

# FLAME DETECTION FOR VIDEO-BASED EARLY FIRE WARNING SYSTEMS AND 3D VISUALIZATION OF FIRE PROPAGATION

Kosmas Dimitropoulos, Filareti Tsalakanidou, Nikos Grammalidis  
Informatics and Telematics Institute, ITI-CERTH,  
1<sup>st</sup> Km Thermi-Panorama Rd, Thessaloniki, Greece  
{dimitrop, filareti, ngramm}@iti.gr

## ABSTRACT

Early and accurate detection and localization of flame is an essential requirement of modern early fire warning systems. Video-based systems can be used for this purpose; however, flame detection remains a challenging issue due to the fact that many natural objects have similar characteristics with fire. In this paper, we present a new algorithm for video based flame detection, which employs various spatio-temporal features such as colour probability, contour irregularity, spatial energy, flickering and spatio-temporal energy. Various background subtraction algorithms are tested and comparative results in terms of computational efficiency and accuracy are presented. Experimental results with two classification methods show that the proposed methodology provides high fire detection rates with a reasonable false alarm ratio. Finally, a 3D visualization tool for the estimation of the fire propagation is outlined and simulation results are presented and discussed.

## KEY WORDS

Wildfires, Flame Detection, 3D Fire Propagation Visualization

## 1. Introduction

Video-based forest surveillance coupled with image processing techniques for flame and smoke detection is one of the most promising solutions for automatic forest fire detection due to its low cost and short response time. Besides early detection of wildfires, early intervention and efficient fire management is also very important in fire fighting. If fire-fighters had access to fire propagation estimation information, then fire management would be considerably more efficient. However, the main disadvantage of early warning systems based on video surveillance is increased false alarm rates due to atmospheric conditions (clouds, shadows, dust particles), light reflections etc [1].

Especially in the case of flame detection, the main challenge that researchers have to face is the chaotic and complex nature of the fire phenomenon and the large variations of flame appearance in video. In [2], Toreyin et al. proposed an algorithm in which flame and fire flicker are detected by analyzing the video in the wavelet domain, while in [3] a hidden Markov model was used to mimic the temporal behaviour of flame. Zhang et al [4] proposed a contour based forest fire detection algorithms using FFT and wavelets, whereas Celik and Demiral [5] presented a rule-based generic colour model for

fire-flame pixel classification. More recently, Ko et al [6] used hierarchical Bayesian networks for fire-flame detection and a fire-flame detection method using fuzzy finite automata [7].

Despite the extensive research results listed in literature, video based flame detection remains an open issue. This is due to the fact that many natural objects have similar colours as those of the fire (including the sun, various artificial lights or reflections of them on various surfaces) and can often be mistakenly detected as flames. In this paper, we present a new algorithm for video based flame detection, which employs a set of spatio-temporal features of fire such as colour probability, contour irregularity, spatial energy, flickering and spatio-temporal energy. More specifically, various background subtraction algorithms are investigated and comparative results in terms of computational efficiency and accuracy are presented. In order to accurately model the colour space of fire, a new colour analysis approach using non-parametric modelling is introduced, while the spatial energy of a candidate region is estimated by applying a 2D wavelet analysis only on the red channel of the image. In addition to the detection of the flickering effect, we introduce a new feature, the spatio-temporal energy, in order to further reduce the false alarm rate. The spatio-temporal energy concerns the variance of the spatial energy in a region of the image within a temporal window and aims to identify the irregular changes of the fire's shape. For the discrimination between fire and non-fire regions, two classification methods are investigated (a Support Support Vector Machine (SVM) classifier and a rule-based approach) and experimental results with both approaches are presented. Finally, a 3D visualization tool for the estimation of the fire propagation is described and simulation results with different sets of parameters are presented and discussed.

The rest of this paper is organized as follows: In Section 2, the proposed methodology is presented and the different processing steps are described in detail. In Section 3 experimental results with fire and non-fire video sequences are presented, while Section 4 describes the 3D fire propagation visualization too. Finally, conclusions are drawn in Section 5.

## **2. Methodology**

The proposed methodology initially applies background subtraction in order to detect moving objects and then colour analysis to identify candidate flame regions. Since candidate regions may correspond either to real fire or to fire-coloured objects, further processing is required for discarding false candidates. This processing aims to identify spatio-temporal characteristics, which distinguish fire from fire-coloured objects. As a result, a set of extracted features is generated for each candidate region and the final decision is made by a classifier as shown in Figure 1.

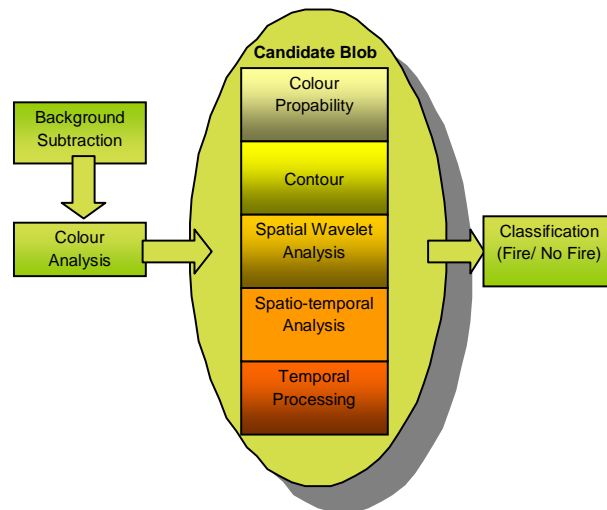


Figure 1. The proposed methodology

The different processing steps of the proposed algorithm are described in detail in the following sub-sections.

## 2.1 Background Subtraction

Thirteen background subtraction methods have been evaluated, both in terms of accuracy (using a test sequence with ground truth data) and speed (which is very important for fire detection applications). More specifically, the following algorithms have been evaluated: BGS 1: Adaptive Median [8], BGS 2: Gaussian Mixture Model [9], BGS 3: Improved Adaptive Gaussian Mixture Model [10], BGS 4: Running Average, BGS 5: Running Gaussian Average [11], BGS 6: Temporal median [12], BGS 7: Eigenbackground [13], BGS 8: Bayes Classification [14], BGS 9: Improved Mixture of Gaussians [15], BGS 10: Adaptive Thresholding [16], BGS 11: Kernel Density Estimation based Background Subtraction [17], BGS 12: Adaptive Background Extraction [18], BGS 13: Frame Differencing. To evaluate the performance of all these background subtraction algorithms, we used a video sequence from the VSSN06 algorithm competition [19]. The selected video sequence contains a moving background (trees, bushes etc) and it was recorded by a fixed camera. The ground truth of foreground objects in the scene is available thus facilitating the evaluation procedure.

In Figure 3 the corresponding execution times are shown, while in Figure 2, the sensitivity vs.1-specificity values of all algorithms are plotted, while. The default algorithm parameter values were used. Results show that the Adaptive Median (BGS1) algorithm outperforms the others in terms of computational time, while in terms of accuracy, the Adaptive Median (BGS1), Temporal median (BGS6) and Bayes classification (BGS8) are the most efficient methods. In our experimental results in Section 3 we have used the Adaptive Median (BGS1) method.

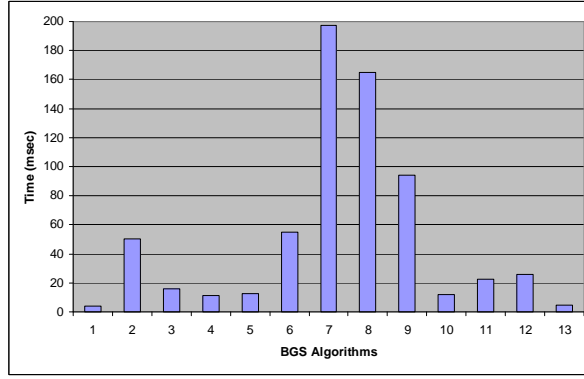


Figure 2. The execution times of the 13 background extraction algorithms using their default parameters.

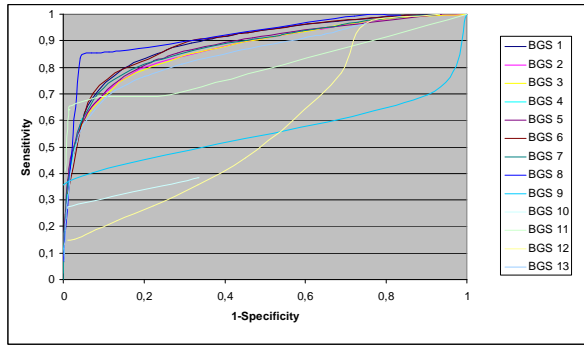


Figure 3. ROC curves of all 13 background extraction algorithms for the video sequence of [19].

## 2.2 Colour Analysis

The second processing step aims to filter out non fire- coloured moving pixels. Only the remaining pixels are considered for blob analysis, thus reducing the required computational time of the whole processing. To filter out non-fire moving pixels, we compare their values with a predefined RGB colour distribution created by a number of pixel-samples from video sequences containing real fires.

Let  $x_1, x_2, \dots, x_N$  be  $N$  fire-coloured samples of the predefined distribution. Using these samples, the probability density function of a pixel  $x_i$  can be non-parametrically estimated using the kernel  $K_h$  (Elgammal, 2000) as:

$$\Pr(x_i) = \frac{1}{N} \sum_{i=1}^N K_h(x_i - x_i)$$

If we choose our kernel estimator function,  $K_h$ , to be a Gaussian kernel,  $K_h = N(0, S)$ , where  $S$  represents the kernel function bandwidth, and we assume a diagonal correlation matrix  $S$  with a different kernel bandwidths  $\sigma_j$  for the  $j^{th}$  colour channel, then the density can be estimated as:

$$\Pr(x_i) = \frac{1}{N} \sum_{i=1}^N \prod_{j=1}^3 \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{1}{2} \frac{(x_{ij} - x_{ij})^2}{\sigma_j^2}}$$

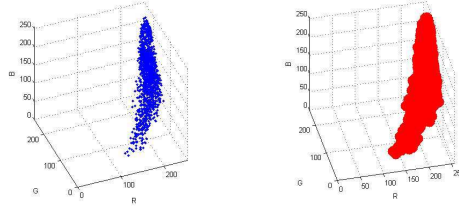


Figure 4. (a) RGB colour distribution and (b) the colour distribution with a global threshold around each sample.

Using this probability estimation, the pixel is considered as a fire-coloured pixel if  $Pr(x_i) < th$ , where the threshold  $th$  is a global threshold for all samples of the predefined distribution and can be adjusted to achieve a desired percentage of false positives. Hence, if the pixel has a RGB value which belongs to the distribution of Figure 4(b), then it is considered as a fire-coloured pixel as shown in Figure 5.

After the blob analysis step, the colour probability of each candidate blob is estimated by summing the colour probability of each pixel in this blob.

$$Pr_{blob} = \frac{1}{N} \sum_{i=1}^N Pr(x_i)$$

where  $N$  is the number of the pixels in this blob and  $Pr(x_i)$  is the probability estimation of each pixel.

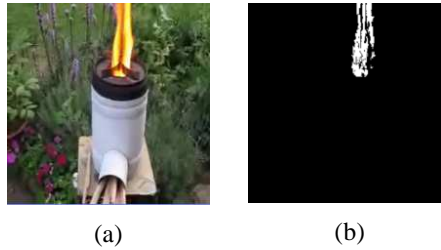


Figure 5. (a) Initial image (b) colour mask.

### 2.3 Contour analysis

The shape of flame objects is often irregular thus high irregularity/variability of the blob contours can also be considered as a flame feature. This irregularity is identified by tracing the object contour, starting from any pixel on it. Thus a direction arrow is defined for each pixel on the contour, which can be specified by a label  $L$ , where  $0 \leq L < 8$ , assuming 8-connected pixels, as shown in Figure 6.

The variability of the contour for each pixel can be measured by calculating the difference (distance) between two consecutive directions (from and to the specific pixel), using the following formula, which returns a distance between 0 and 4 for each pixel:

$$d = \begin{cases} d_{\max} - d_{\min} & \text{if } d_{\max} - d_{\min} \leq 4 \\ 8 + d_{\min} - d_{\max} & \text{otherwise} \end{cases}$$

where  $d_{\min} = \min(d_1, d_2)$  and  $d_{\max} = \max(d_1, d_2)$ .

The average value of this distance function can be used as a measure of the irregularity of the contour.



Figure 6. (a) The direction to the next boundary pixel is represented by a code (0 – 7) (b) An example of a boundary and the direction of the arrows.

## 2.4 Spatial Wavelet Analysis

Since there is higher spatial variation in regions containing actual fire compared to fire-coloured objects, the next step aims to detect the spatial variation in a moving fire-coloured blob. To this end, a two-dimensional wavelet is applied on the red channel of the image as shown in Figure 7.

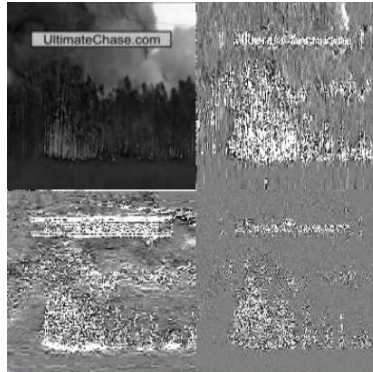


Figure 7. Two dimensional spatial wavelet analysis (The original image was downloaded from [20])

For the single stage wavelet transform, the weights of the low pass and high pass filters are  $[0.25 \ 0.5 \ 0.25]$  and  $[-0.25 \ 0.5 \ -0.25]$  respectively. The final energy mask (Figure 8), which contains the spatial variation of the image, is obtained by adding low-high, high-low and high-high wavelet sub-images according to the following equation.

$$E(i, j) = \frac{1}{N} (HL(x, y)^2 + LH(x, y)^2 + HH(x, y)^2)$$

where  $N$  is the total number of pixels.

For each blob, the spatial wavelet energy is estimated by summing the individual energy of each pixel belonging to the blob.

$$E_{blob} = \frac{1}{N_b} \sum_{i,j} E(i, j)$$

where  $N_b$  is the number of pixels in a blob.



Figure 8. (a) The energy image corresponding to wavelet subimages of Figure 6 and (b) the initial image along with the detected blobs.

## 2.5 Spatio-temporal Analysis

The shape of fire changes irregularly due to the airflow caused by wind or due to the type of burning material. The spatio-temporal analysis step aims to identify these changes in order to discriminate between fire regions and fire-coloured objects. For this reason, a new feature is extracted considering the variation of the spatial energy in a blob within a temporal window of  $N$  frames. The variance of a pixel's spatial energy is estimated as follows:

$$V(i, j) = \frac{1}{N} \sum_{t=1}^N (E_t(i, j) - \bar{E}(i, j))^2$$

where  $N$  is the size of the temporal window,  $E_t$  is the spatial energy of the pixel in time instance  $t$  and  $\bar{E}$  is the average value of its spatial energy. The final spatio-temporal energy mask is shown in Figure 9.



Figure 9. Flame creates high spatio-temporal energy values.

For each blob, the total spatio-temporal energy,  $S_{blob}$ , is estimated by summing the individual energy of its pixels:

$$S_{blob} = \frac{1}{N_b} \sum_{i,j} V(i, j)$$

where  $N_b$  is the number of pixels in a blob.

## 2.6 Temporal Processing

Flickering is a very important characteristic of flame and it is very significant for discriminating between “flame” and “non-flame” regions. In our approach, we use a temporal window of  $N$  frames ( $N$  equals 50 in our experiments), yielding a 1-D temporal sequence of  $N$  binary values for each pixel position. Each binary value is set to 0 if the pixel was labeled as “no flame candidate” or 1 if the pixel was labeled as “flame candidate” after the background extraction and colour analysis processing steps. To quantify the effect of flickering, we traverse this temporal sequence for each “flame candidate” pixel and measure the number of transitions between “flame candidate” and “no flame candidate” ( $0 \rightarrow 1$ ). This number can be directly used as a flame flickering feature, with flame regions characterized by a sufficiently large value of flame flickering.

## 2.7 Classification

For the classification of the 5-dimensional feature vectors (colour probability, irregularity of the contour, spatial wavelet energy, spatio-temporal energy, flame flickering) of each blob, we employed a Support Vector Machine (SVM) classifier with RBF kernels. In our experiments, the training of the SVM classifier was based on approximately 500 frames of fire and non-fire video sequences.

In addition to SVM, a second classification approach, which is based on a number of thresholds and rules, was also adopted. More specifically, a threshold  $th_i$  is empirically defined for each feature  $i$  after a number of experiments (Colour probability:  $th_1 = 0.002$ , Contour:  $th_2 = 0.8$ , Spatial wavelet energy:  $th_3 = 100$ , Temporal energy:  $th_4 = 20$ , Spatio-temporal variance:  $th_5 = 30$ ). Then the following classification technique is applied: first, the value of metric  $C$  for each feature vector  $f_i$  is computed by the following equation:

$$C = \sum_{i=1}^5 F(th_i, f_i)$$

where  $F$  is a function defined as follows:

$$F(th, f) = \begin{cases} 0 & , f < th \\ 1 & , f > th \end{cases}$$

Then, the following rule is applied for each feature vector: If  $C > M$  ( $M \in [1, 5]$ ), then the feature vector is classified as a fire, otherwise it is considered as a false alarm i.e. non-fire (in our experiments  $M=3$ ).

## 3. Experimental Results

To evaluate the performance of the proposed method, videos containing fire or fire-coloured objects were used. Figure 10 shows the detection of flame along with the intermediate feature masks (background, colour, spatial wavelet, spatiotemporal and temporal map), while in Figure 11, an example with a video containing a moving fire-coloured object is presented. As it is clear from the intermediate masks, extracted feature values are higher in case of flame detection due to the random behaviour of fire.



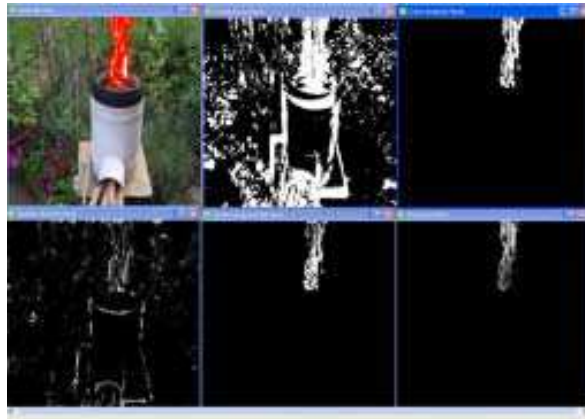


Figure 10. True detection of fire



Figure 11. True rejection of a fire-coloured object (orange trunk)

Fourteen test videos were used for the evaluation of the algorithm. The first seven videos contain actual fires, while the rest contain fire coloured moving objects e.g. car lights, sun reflections etc. Screenshots from these videos are presented in Table 1 (the first column presents fire detection results, while the second column contains screenshots from videos with fire coloured moving objects).

Results are summarized in Figure 12 and Figure 13 in terms of the true positive and true negative ratios. The definition of these terms is given below:

- True Positive: the number of frames in which fire is correctly detected out of the total number of frames in a fire test video.
- True Negative: the number of frames in which no fire was detected out of the total number of frames in a non-fire test video.

Experimental results show that the proposed method provides high detection rates in all videos containing fire, with a reasonable false alarm ratio in videos without fire. In most cases, the SVM classification provides higher true positive rates in videos containing fire, while rule-based classification outperforms in non-fire videos. The lower true negative rates, especially with SVM classification, are shown in “Non\_fire\_video3” due to the continuous reflections of car lights on the road. However, we believe that the results may be improved in the future with a better training of the SVM classifier. The

speed of the proposed method was an average of 13.2 fps, while the size of the video sequences of Table 1 was 320x240. The experiments were performed with a PC that has a Core 2 Duo 2.66 GHz processor.

Table 1

Test videos used for the evaluation of the proposed algorithm.



Fire\_video1



Non\_fire\_video1



Fire\_video2



Non\_fire\_video2



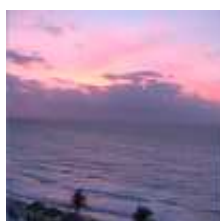
Fire\_video3



Non\_fire\_video3



Fire\_video4



Non\_fire\_video4



Fire\_video5



Non\_fire\_video5



Fire\_video6



Non\_fire\_video6



Fire\_video7



Non\_fire\_video7

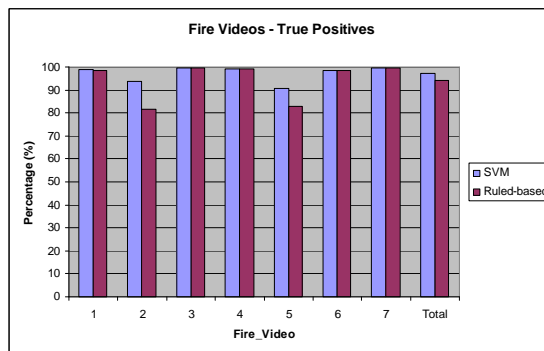


Figure 12. Experimental results with videos containing real fires

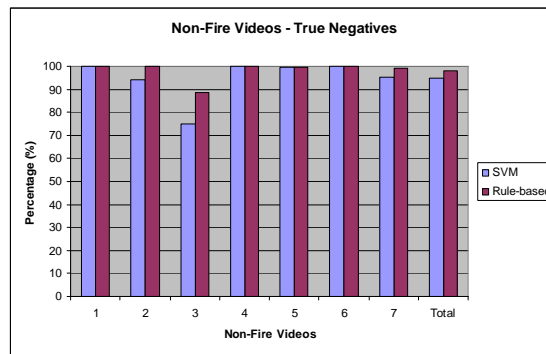


Figure 13. Experimental results with videos containing fire coloured objects

#### 4. 3D Visualization of Fire Propagation

The proposed flame detection algorithm can be used for the detection and localization of fire by a video-based early warning system. However, the detection of the starting point of fire is just the first step in fire fighting. The next step is efficient fire management. To this end the estimation and visualization of fire propagation is extremely significant since it will enable the fire fighting forces to cope with the fire and manage their resources effectively. Most of the fire propagation simulation software presented in the literature yields a 2D view (mostly a top view) of the fire area, which, however, may not provide a clear view of the situation to the persons responsible for the deployment of fire-fighting forces. In this paper, we present a user-friendly GIS simulator (Figure 14), which provides 2-D/3-D visualization of fire propagation estimation output (ignition time and flame length) and enables interactive selection of some parameters (e.g. ignition point, humidity parameters, wind direction etc). The system is based on Google Earth API [21], which is publicly available and allows the creation of impressive 3-D animations of the fire propagation, in addition to the static views.

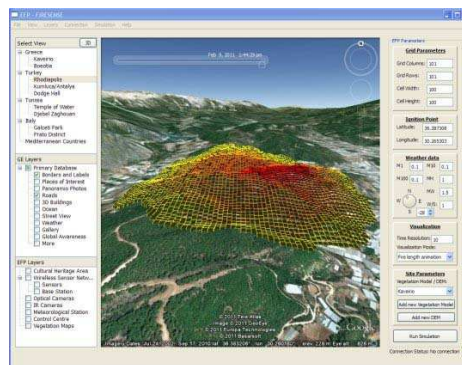
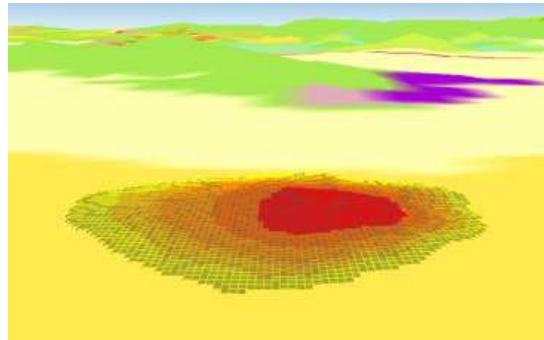


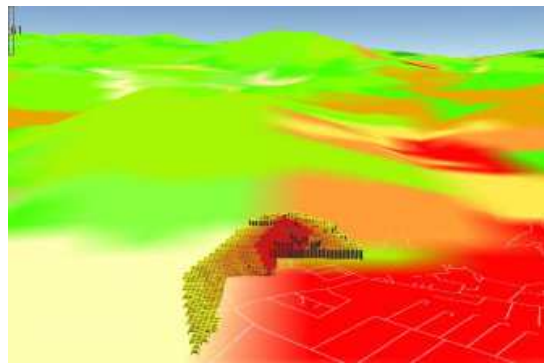
Figure 14. 3D Visualization of Fire Propagation

In this work, fire spread calculations are mainly based on Fire Behavior SDK [22] that implements the popular BEHAVE algorithm [23], instead of FireLib library [24], which was used in a previous work [25]. Fire Behaviour SDK has significant advantages over FireLib such as: a) built-in support for all Scott-Burgan fuel models [26] (40 models, including 17 dynamic models, where live/dead fuel load changes dynamically depending on moisture conditions), b) support of mixture of two fuel

models at the same cell, c) modelling of crown fires, in addition to surface fires, and d) provision of additional information (e.g. fire ignition probability).



(a)



(b)

Figure 15 Simulation results with (a) weak wind and uniform vegetation cover, (b) stronger wind and non-uniform vegetation cover

According to BEHAVE, fire propagation depends on a number of parameters such as. ignition point, fuel model, humidity, wind, terrain data and other factors. For the terrain data, we used Digital Terrain Models (DTMs) by SRTM, which are freely available with a resolution of 90m. Wind information (direction and speed) can be obtained in real time from existing weather stations via Internet weather portals, while vegetation information is based on CORINE land cover maps [27], which support 44 land cover and land use classes representing the major surface types across Europe.

Figure 15 shows two simulation results with different sets of parameters. More specifically, Figure 15(a) shows a simulation result from a valley area in Thebes, Greece, with uniform vegetation cover (the vegetation layer is activated) and low wind speed values (1m/s). On the other hand, in the second simulation in Figure 15(b) (Prato, Italy), the wind speed has been increased (4m/s) and for this reason the final result is more directional. It is also worth mentioning that the propagation of the fire is halted in the edges of the red (urban) area, where the simulation considers that there is lack of vegetation (fuel).

## 5. Conclusion

Early detection of fire is a crucial issue for the suppression of wildfires and the mitigation of disaster. Video based systems for automatic early forest fire detection is a promising technology, which can provide real-time fire detection information with high accuracy. In this paper, we presented a flame detection methodology, which identifies spatio-temporal features of fire such as colour probability, contour irregularity, spatial energy, flickering and spatio-temporal energy. Experimental results with a number of test videos have already shown the great potential of the proposed method.

In the future, we plan to investigate the use of blocks instead of blobs in order to increase the computational efficiency of the flame detection algorithm and to compare our simulation results with real data from past fires.

## Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7-ENV-2009-1) under grant agreement no FP7-ENV-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". The authors would like to thank all project partners for their fruitful cooperation within FIRESENSE project.

## References

- [1] Stipaničev, D., Vuko, T., Krstinić, D., Štula, M., Bodrožić, Lj., Forest Fire Protection by Advanced Video Detection System - Croatian Experiences, Third TIEMS Workshop – Improvement of Disaster Management System, Trogir, 26-27 September, 2006
- [2] Töreyn, B., Dedeoglu, Y., Gudukbay, U., Cetin, A.E., "Flame detection in video using hidden markov models," IEEE Int. Conf. on Image Processing, pp. 1230-1233, 2005.
- [3] Töreyn, B., Dedeoglu, Y., Gudukbay, U. Cetin, A.E., Computer vision based method for real-time fire and flame detection, Pattern Recognition Letters, 27, pp. 49–58, 2006.
- [4] Zhang, Z., Zhao, J., Zhang, D., Qu, C., Ke, Y., and Cai, B. "Contour Based Forest Fire Detection Using FFT and Wavelet", In Proceedings of CSSE (1). 2008, 760-763.
- [5] Celik, T., Demirel, H., "Fire detection in video sequences using a generic colour model," Fire Safety Journal, Vol. 44, pp. 147-158, 2009.
- [6] Ko, B. C., Ham, S. J., Nam, J. Y., Modeling and Formalization of Fuzzy Finite Automata for Detection of Irregular Fire Flames, IEEE Transactions on Circuits and Systems for Video Technology, May 2011.
- [7] Ko, B., Cheong, K., Nam, J., Fire detection based on vision sensor and support vector machines, Fire Safety Journal, Vol. 44, pp. 322-329, 2009.
- [8] McFarlane, N. and Schofield, C., Segmentation and tracking of piglets in images, British Machine Vision and Applications, Vol 8, pp. 187-193, 1995.
- [9] Stauffer, C., Grimson, W.E.L., Adaptive background mixture models for real-time tracking, IEEE Conference on Computer Vision and Pattern Recognition, pp. 246-252, 1999
- [10] Zivkovic, Z., Improved adaptive Gaussian mixture model for background subtraction, International Conference of Pattern Recognition, Vol.2, pp. 28-31, 2004

- [11] Wren, C., Azarhayejani, A., Darrell, T., Pentland, A.P., Pfunder: real-time tracking of the human body, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 19, No.7, pp.780–785, 1997.
- [12] Cucchiara, R., Grana, C., Piccardi, M. and Prati, A., Detecting moving objects, shots, and shadows in video stream, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 10, pp. 1337—1342, 2003.
- [13] Oliver, N., Rosario, B., Pentland, A.P., A Bayesian computer vision system for modeling human interactions, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 8, pp. 831 – 843, 2000.
- [14] Li, L., Huang, W., Gu, I.Y.H., Tian, Q., Foreground object detection from videos containing complex background, ACM International Conference on Multimedia, pp.2-10, New York, USA, 2003
- [15] KadewTraKuPong, P., Bowden, R., An improved adaptive background mixture model for real-time tracking with shadow detection, 2nd European Workshop on Advanced Video-Based Surveillance Systems, Kingston, UK, 2001.
- [16] Lluís, J., Miralles, X., Bastidas, O., Reliable real-time foreground detection for video surveillance applications, 3rd ACM International Workshop on Video Surveillance and Sensor Networks, Singapore, pp. 59-62, 2005.
- [17] Elgammal, A. Harwood, D., Davis L. “Non-parametric model for background subtraction” 6th European Conference on Computer Vision, Vol. 1843, pp. 751-767, Dublin, Ireland, 2000.
- [18] Collins, R.T., Lipton, A.J., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., Hasegawa, O., Burt, P., Wixson, L., A system for video surveillance and monitoring, Technical Report CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University, May, 2000.
- [19] VSSN, [http://mmc36.informatik.uni-ugsburg.de/VSSN06\\_OSAC/](http://mmc36.informatik.uni-ugsburg.de/VSSN06_OSAC/)
- [20] Ultimate Chase, <http://www.ultimatechase.com/>
- [21] Google Earth API <http://code.google.com/intl/el-GR/apis/earth/>
- [22] Fire Behaviour SDK, [http://www.fire.org/index.php?option=com\\_content&task=category&sectionid=2&id=31&Itemid=295](http://www.fire.org/index.php?option=com_content&task=category&sectionid=2&id=31&Itemid=295)
- [23] P. L. Andrews, “BEHAVE: fire behavior prediction and fuel modeling system-BURN subsystem, Part 1”, USDA Forest Service Gen. Tech. Rep. INT-194.
- [24] Firelib, 2010. “Firelib software implementation and documentation”, <http://www.fire.org/index.php?option=content&task=category&sectionid=2&id=11&Itemid=29>, (accessed 20 May 2010)
- [25] K. Dimitropoulos, K. Köse, N. Grammalidis, A. E. Cetin, "Fire Detection and 3-D Fire Propagation Estimation for the Protection of Cultural Heritage Areas", ISPRS Technical Commission VIII Symposium, 9-12 August 2010, Kyoto, Japan.
- [26] Scott, Joe H.; Burgan, Robert E. 2005. Standard fire behavior fuel models: a comprehensive set for use with Rothermel's surface fire spread model. Gen. Tech. Rep. RMRS-GTR-153. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station.
- [27] Corine Land Cover, <http://www.eea.europa.eu/publications/COR0-landcover>