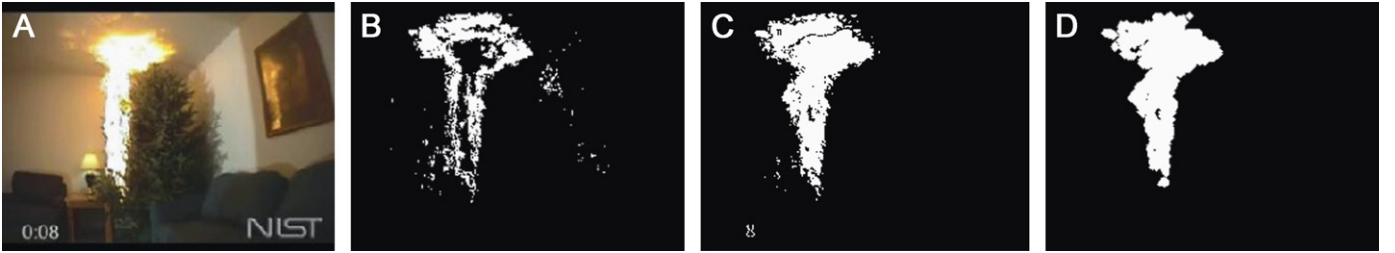


Fig. 2. (A) original image; (B) movement-containing regions between consecutive frames; (C) resulting image with the adaptive GMM method and (D) resulting image with both the adaptive GMM method and morphological operations.

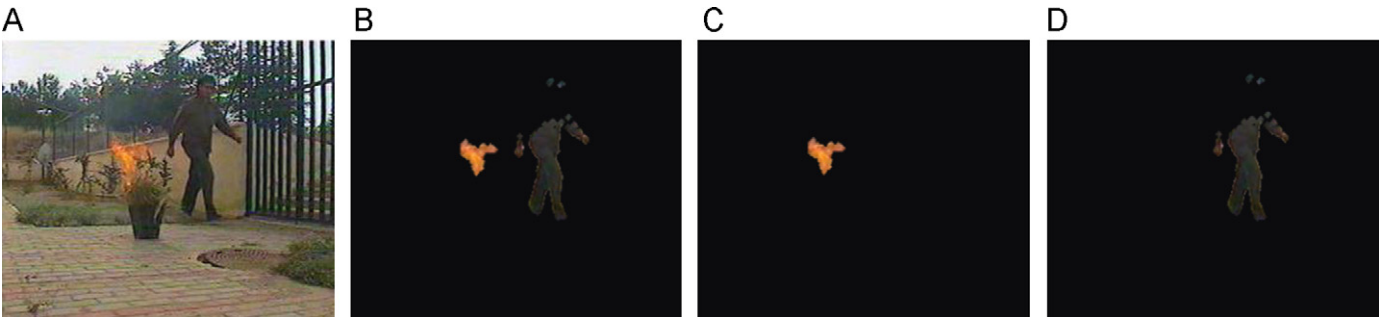


Fig. 3. Segmentation steps of fire color using the fuzzy c-means algorithm: (A) original image; (B) movement-containing regions; (C) candidate fire regions and (D) non-fire regions.

Fig. 3 shows segmentation results based on fire color from the movement-containing regions using the FCM algorithm. The original frame from the video clip and the movement-containing regions are shown in Fig. 3(A) and (B), respectively. The candidate regions of fire are shown in Fig. 3(C), and (D) presents the non-fire regions.

3.3. Parameter extraction

The selected candidate regions of fire, shown in Fig. 3(C), may still be non-fire because several moving objects have the same colors as that of fire, such as red vehicles, vehicle brake lights, or persons carrying red objects or wearing red clothes. By investigating several videos, we observed that the temporal and spatial features of fire regions vary during the manifestation of the fire. Thus, we extract the following features of the candidate regions to identify them as either fire or non-fire.

Let $X = \{x_1, x_2, x_3, \dots, x_n\}$ represent the data set. The mean and variance values of the data set are defined as follows:

$$M_x = \frac{1}{n} \sum_{i=1}^n x_i, \quad (6)$$

$$V_x = \frac{1}{n} \sum_{i=1}^n |x_i - M_x|^2, \quad (7)$$

where M_x is the mean value of X , and V_x is the variance value of the data set X .

3.3.1. Area randomness

We know that the size of a fire region randomly changes from frame to frame. In contrast, non-fire regions do not show such random changes. Using this feature, we can extract parameters by calculating the difference in area between two consecutive frames and then summing these differences among the previous frames and the current frame. The changes in the sizes of the candidate regions from frame to frame are calculated using the following rule:

$$\Delta A_n = |A_n - A_{n-1}|, \quad (8)$$

where ΔA is the difference size between the areas of the candidate regions in the n -th and $(n-1)$ -th frames. Using all of the ΔA s, the mean value (M_A) and variance value (V_A) are calculated using Eqs. (6) and (7).

3.3.2. Surface roughness

Unlike moving non-fire regions with the same color as fire, the intensity of the fire region pixels in a gray image is always non-uniform, and the surfaces of these fire regions are coarse. Therefore, at every frame in the video sequence, we can calculate feature parameters of the fire (e.g., mean (M_I) and variance (V_I) of intensity of all the pixels in the candidate regions) using Eqs. (6) and (7).

3.3.3. Contour roughness

Different from non-fire moving regions, the contour of moving fire regions is usually rough and coarse. In order to utilize this feature, we perform a simulation using different shape descriptor techniques and find that Hu invariant moments (Hu 1962) are very effective for the description of the fire regions because they clearly distinguish between moving fire and non-fire regions. For every new frame, we calculate the Hu invariant moments of the candidate moving regions ($H_1, H_2, H_3, H_4, H_5, H_6$, and H_7).

3.3.4. Motion estimation using motion templates

Motion plays an important role in video fire detection, providing both spatial and temporal characteristics of fire regions. Hence, some parameters are extracted from the motion features of fire. In order to accomplish this, we use motion templates to estimate the motion of candidate fire regions. Motion templates were introduced by Bobick and Davis (Bobick and Davis, 1996; Davis and Bobick, 1997) and were further developed by Davis (Davis and Bradski, 2009). They are an effective way to track general movement and are especially applicable to the movement of fire. Using motion templates, two parameters can be extracted from every new frame: the magnitude (M) and the orientation (O) of the moving vector of the candidate regions. On the other hand, it is not sufficient to distinguish between moving fire regions and moving non-fire regions because there are some moving regions that have the same values. Therefore, the mean and variance of M_v and O_v are calculated using Eqs. (6) and (7). As a result, six parameters are extracted from the features of the motion vector for further analysis: the magnitude (M_v), orientation (O_v), mean of M_v (M_M), variance of M_v (V_M), mean of O_v (M_O), and variance of O_v (V_O).

3.4. Fire identification using support vector machines

Support vector machines are supervised learning techniques and were introduced by Vapnik (Vapnik, 1982, 1995). SVM is widely applied to many fields of pattern recognition (Borges, 1998). The SVM is not only capable of learning in high-dimensional spaces but also can provide high performance with limited training data sets. Therefore, many techniques have been proposed to improve the performance of SVM (Platt, 1998; Cristianini and Shawe-Taylor, 2000). The basic idea of the linear SVM is to create a suitable hyper-plane to divide a given data set into two parts with maximum margin. After that, the SVM is utilized to classify unlabeled data sets. However, in practice, many data that are not fully linearly or nonlinearly separable. For that reason, no hyper-plane may exist that can split the data into two parts. In order to solve this problem, we used a non-linear SVM.

In the previous section, we described a set of 17 parameters extracted from the spatial and temporal characteristics of fire: $\mathbf{sv}_i = [M_A, V_A, M_I, V_I, H_1, H_2, H_3, H_4, H_5, H_6, H_7, M_v, O_v, M_M, V_M, M_O, V_O]$. These 17 parameters are used as an input feature vector to train the SVM classifier, which is then used to produce a fire alarm in the video frame. In order to accomplish this, both the training data set (SV) and the kernel function $k(\mathbf{sv}_i, \mathbf{sv}_j)$ are required. The training data set $\mathbf{SV} = \{\mathbf{sv}_1, \mathbf{sv}_2, \mathbf{sv}_3, \dots, \mathbf{sv}_L\}$ is obtained from several video clips of fire, non-fire, or moving objects in different scenes. The selection of the kernel function is very important for the training stage because it affects the accuracy of the SVM classification. If the kernel function is suitable for the data set, the classification result is going to be good, and vice versa. Through the analysis of several video clips, we selected the radial basis function kernel because it is the most useful for the features of fire. This kernel is defined as follows:

$$k(\mathbf{sv}_i, \mathbf{sv}_j) = \exp\left(-\frac{\|\mathbf{sv}_i - \mathbf{sv}_j\|^2}{2\delta^2}\right), \quad (9)$$

where $k(\mathbf{sv}_i, \mathbf{sv}_j)$ is the kernel function, \mathbf{sv}_i and \mathbf{sv}_j are the input feature vectors, and δ is a parameter set by the user that determines the width of the effective basis kernel function. If small δ values are used, overtraining occurs with the basis function wrapped tightly around the data points. If large δ values are used, the basis function forms an oval around the points without defining the shape or pattern. In this study, our simulations showed that the best performance was achieved when the value of δ ranged from 5 to 10. Therefore, the default value for δ was set

at 7.5. The trained SVM was then used to distinguish between fire and non-fire.

4. Experimental results

4.1. Experimental environment

The performance of the proposed fire detection algorithm is compared with those of four state-of-the-art algorithms: fire detection in video sequences using a generic color model (*Algorithm 1*) (Celik and Demirel, 2009), fire detection based on vision sensors and support vector machines (*Algorithm 2*) (Ko et al., 2009), a computer vision-based method for real-time fire and flame detection (*Algorithm 3*) (Toreyin et al., 2006), and a probabilistic approach for vision-based fire detection in videos (*Algorithm 4*) (Borges and Izquierdo, 2010). The proposed and existing algorithms are implemented using MATLAB 7.5 and are tested on a Pentium Quad-Core 2.8 GHz PC platform. In order to train the support vector machines, we use several training movies which include indoor fire, outdoor fire, non-fire, and moving objects that are the same color as fire and we utilized 25,000 fire-containing frames and 15,000 non-fire-containing frames, respectively. In addition, the test is performed using ten video sequences, each of which has a 320×240 image size, as shown in Fig. 4. Movies 1, 6 and 8 are indoor fires, 2–5 and 7 are outdoor fires, 9–11 are indoor non-fires, and the others are outdoor non-fires.

Through the analysis of several videos, we empirically set the feature parameter values for the proposed approach: the number

of Gaussian components per pixel K value in Eq. (1) is 3, the fuzziness coefficient m value in Eq. (4) and (5) are 2, the terminal condition ε value for FCM is 0.001 and the scaling factor δ value is 7.5 in Eq. (9).

4.2. Experimental evaluation

The comparison results are presented in Table 1, where the true positive (TP) is the rate of correctly detecting a real fire as a fire and false positive (FP) is the rate of recognizing a real fire as a non-fire, respectively. For Movie 1 and Movie 8, the accuracies of all algorithms are the highest because it contains a clearly visible fire in an indoor space. For Movie 3 and Movie 4, since the fire is blurred at some frame in the movie, the accuracies of all algorithms are relatively low. For all fire-containing movies, the proposed approach outperforms the other algorithms in terms of TP and FP for both the indoor and outdoor test videos, showing an average TP of 94.78% versus TPs of 94.13%, 94.25%, 94.23%, and 94.41% for other methods.

Table 2 shows additional data on the accuracy of fire detection in four videos with no fire, where the true negative (TN) is the rate of correctly detecting a non-fire as a non-fire and false negative (FN) is the rate of recognizing a non-fire as a fire, respectively. For Movies 9 and 12, FN is zero because the color of the moving regions is different from the color of the fire regions. For Movie 10, the moving regions include cars with colors similar to that of fire. Thus, FN is high for all algorithms. For Movie 11, the moving regions are a red color, but the regions are uniform, and the shape and the movement are different from



Fig. 4. Examples of test movies.

Table 1

Fire detection results of the proposed algorithm and conventional algorithms on positive fire movies.

Number of frames		Algorithm 1		Algorithm 2		Algorithm 3		Algorithm 4		Proposed	
		TP (%)	FP (%)	TP (%)	FP (%)	TP (%)	FP (%)	TP (%)	FP (%)	TP (%)	FP (%)
Movie 1	75	96.00	4.00	96.00	4.00	96.00	4.00	96.00	4.00	96.00	4.00
Movie 2	190	94.74	3.26	95.26	4.74	94.74	5.26	95.26	4.74	95.79	4.21
Movie 3	200	92.50	7.50	92.00	8.00	92.50	7.50	93.00	7.00	93.00	7.00
Movie 4	400	92.25	7.75	92.50	7.50	92.75	7.25	92.25	7.25	93.00	7.00
Movie 5	200	94.50	5.50	94.50	5.50	94.00	6.00	94.50	5.50	95.00	5.00
Movie 6	165	92.73	7.27	93.94	6.06	93.33	6.67	93.94	6.06	94.55	5.45
Movie 7	259	94.59	5.41	94.21	5.79	94.59	5.41	94.59	5.41	94.98	5.02
Movie 8	520	95.77	4.23	95.58	4.42	95.96	4.04	95.77	4.23	95.96	4.04
Average		94.13	5.87	94.25	5.75	94.23	5.77	94.41	5.59	94.78	5.22

Table 2

False fire detection results of the proposed algorithm and conventional algorithms on negative movies.

Number of frames		Algorithm 1		Algorithm 2		Algorithm 3		Algorithm 4		Proposed	
		TN (%)	FN (%)	TN (%)	FN (%)	TN (%)	FN (%)	TN (%)	FN (%)	TN (%)	FN (%)
Movie 9	151	100.0	0.00	100.0	0.00	100.0	0.00	100.0	0.00	100.0	0.00
Movie 10	286	95.10	4.90	95.45	4.55	95.45	4.55	96.15	3.85	96.50	3.50
Movie 11	112	95.54	4.46	95.54	4.46	96.43	3.57	96.4	3.57	98.21	1.79
Movie 12	616	100.0	0.00	100.0	0.00	100.0	0.00	100.0	0.00	100.0	0.00
Average		97.66	2.34	97.75	2.25	97.97	2.03	98.15	1.85	98.68	1.32

Table 3

The result of the Wilcoxon signed-rank test between the proposed technique and conventional algorithm 1.

	Proposed (X)	Algorithm 1 (Y1)	Sign of X-Y1	X-Y1	Absolute X-Y1	Rank of Absolute	Signed Rank
Movie 1	96.00	96.00		0.00	0.00		
Movie 2	95.79	94.74	+	1.05	1.05	6	6
Movie 3	93.00	92.50	+	0.50	0.50	3.5	3.5
Movie 4	93.00	92.25	+	0.75	0.75	5	5
Movie 5	95.00	94.50	+	0.50	0.50	3.5	3.5
Movie 6	94.55	92.73	+	1.82	1.82	7	7
Movie 7	94.98	94.59	+	0.39	0.39	2	2
Movie 8	95.96	95.77	+	0.19	0.19	1	1

Table 4

The result of the Wilcoxon signed-rank test between the proposed technique and conventional algorithm 2.

	Proposed (X)	Algorithm 1 (Y2)	Sign of X-Y2	X-Y2	Absolute X-Y2	Rank of Absolute	Signed Rank
Movie 1	96.00	96.00		0.00	0.00		
Movie 2	95.79	95.26	+	0.53	0.53	4	4
Movie 3	93.00	92.00	+	1.00	1.00	7	7
Movie 4	93.00	92.50	+	0.50	0.50	2.5	2.5
Movie 5	95.00	94.50	+	0.50	0.50	2.5	2.5
Movie 6	94.55	93.94	+	0.61	0.61	5	5
Movie 7	94.98	94.21	+	0.77	0.77	6	6
Movie 8	95.96	95.58	+	0.38	0.38	1	1

those in the fire region. Thus, FN of the proposed algorithm is low. For all non-fire-containing movies, the proposed approach outperforms the four conventional algorithms in terms of reducing the false fire detection rate, showing an average FN of 1.32% versus FNs of 2.34%, 2.25%, 2.03%, and 1.85% for other methods.

In addition, we used a Wilcoxon signed-rank test that is a non-parametric statistical hypothesis test to evaluate the performance of the proposed technique. In this Wilcoxon signed-rank test, two related samples are compared to assess whether their population means differ (Wilcoxon, 1945; Corder and Foreman, 2009). Tables 3–6 show the results of the Wilcoxon signed-rank test between our proposed technique (X) and conventional techniques (Y1, Y2, Y3, Y4).

For example, Table 1 shows the result of the Wilcoxon signed-rank test between the proposed technique (X) and conventional algorithm 1 (Y1). The Wilcoxon signed-rank test includes 6 steps:

1. The sign of X-Y1 is denoted in the Sign column by either (+) or (−). If X and Y1 are equal, the value is then thrown out.
2. The values of X-Y1 are given in the next two columns.
3. The last two columns are the ranks. The Absolute Rank column has no signs, and the Signed Rank column gives the ranks along with their signs.
4. The data is ranked from the smallest value to the largest value. In the case of a tie, ranks are added together and divided by the

Table 5

The result of the Wilcoxon signed-rank test between the proposed technique and conventional algorithm 3.

	Proposed (X)	Algorithm 1 (Y3)	Sign of X-Y3	X-Y3	Absolute X-Y3	Rank of Absolute	Signed Rank
Movie 1	96.00	96.00		0.00	0.00		
Movie 2	95.79	94.74	+	1.05	1.05	5	5
Movie 3	93.00	92.50	+	0.50	0.50	3	3
Movie 4	93.00	92.75	+	0.25	0.25	1	1
Movie 5	95.00	94.00	+	1.00	1.00	4	4
Movie 6	94.55	93.33	+	1.22	1.22	6	6
Movie 7	94.98	94.59	+	0.39	0.39	2	2
Movie 8	95.96	95.96		0.00	0.00		

Table 6

The result of the Wilcoxon signed-rank test between the proposed technique and conventional algorithm 4.

	Proposed (X)	Algorithm 4 (Y4)	Sign of X-Y4	X-Y4	Absolute X-Y4	Rank of Absolute	Signed Rank
Movie 1	96.00	96.00		0.00	0.00		
Movie 2	95.79	95.26	+	0.53	0.53	4	4
Movie 3	93.00	93.00		0.00	0.00		
Movie 4	93.00	92.25	+	0.75	0.75	6	6
Movie 5	95.00	94.50	+	0.50	0.50	3	3
Movie 6	94.55	93.94	+	0.61	0.61	5	5
Movie 7	94.98	94.59	+	0.39	0.39	2	2
Movie 8	95.96	95.77	+	0.19	0.19	1	1

number of ties. For example, in this data, there were two instances of the value 0.50. The ranks corresponding to 0.50 are 3 and 4. The sum of these ranks is 7. After dividing by the number of ties, we get a mean rank of 3.5, and this value is assigned to both instances of 0.50.

- The test statistic, W_+ , is given by the sum of all of the positive values in the Signed Rank column. The test statistic, W_- , is given by the sum of all of the negative values in the Signed Rank column. For this example, $W_+ = 28$ and $W_- = 0$. The minimum of these is 0.
- Lastly, this test statistic was analyzed using a table of critical values for the Wilcoxon test. If the test statistic is less than or equal to the critical value based on the number of observations n , then the null hypothesis is rejected for the alternative hypothesis. Otherwise, the null hypothesis is not rejected. In this case, the test statistic is $W = 0$ and the critical value is 2 for a two-tailed p -value of 0.05. The test statistic is less than this to be significant at this level. Therefore, the proposed technique outperforms conventional algorithm 1.

We have similar results of the Wilcoxon signed-rank test between our proposed technique and other techniques. Overall, our proposed technique outperforms conventional techniques.

Overall, the proposed approach clearly outperformed other conventional algorithms by consistently increasing the accuracy of fire detection and decreasing the error of false fire detection for each video. The proposed method was also tested with movies that were downloaded from two websites: http://www.ultimatechase.com/Fire_Video.htm and <http://signal.ee.bilkent.edu.tr/VisiFire/Demo/FireClips>. The results were the same.

Table 7 shows an additional result of computational time of our proposed algorithm and conventional algorithms. The proposed algorithm requires more computational time than conventional algorithms due to the four stages of the proposed algorithm is more complex than others. Since today's high performance processors including NVIDIA GPUs (Graphics Processing Unit) and TI DSPs (Digital Signal Processor) can support these algorithms in real time, the accuracy of fire detection is more important. In the future, we will work on reducing the computational time of our proposed algorithm while maintaining the accuracy.

Table 7

Computational times of the proposed and conventional algorithms.

Algorithms	Computational Time Average Processing Time Per Frame (s)
Algorithm 1	0.359
Algorithm 2	1.164
Algorithm 3	1.096
Algorithm 4	0.892
Proposed Algorithm	1.486

5. Conclusions

In this paper, we proposed an effective novel approach for fire detection in video sequences using segmentation and classification algorithms. The proposed approach consists of four stages: (1) an adaptive mixture of Gaussian models for detecting moving regions, (2) the FCM algorithm for selecting candidate fire regions from other moving regions based on the color of fire, (3) parameter extraction for extracting special parameters based on the tempo-spatial characteristics of fire regions, and (4) an SVM for distinguishing between fire and non-fire. Experimental results showed that the proposed approach outperforms other state-of-the-art fire detection algorithms in terms of fire detection accuracy, providing a low false alarm rate and high reliability in open and large spaces.

Acknowledgment

This work was supported by 2012 Research Fund of University of Ulsan.

References

- Borges, P.V.K., Izquierdo, E., 2010. A probabilistic approach for vision-based fire detection in videos. *IEEE. Trans. Circuits Syst. Video Technol.* 20, 721–731.
- Bezdek, J.C., 1981. *Pattern recognition with fuzzy objective function algorithms*. Plenum Press, New York.
- Bezdek, J.C., Keller, J., Krisnapuram, R., Pal, N., 2005. *Fuzzy models and algorithms for pattern recognition and image processing*. Springer.

- Bhaskaran, V., Konstantinides, K., 1997. Image and video compression standards: algorithms and architectures. Kluwer Academic Publishers 410–412.
- Bobick, A., Davis, J., 1996. Real-time Recognition of Activity using Temporal Templates. In: IEEE Workshop on Applications of Computer Vision, pp. 39–42.
- Burges, C.J.C., 1998. A tutorial on support vector machines for pattern recognition. *J. Knowl. Discovery Data Mining* 2, 121–167.
- Chen, T.H., Wu, P.H., Chiou, Y.C., 2004. An Early Fire-Detection Method based on Image Processing. In: IEEE. Int. Conf. on Image Processing, vol. 3, pp. 1707–1710.
- Celik, T., Demirel, H., 2009. Fire detection in video sequences using a generic color model. *Fire Saf. J.* 44, 147–158.
- Celik, T., Demirel, H., Ozkaramanli, H., Uyguroglu, M., 2007. Fire detection using statistical color model in video sequences. *J. Visual Commun. Rep.* 18, 176–185.
- Cristianini, N., Shawe-Taylor, J., 2000. An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press.
- Corder, G.W., Foreman, D.J., 2009. Nonparametric statistics for non-statisticians: a step-by-step approach. Wiley.
- Davis, J., Bobick, A., 1997. The Representation and Recognition of Action using Temporal Templates. Technical Report 402, MIT Media Lab, Cambridge, MA, 1.
- Davis, J., Bradski, G., 2009. Real-time Motion Template Gradients using Intel CVLib. ICCV Workshop on Framerate Vision.
- Han, D., Lee, B., 2006. Development of early tunnel fire detection algorithm using the image processing. *Adv. Visual Comput.*, 39–48.
- Hu, M.K., 1962. Visual pattern recognition by moment invariants. *IRE Trans. Inf. Theory* 8, 179–187.
- Jones, W.W., 2004. An Algorithm for Fast and Reliable Fire Detection. In: 8th Fire Suppression and Detection Research Application Symposium, pp. 1–3.
- Ko, B.C., Cheong, K.H., Nam, J.Y., 2009. Fire detection based on vision sensor a support vector machines. *Fire Saf. J.* 44, 322–329.
- KaewTraKulPong, P., Bowden, R., 2002. An Improved Adaptive Background Mixture Model for Real-Time Tracking with Shadow Detection. In: Proc. 2nd European Workshop on Advanced Video-Based Surveillance System, AVBS01, pp. 135–144.
- Piccardi, M., 2004. Background Subtraction Techniques: A Review. In: IEEE. Int. Conf. on Systems. Man and Cybernetics, pp. 3099–3104.
- Platt, J.C., 1998. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. Microsoft Research Technical Report MSR-TR-98-14.
- Stauffer, C., Grimson, W.E.L., 1999. Adaptive Background Mixture Models for Real-Time Tracking. In: IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 246–252.
- Stauffer, C., Grimson, W.E.L., 2000. Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Machines Intel.* 22, 747–757.
- Toreyin, B.U., Centin, A.E., 2007. Online Detection of Fire in Video. In: IEEE. Conf. on Computer Vision and Pattern Recognition, pp. 1–5.
- Toreyin, B.U., Dedeoglu, Y., Gudukbay, U., Centin, A.E., 2006. Computer vision-based method for real-time fire and flame detection. *Pattern Recog. Lett.* 27, 49–58.
- Tan, K.S., Isa, N.A.M., 2010. Color image segmentation using histogram thresholding-fuzzy c-means hybrid approach. *Pattern Recognition* 44, 1–15.
- Vapnik, V., 1982. Estimation of dependences based on empirical data. Springer-Verlag.
- Vapnik, V., 1995. Statistical learning theory. Springer, New York.
- Wilcoxon, F., 1945. Individual comparisons by ranking methods. *Biometrics Bull.* 80–83.