

# FLAME DETECTION METHOD IN VIDEO USING COVARIANCE DESCRIPTORS

*Yusuf Hakan Habiboğlu*

Boğaz Komutanlığı  
Turkish Navy  
17000, Çanakkale, Turkey

*Osman Günay, A. Enis Çetin*

Bilkent University  
Department of Electrical and Electronics Engineering  
06800, Bilkent, Ankara, Turkey

## ABSTRACT

Video fire detection system which uses a spatio-temporal covariance matrix of video data is proposed. This system divides the video into spatio-temporal blocks and computes covariance features extracted from these blocks to detect fire. Feature vectors taking advantage of both the spatial and the temporal characteristics of flame colored regions are classified using an SVM classifier which is trained and tested using video data containing flames and flame colored objects. Experimental results are presented.

**Index Terms**— fire detection, covariance descriptors, support vector machines.

## 1. INTRODUCTION

Video based systems can detect uncontrolled fires at an early stage before they turn into catastrophic events. There are several methods in the literature developed for fire detection from images and videos [1] -[7]. In [1], a Gaussian-smoothed histogram is used as a color model. Considering temporal variation of fire and non-fire pixels with fire color probability a heuristic analysis is performed.

An adaptive flame detector is implemented and trained by using weighted majority based online training in [6]. Outputs of Markov models representing the flame and flame colored ordinary moving objects and spatial wavelet analysis of boundaries of flames are used as weak classifiers for training. In [7], HSI color model based segmentation algorithm is used to find fire colored regions. Change detection is used to separate fire colored objects from flames. A method for estimating degree of fire flames is proposed.

Most fire detection systems first find the flame colored regions using background subtraction and flame color analysis. These regions are then analyzed spatially and temporally to detect the irregular and flickering characteristics of flames. In this work, we use a different approach by combining color, spatial and temporal domain information in feature vectors

---

This work is supported by European Commission Seventh Framework Program with EU Grant: 244088(FIRESENSE - Fire Detection and Management through a Multi-Sensor Network for the Protection of Cultural Heritage Areas from the Risk of Fire and Extreme Weather Conditions

for each spatio-temporal block using region covariance descriptors [8]. The blocks are obtained by dividing the flame colored regions into 3D regions that overlap in time.

To the best of our knowledge none of the methods proposed in the literature including our previous works [4, 5] can handle video captured by moving cameras. The proposed covariance matrix approach does not use a background subtraction method that requires a stationary camera for moving flame regions detection and therefore can be used with moving cameras.

## 2. COVARIANCE MATRIX BASED DETECTION ALGORITHM

### 2.1. Chromatic Color Model

In [3] Chen, Wu and Chiou suggested a chromatic model to classify pixel colors. They analyzed fire colored pixels and realized that hue of fire colored pixels is in the range of  $0^\circ$  and  $60^\circ$ . RGB domain equivalent of this condition is

**Condition 1**  $R \geq G > B$

Since fire is a light source its pixel values must be larger than some threshold.  $R_T$  is the threshold for red channel:

**Condition 2**  $R > R_T$

Last rule is about saturation.  $S$  is the saturation value of a pixel and  $S_T$  is the saturation value of this pixel when  $R$  is  $R_T$ .

**Condition 3**  $S > (255 - R)S_T R_T$

We modified this model by excluding condition 3 to reduce the computational cost and used this new model as the first step of our algorithm. This simple chromatic model is sufficient for our purposes because regions satisfying this simple condition are further analyzed by the spatio-temporal covariance descriptors.

### 2.2. Covariance Descriptors for Videos

#### 2.2.1. Definition of Property Sets

Tuzel, Porikli and Meer proposed covariance descriptors and applied this method to object detection and texture classification problems [8]. Let  $\Phi_{i,j}$  be a property vector of the pixel

$(P_{i,j})$  at location  $(i, j)$ . The property set  $S$  is formed by using property vectors,  $\Phi_{i,j}$ , of images. The covariance matrix of the property set  $S$  for an image region can be estimated as follows:

$$\hat{\Sigma} = \frac{1}{N-1} \sum_i \sum_j (\Phi_{i,j} - \bar{\Phi})(\Phi_{i,j} - \bar{\Phi})^T \quad (1)$$

where  $N$  is the number of pixels,  $\bar{\Phi} = \frac{1}{N} \sum_i \sum_j \Phi_{i,j}$  is the mean of the property vector.

As explained in [8], assuming a property set of a region in an image has a wide-sense stationary multivariate normal distribution, covariance descriptors provide an excellent description of the given image region. Wide-sense stationarity is a reasonable assumption for a flame colored image region because such regions do not contain strong edges in video. Therefore, covariance descriptors can be used to model spatial characteristics of fire regions in images. It is experimentally observed that wide-sense stationarity assumption is valid temporally as well. To model the temporal variation and flicker in fire flames we introduce temporally extended covariance descriptors in this article. To the best of our knowledge spatio-temporal parameters have not been used to construct covariance descriptors by other researchers.

Temporally extended covariance descriptors are designed to describe spatio-temporal video blocks. Let  $I(i, j, n)$  be the intensity of  $(i, j)^{th}$  pixel of the  $n^{th}$  image frame of a spatio-temporal block in video and *Red*, *Green*, *Blue* represent the color values of pixels of the block. The property parameters defined in Equation (2) to Equation (5) are used to form a covariance matrix representing spatial information. In addition to spatial parameters we introduce temporal derivatives, *It* and *Itt* which are the first and second derivatives of intensity with respect to time, respectively. By adding these two features to the previous property set, covariance descriptors can be used to define spatio-temporal blocks in video.

$$R_{i,j,n} = Red(i, j, n), \quad G_{i,j,n} = Green(i, j, n), \quad (2)$$

$$B_{i,j,n} = Blue(i, j, n), \quad I_{i,j,n} = I(i, j, n), \quad (3)$$

$$Ix_{i,j,n} = \left| \frac{\partial I(i, j, n)}{\partial i} \right|, \quad Iy_{i,j,n} = \left| \frac{\partial I(i, j, n)}{\partial j} \right|, \quad (4)$$

$$Ixx_{i,j,n} = \left| \frac{\partial^2 I(i, j, n)}{\partial i^2} \right|, \quad Iyy_{i,j,n} = \left| \frac{\partial^2 I(i, j, n)}{\partial j^2} \right|, \quad (5)$$

$$It_{i,j,n} = \left| \frac{\partial I(i, j, n)}{\partial n} \right|, \quad Itt_{i,j,n} = \left| \frac{\partial^2 I(i, j, n)}{\partial n^2} \right| \quad (6)$$

During the implementation of the covariance method, the first derivative of the image is computed by filtering the image with  $[-1 \ 0 \ 1]$  and second derivative is found by filtering the image with  $[1 \ -2 \ 1]$  filters, respectively.

## 2.2.2. Computation of Covariance Values in Spatio-temporal Blocks

In this section details of covariance matrix computation in video is described. We first divide the video into blocks of size  $16 \times 16 \times F_{rate}$  where  $F_{rate}$  is the frame rate of the video. Computing the covariance parameters for each block of the video would be computationally inefficient. We use the simple color model to eliminate the blocks that do not contain any fire colored pixels. Therefore, only pixels corresponding to the non-zero values of the following color mask are used in the selection of blocks. The color mask is defined by the following function:

$$\Psi(i, j, n) = \begin{cases} 1 & \text{if } (R(i, j, n) > G(i, j, n) \text{ and } G(i, j, n) \geq B(i, j, n) \\ & \text{and } R(i, j, n) > R_T = 110) \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

A total of 10 property parameters are used for each pixel satisfying the color condition. However this requires  $\frac{10 \times 11}{2} = 55$  covariance computations. To further reduce the computational cost we compute the covariance values of the pixel property vectors

$$\Phi_C(i, j, n) = \begin{bmatrix} R_{i,j,n} \\ G_{i,j,n} \\ B_{i,j,n} \end{bmatrix}, \quad \Phi_{ST}(i, j, n) = \begin{bmatrix} I_{i,j,n} \\ Ix_{i,j,n} \\ Iy_{i,j,n} \\ Ixx_{i,j,n} \\ Iyy_{i,j,n} \\ It_{i,j,n} \\ Itt_{i,j,n} \end{bmatrix} \quad (8)$$

in a separate manner. Therefore, the property vector  $\Phi_C(i, j, n)$  produces  $\frac{3 \times 4}{2} = 6$  and the property vector  $\Phi_{ST}(i, j, n)$  produces  $\frac{7 \times 8}{2} = 28$  covariance values, respectively and 34 covariance parameters are used in training and testing of the SVM instead of 55 covariance parameters.

We also assume that the size of the image frames in video is 320 by 240. If not the video is scaled to 320 by 240 in order to run the fire detection algorithm in real-time.

## 3. TRAINING AND TESTING

For training and testing,  $16 \times 16 \times F_{rate}$  blocks are extracted from various video clips. The temporal dimension of the blocks are determined by the frame rate parameter  $F_{rate}$  which is between 10 and 25 in our train and test videos. These blocks do not overlap in spatial domain but there is 50% overlap in time domain. This means that classification is not repeated after every frame of the video. After the blocks are constructed, features are extracted and used to form a training set. A support vector machine (SVM) [9] is trained for classification.

The classification is done periodically with the period  $F_{rate}/2$ . This decreases the cost of classification. On the other hand estimating covariance matrix periodically with the formula given in Equation (1) requires accumulating your data at the end of each period. But luckily there is another covariance matrix estimation formula given in Equation (3), that can be started to calculate without waiting for the entire data.

$$\hat{\Sigma}(a, b) = \frac{1}{N-1} \left( \sum_i \sum_j \Phi_{i,j}(a) \Phi_{i,j}(b) - C_N \right) \quad (9)$$

where

$$C_N = \frac{1}{N} \left( \sum_i \sum_j \Phi_{i,j}(a) \right) \left( \sum_i \sum_j \Phi_{i,j}(b) \right)$$

During the implementation, in each spatio-temporal block, the number of chromatically fire-pixels is found. If this number is higher than or equal to  $\frac{2}{5}$  of the number of the elements of block ( $16 \times 16 \times F_{rate}$ ) then that block is classified as a flame colored block. If the number of possible fire-pixels is enough, then classification is done by the SVM classifier using the augmented feature vector described in Section 2.2.

In this article, 7 positive and 10 negative video clips are used for training. Negative video clips contain flame colored regions as shown in Figure 1. For positive videos (video clips containing fire) only parts of the video clips that contain fire are used.

At the final step of our flame detection method a confidence value is determined according to the number of positively classified video blocks and their positions. After every block is classified, if there is no neighbor block classified as fire, the confidence level is set to 1. If there is a single neighbor block, which is classified as fire, then the confidence level is set to 2. If there are more than 2 neighbor blocks classified as fire then the confidence level of that block is set to 3. In Figure 1 there are sample frames after classification.

#### 4. EXPERIMENTAL RESULTS

The proposed system is compared with one of our previous fire detection methods [4]. In the decision process, if the confidence level of any block of the frame is greater than or equal to 3 then that frame is marked as a fire containing frame. The method described in [4] has a similar confidence level metric to determine the alarm level. Results are summarized in Table 1 and Table 2 in terms of the true detection and the false alarm ratios, respectively. The true detection rate in a given video clip is defined as the number of correctly classified frames containing fire divided by the total number of frames which contain fire. Similarly, the false alarm rate in a given test video is defined as the number of misclassified



(a) true detection



(c) true rejection



(d) false alarm

**Fig. 1.** Sample image frames from test videos.

frames, which do not contain fire divided by the total number of frames which do not contain fire.

In the experiments 19 video clips are used to test the proposed system. First 11 videos do contain fire and the remaining 8 videos do not contain fire and flames but contain flame colored regions. In Table 1 the true detection rates of the two algorithms are presented for the 11 videos containing fire. In Table 2 the false alarm rates of the two algorithms are presented for the 8 videos that do not contain fire. According to the results, our system is able to classify all video files containing fire, with a reasonable false alarm ratio in videos without fire. Although the true detection rate is low in some videos, we do not need to detect all fire frames correctly to issue an alarm. The first detection time is less than 2 seconds in all the test video clips except for one video clip containing a small uncontrolled fire in which the fire is detected in 7 seconds.

Compared to the previous method the new method has higher true detection rate in all of the videos that contain

actual fires. In some of the videos that do not contain fire the older method has a lower false alarm rate than the new method. Some of the positive videos in the test set are recorded with hand-held moving cameras and since the old method assumes a stationary camera for background subtraction it cannot correctly classify most of the actual fire regions.

The proposed method is computationally efficient. The experiments are performed with a PC that has a Core 2 Duo 2.2 GHz processor and the video clips are generally processed around 20 fps when image frames of size 320 by 240 are used. The detection resolution of the algorithm is determined by the video block size. Since we require three neighboring blocks to reach the highest confidence level the fire should occupy a region of size 32 by 32 in video.

**Table 1.** Comparison of our new method with the previous method proposed in [4] in terms of true detection rates in video clips that contain fire.

Video name	True Detection Rates	
	New Method	Old Method
posVideo1	161/ 293 (54.9%)	0/ 293 ( 0.0%)
posVideo2	413/ 510 (81.0%)	0/ 510 ( 0.0%)
posVideo3	310/ 381 (81.4%)	0/ 381 ( 0.0%)
posVideo4	1643/ 1655 (99.3%)	627/ 1655 (37.9%)
posVideo5	495/ 547 (90.5%)	404/ 547 (73.9%)
posVideo6	501/ 513 (97.7%)	0/ 513 ( 0.0%)
posVideo7	651/ 663 (98.2%)	64/ 663 ( 9.7%)
posVideo8	223/ 235 (94.9%)	181/ 235 (77.0%)
posVideo9	35/ 178 (19.7%)	23/ 178 (12.9%)
posVideo10	234/ 246 (95.1%)	139/ 246 (56.5%)
posVideo11	196/ 208 (94.2%)	164/ 208 (78.8%)

**Table 2.** Comparison of our new method with the previous method proposed in [4] in terms of false alarm rates in video clips that do not contain fire.

Video name	False Alarm Rates	
	New Method	Old Method
negVideo1	160/ 4539 ( 3.5%)	260/ 4539 ( 5.7%)
negVideo2	0/ 155 ( 0.0%)	0/ 155 ( 0.0%)
negVideo3	0/ 160 ( 0.0%)	0/ 160 ( 0.0%)
negVideo4	140/ 1931 ( 7.3%)	0/ 1931 ( 0.0%)
negVideo5	10/ 439 ( 2.3%)	228/ 439 (51.9%)
negVideo6	0/ 541 ( 0.0%)	0/ 541 ( 0.0%)
negVideo7	0/ 3761 ( 0.0%)	0/ 2943 ( 0.0%)
negVideo8	5/ 645 ( 0.8%)	20/ 645 ( 3.1%)

## 5. CONCLUSIONS

In this paper, a real-time video fire detection system is developed based on the covariance texture representation method.

Covariance method is ideally suited for flame detection because flames exhibit random behaviour [4] and it is experimentally observed that the underlying random process can be considered as a wide-sense stationary process in a flame region in video. Therefore, second order statistical modeling using the covariance method provides a good solution to flame detection in video. An important contribution of this article is the use of temporal covariance information in the decision process. Most fire detection methods use color, spatial and temporal information separately, but in this work we use temporally extended covariance matrices representing all the information together. The method works well when the fire is clearly visible and in close range such that the flicker and irregular nature of flames are observable. On the other hand, if the fire is small and far away from the camera or covered by dense smoke the method might perform poorly.

## 6. REFERENCES

- [1] W. Phillips, M. Shah, and N. Da Vitoria Lobo, "Flame recognition in video," in *Applications of Computer Vision, 2000*, 2000.
- [2] Che-Bin Liu and Narendra Ahuja, "Vision based fire detection," in *ICPR '04*, 2004.
- [3] Thou-Ho Chen, Ping-Hsueh Wu, and Yung-Chuen Chiou, "An early fire-detection method based on image processing," in *ICIP '04.*, 24-27 2004, vol. 3, pp. 1707 – 1710 Vol. 3.
- [4] B. Uğur Töreyn, Yiğithan Dedeoğlu, Uğur Güdükbay, and A. Enis Çetin, "Computer vision based method for real-time fire and flame detection," *Pattern recognition letters*, vol. 27, no. 1, 2006.
- [5] B. Uğur Töreyn, Yiğithan Dedeoğlu, and A. Enis Çetin, "Flame detection in video using hidden markov models," in *ICIP 2005.*, 11-14 2005, vol. 2.
- [6] B. Uğur Töreyn and A. Enis Çetin, "Online detection of fire in video," in *CVPR '07.*, 17-22 2007.
- [7] W.B. Hong, J.W. Peng, and C.Y. Chen, "A New Image-Based Real-Time Flame Detection Method Using Color Analysis," in *IEEE International Conference on Networking, Sensing and Control*, 2005, pp. 100–105.
- [8] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," *Computer Vision–ECCV 2006*, pp. 589–600, 2006.
- [9] Chih-Chung Chang and Chih-Jen Lin, *LIBSVM: a library for support vector machines*, 2001, Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.