

A real-time method for smoke detection in monitored forest areas

Vladimír Sedlák, Viera Stopjaková and Juraj Brenkuš

Institute of Electronics and Photonics
Slovak University of Technology in Bratislava
Bratislava, Slovakia
vladimir.sedlak@stuba.sk

Abstract – An improved real-time method for smoke detection based on motion and color analysis is presented. Smoke detection in video surveillance is very important for early fire detection. In general, it is assumed that smoke is a low frequency signal, which may smooth the background. However, these conditions can be met by some pure-color objects, and smoke also produces high frequency signal on the edges. An improved smoke detection method was proposed in order to prevent these problems. The presented technique uses well known background subtraction for identifying of moving regions as well as novel color based algorithm for smoke detection. Detection efficiency is improving using a classifier that provides the identification of smoke-like regions. The main advantage is that it is not required to process each frame of the video signal for sufficient accuracy. Sampling rate was set to a few second in order to find the optimum value between performance and system requirements. Performance of the proposed method was evaluated using different kinds of the input signal, where satisfactory results were achieved.

Keywords-smoke detection, image processing, background subtraction, motion features.

I. INTRODUCTION AND MOTIVATION

Early fire detection is a critical task for fire alarm systems, which are designed for monitoring of fire in buildings or in outdoor environment. The most of current systems is based on infrared sensors, optical sensors, or ion sensors that detect the main characteristics of fire, such as heat, smoke or radiation. Disadvantage of the traditional approach is that it cannot provide any other useful information towards potential warnings such as localization and size of fire or the degree of burning. Another drawback is that these systems are not alerted until the particles reach the sensors.

Compared to conventional fire detection methods the video-based fire detectors have many advantages, such as fast response, long distance of detection, large protection area and others. Early-warning and immediate response is the important preventing factors to avoid losses affecting environment as well as human casualties. On the other hand, the disadvantage of video-based detection methods is that they generate relatively high number of false alarms. This fact is motivation for many researchers all over the world to create more reliable and effective detection technique.

II. BACKGROUND

Smoke is very often used for video-based detection of fire because it might be observed much sooner than flames. That is the reason why the extraction of smoke features is more complicated. The basic requirements to smoke detection in video sequences are low probability of false operations on moving objects, movement of clouds, present fog, and processing of video sequences to a real time. Smoke detection algorithms very often use color and motion information to detect smoke in digital video sequences. The disadvantage of this approach is that visual characteristics of smoke like intensity and color are less visible than characteristics of flames.

Many different approaches and methods have been developed for reliable detection of smoke. In the first step, the segmentation of the input video sequence into region of interests (which can contain some amount of smoke) is commonly performed. Currently, there are several approaches to segment the moving smoke regions, e.g. background subtraction [1], Gaussian mixture model [2, 3] and MSER (Maximum Stable External Regions) approach [4]. Next step is usually based on the fact that smoke regions exhibit some features that distinguish from other regions in shape, variance, and the value of gray color. In some cases, there are also dynamic features included, such as the directions of movements, the decrease of coefficients of wavelet transform.

Survey on different smoke detection techniques is presented in [5], where the authors claim that it is very crucial to develop an appropriate detection system to avoid dangerous situation caused by fire. This work also introduced the Wavelet-based smoke detection method [6] as a useful approach for outdoor sequences. There is also mentioned the covariance method, method based on supported vector machine and method based on color. Another survey work [7] shows the evolution of smoke detection methods and presents a lot of different approaches, where it was concluded that the biggest problem is still a high risk of false alarms. Authors expect that efficient dynamic texture segmentation might help reduce this risk.

A lot of detection methods is based on analysis of color and motion [1, 8]. Currently, there are presented some extensions of this approach, which try to improve the standard procedure. One of them is presented in [9], where authors use synthetic smoke plume generation for the performance enhancement. The advantage is that the system can be tuned using real samples containing synthetic smoke that reduces

generation of false alarms. Many methods use different classifiers (support vector machine, neural networks or hidden Mark models), which are trained for determination of smoke-like objects in the input signal. These classifiers often use features as color, direction of movement, size of area, etc. Recently, a popular approach for smoke detection is based on optical flow [10], which is used for extraction of moving objects.

Another approach is presented in [11], where the proposed solution for fire and smoke detection is integrated in an FPGA circuit. The advantage is that FPGA can meet the needs for high-speed image processing (with its hardware features) and can make the design flexible. This solution is proposed as a modular system, where the core consists of a detection module based on 2-dimensional discrete wavelet transform. The aim is to estimate the movement in the input image and growth of smoke areas.

The aim of this paper is to propose an improved real-time video-based method for smoke detection. The paper is organized as follows. Section III introduces the proposed algorithm and describes each step of the whole process. Achieved results are presents in Section IV, and the paper is concluded in the last section.

III. PROPOSED DETECTION METHOD

The proposed approach for detection of smoke is based on algorithm designed for low sampling rate (0.5 – 5 Hz) of the input image. This enables to process signal from multiple cameras using standard computing hardware. Therefore, the proposed method requires a less expensive hardware setup in comparison to standard smoke detection techniques. It can use low-cost cameras operating under visible light conditions. The detection process, shown in Fig. 1, is divided into three main blocks (tasks): signal pre-processing, feature selection and feature classification. This is a classic concept based on the color and motion analysis, and the initial part is inspired by method presented in [12].

A. Pre-processing

The first step of pre-processing is to acquire an image from one of the cameras. The image is then divided into squares of the equal height and width. This brings a reduction of computing time because the count of regions is $M \times M$ times lower than the number of pixels (symbol M represents reduction index). The drawback is a loss of the resolution. In the next step, the characteristic feature for each region using average value or median is identified. In article [12], it has been reported that median is a better option than the average value. The proposed algorithm uses average value because of a simpler calculation, and additionally, our experiments do not confirm positive impact of median.



Figure 1. Block diagram of smoke detection system.

Afterwards, simple moving average (SMA) and moving standard deviation (MSD) are updated. Those are used for motion analysis. The algorithm uses very effective and fast computation method described by (1), where symbol A represents the actual average value of RGB pixel, $LastA$ is the last average value in a buffer, $LastIx$ is the last index of the buffer and $Window$ is the length of a window.

$$SMA = \frac{SMA + (LastIx - 1) + A}{LastIx} \rightarrow \text{buffering}$$

$$SMA = \frac{SMA + (A - LastA)}{Window} \rightarrow \text{computing} \quad (1)$$

$$MSD = \sqrt{\left(AVG(A^2) - (AVG(A))^2 \right)}$$

B. Feature extraction

The feature extraction part is performed in a few steps and as shown in Fig. 2. The motion detection is inspired by [12] and is performed in the first step after signal pre-processing. It is necessary to wait for a specific time at the beginning until the sampling buffer is ready. Next, the analysis of color is done and the result is masked by the motion mask. Then, the regions based on their area and direction of motion are analyzed. An important step is filtering out noise and regions, which are active for a very short time in order to improve the effectivity of detection. Finally, the classification of active regions based on their features is carried out.

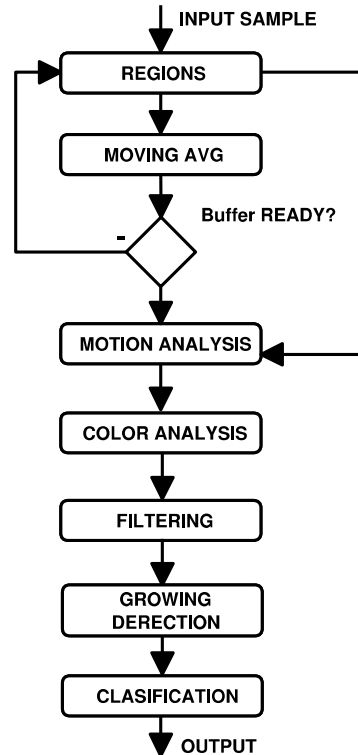


Figure 2. Proposed algorithm flow

1) Moving region detection

The detection of moving regions includes the identification of the candidate smoke regions by estimating the differences between a sample and the average value of the background image. The basic idea behind this is that if we subtract an image of the same landscape from the average (reference) value we can see what was changed. If there is any sector with the subtracted value higher than a threshold, a motion is detected. The question is how to specify the interval of subtraction and the threshold value. In [12], it was reported that the interval should be short enough to catch fast changes and long enough to catch slow changes. Fast changes and slow changes represent moving objects in a short distance and in a long distance, respectively. The authors used three different intervals (short, medium and long) to cover the whole spectrum of cases. We decided to make a compromise and therefore, we used only medium option (30 seconds). The motivation for this step was reduction of computation time. Another setting that has to be determined is the threshold value. We decided to use a multiplication of moving standard deviation what creates dynamic borders. This parameter was computed in pre-processing stage and the window length is the same like that used for calculation of moving average. Moving region detection can be described by (2), where $I(x, y, t)$ is the intensity of the pixel (x, y) at time t , symbol N represents the threshold multiplier and MM is acronym for the motion mask.

$$MM = \begin{cases} 1, & \text{if } |I - SMA| > N \times MSD \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

2) Color analysis

Many different approaches for detection smoke-like colors have been published. A group of them is based on the analysis of RGB color space, and uses a comparison of color channels against the threshold. There are also methods that use similar threshold-based approach in YUV color space for identifying smoke colored regions. It is understood that the luminance value of smoke regions should be large, while the chrominance value should be very small. Other approaches using Fourier transform, statistical models, histograms or Wavelet transform are more sophisticated but more computationally demanding. Those are reasons why the threshold-based approach was used in our research.

We analyzed many different methods for determination of the threshold. The results were quite good but still not fully acceptable due to not being able to detect the smoke areas with low gradient. This low gradient case is shown in Fig.3 for a chosen pixel of smoke. Based on it, we have defined a custom threshold function that uses average values of RGB color channels and limitation of min and max values. The custom threshold function is depicted in Fig.4, where x-axis shows the average value of color channels and y-axis represents the threshold value in percentage. There are two additional rules that have to be met to mark the pixel as smoke colored pixel. Mathematically, all the rules are described by (3), and three different color masks are produced.

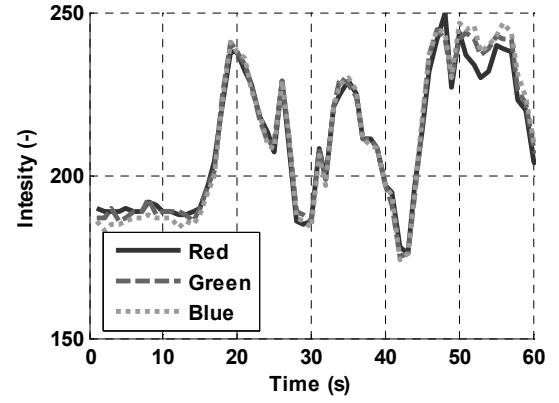


Figure 3. RGB color channels of specific smoke colored pixel.

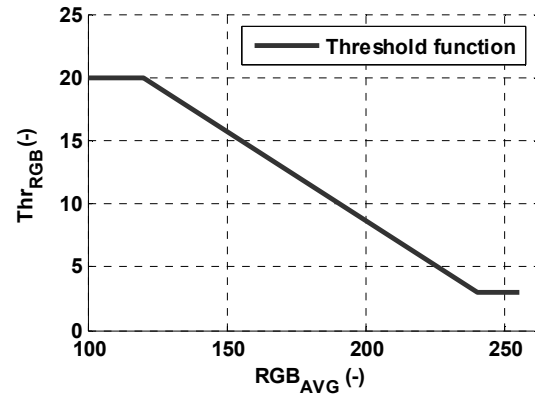


Figure 4. Custom threshold function.

3) Filtering & Growing detection

All presented masks (motion and color) are merged into the result mask (RM) in the last step. This process is mathematically described by (4). It is necessary to filter out noise and a short term movement. This operation is done by defining the regions of interest (ROI), which have a known area and central position. The position of each ROI is compared to the previous RM to find a relative movement of objects and determination of area change. The rule of the minimum area is also applied for elimination of computing error.

$$CM_{R1} = \begin{cases} 1, & \text{if } RGB > 150 \\ 0, & \text{otherwise} \end{cases}$$

$$CM_{R2} = \begin{cases} 1, & \text{if } |R - B| \& |R - G| \& |B - G| > 10 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

$$CM_{R3} = \begin{cases} 1, & \text{if } RGB > AVG \times (1 + T_{AVG}(AVG)) \\ 0, & \text{otherwise} \end{cases}$$

$$RM = MM \& CM_{R1} \& CM_{R2} \& CM_{R3} \quad (4)$$

C. Feature classification

The previous procedure returns a list of ROI with known features as duration of activity, position of starting point, relative movement or direction and area value. Based on these features, the classification procedure has to be able to identify the behavior (of changes) similar to smoke. These features are stored in a queue in the order they were generated. This gives chronological information about the process. The queue has a limited size (16 samples), and old samples are sequentially replaced by new ones.

The essential assumption for the correct detection of smoke based on observed features is that the starting point must be below horizon, the starting position of smoke plume is connected to the this point, the actual position of smoke plume is connected to the previous position and so on. Based on these rules, the classification procedure makes decision if the ROI represents smoke or not. The final step is the generation of warning, delayed for a few cycles in order to avoid a false alarm.

D. Hardware implementation

Our research was focused on development of an effective algorithm, which could be used for a multi-input (or multi-camera) solution. There are three different approaches that could be taken account. The first one employs high performance hardware, able to perform many operations at time. The disadvantage of this option is higher cost in comparison to “standard” hardware. The second approach is playing with the complexity of detection algorithms. Motivation is to use less complex algorithms, which consume less computing resources. The problem is that the complexity usually goes hand in hand with the effectivity. It means that less complex algorithm exhibits lower effectivity than the algorithm of higher complexity (this assertion is not a general rule). Development of an algorithm, which can be marked as less complex algorithm with the sufficient quality is very time consuming, since many requirements (e.g. optimal use of resources, code optimization, parallelism, multi-threading, familiarity with the smoke detection, etc.) have to be taken into account. The last option is focused on the sampling time - how often is the detection algorithm invoked. Low sampling time enables to process multiple input signals using more complex algorithm, what can improve the detection efficiency. High sampling time can smooth changes in the input image but reduces the detection accuracy.

In our case, the main requirements for the system design included the possibility of processing up to 24 input signals and low-cost hardware. Thus, we have decided to use combination of the second and the third options presented above. It means that the proposed system will use low sampling frequency and the detection algorithm with fast response (lower complexity). These conditions together with system requirements define a critical rule for sampling time, described by (5), where the symbol T_S represents sampling time, T_P is processing time and N_{MAX} is the maximum count of the input signals. The proposed implementation of the system is shown in Fig. 5.

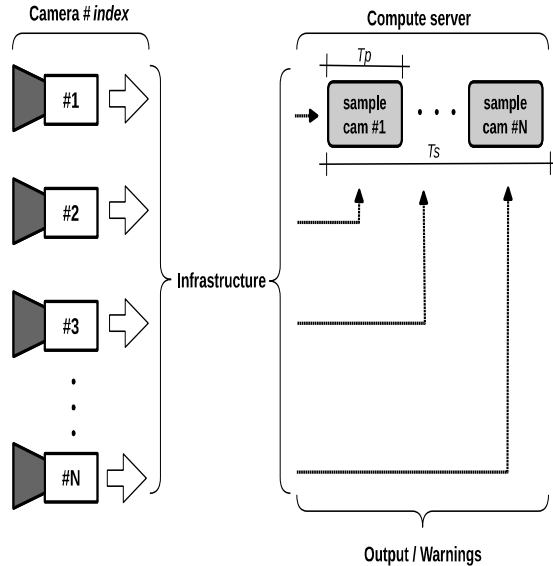


Figure 5. Implementaion scheme of the system.

The input samples are ordered in a queue and processed in a row. Inside each processing block (dark color) are all operations performed in parallel (using multithreading) for optimal use of resources and code should be optimized for speed. An alternative approach could have multiple queues which run in parallel. It is important to mention that this paper is focused on design of the detection algorithm suitable for multi input approach. We are going to present the system implementation in future work.

$$T_S \geq N_{MAX} \times T_P \quad (5)$$

IV. EXPERIMENTS

This section describes the verification procedure based on the verification data presented in [13]. This database contains a large set of fire videos acquired both in real environments and from the web. It is very useful for testing both smoke and fire detection algorithms because there are samples containing cloud, smoke, no smoke, reflections, and also samples affected by sunlight.

The proposed algorithm was implemented in the Visual Studio C++ for development and test purposes. It uses OpenCV library for simpler and faster image processing and reduction of development time. The dimensions of the input samples were 292x240, so they were processed directly without further size reduction. All experiments have been done on a personal computer (Intel® Core™ i7-4700MQ, 2.4GHz, 8GHz). The detection example is depicted in Fig. 6, where image (a) shows detected smoke that is bordered by orange circle. Image (b) shows moving average of the input video sample, while (c) shows the moving standard deviation. The motion mask MM computed using (2) is shown in (d), and the color mask generated by three different rules is shown in (e).

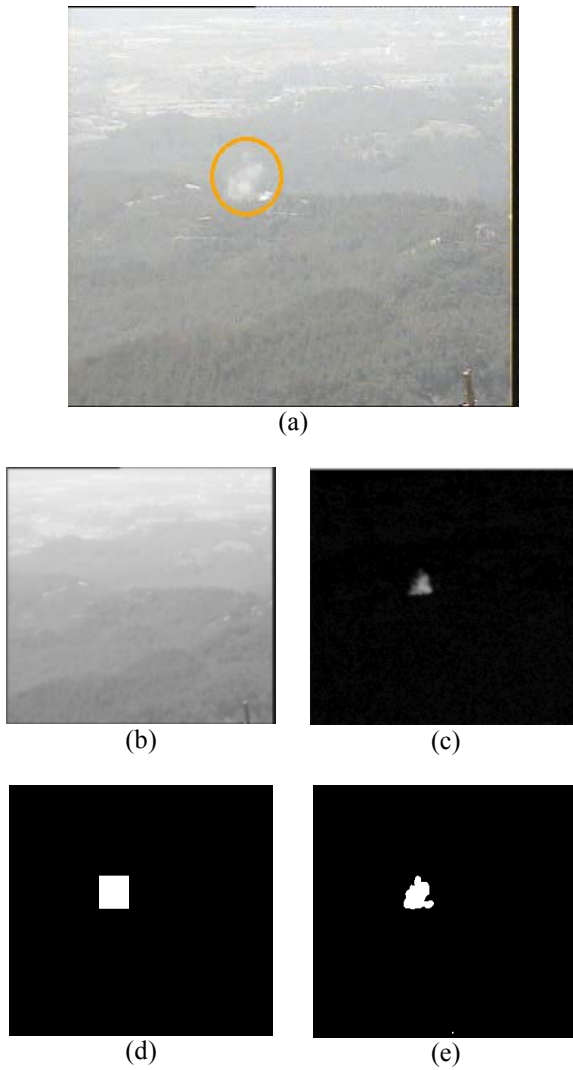


Figure 6. Achieved results and generated masks; (a) detected smoke bordered by orange cycle; (b) moving average of the input sample; (c) moving standard deviation of the input sample; (d) generated motion mask (MM); (e) generated color mask (CM).

The achieved results are summarized in Table I. The first and the fifth samples represent an ideal case, when there are no clouds, fog or smoke-like objects. In the second sample, there are some clouds but the algorithm is able to detect the smoke. The case represented by the third sample is not detected, which is an error because the smoke is hidden by fog. In the fourth case, the input sample is affected by sunlight but the algorithm is able to detect smoke correctly. The sixth sample represents smoke and fire what was correctly classified as warning. The last two samples represent a case when there is no smoke but instead of it there are smoke-like objects as fog and cloud. Correct response on these inputs is no warning that was met.

TABLE I. PERFORMANCE OF THE ALGORITHM UNDER SPECIFIC TEST CONDITIONS

Video index	Description	Smoke Alarm (Y/N)	State (OK / FAIL)
1	Smoke	Y	OK
2	Smoke + Clouds	Y	OK
3	Smoke + Fog	N	FAIL
4	Smoke + Sunlight	Y	OK
5	Smoke	Y	OK
6	Smoke + Fire	Y	OK
7	Fog only	N	OK
8	Clouds only	N	OK
9	Smoke + Fire	Y	OK
10	Clouds + Fog	N	OK

V. CONCLUSION

A real-time method for smoke detection in the monitored forest area was presented. The algorithm uses the motion detection based on moving averaging, color analysis and feature classification. Although this approach is well known, the article presents some innovative techniques such as the color analysis using the custom threshold function or filtering and merging of ROI. The algorithm was tuned for low sampling frequency because its implementation should process multiple-input signals at time. This fact is the main difference between this algorithm and standard approaches, which typically process multiple frames of one input signal at time. The presented algorithm was verified using the database of forest fires captured under different environmental conditions. The achieved results show that the performance of the presented algorithm is comparable with existing similar techniques. The average value of the maximum detection time was 10 seconds, while 90% correct smoke detection and 10% false alarms were reported. It is important to note that the developed algorithm was tested for different environment conditions. In the future research, the optimal algorithm implementation will be targeted in order to realize a smoke detection system for real application.

ACKNOWLEDGMENT

This work was supported in part by the Slovak Research and Development Agency under grant APVV-15-0254 and by the Ministry of Education, Science, Research and Sport of the Slovak Republic under grant VEGA 1/0905/17.

REFERENCES

- [1] L. Shuai, W. Bo, D. Ranran, Z. Zhiqianq and L. Sun, "A Novel Smoke Detection Algorithm Based on Self Fast-tuning Background subtraction", Proceedings of Control and Decision Conference, Yinchuan, China, 2011.
- [2] P. B. Pagar and A. N. Shaikh, "Real Time based Fire & Smoke Detection without Sensor by Image Processing," International Journal of Advanced Electrical and Electronics Engineering , vol. 2, no. 6, pp. 25-34, 2013.
- [3] S.-H. Yoon and J. Min, "An Intelligent Automatic Early Detection System of Forest Fire Smoke Signatures using GMM," Journal of Information Processing Systems, vol. 9, no. 4, pp. 621-632, 2013.
- [4] S. Li, B. Wang, L. Gong, Z. Zhou and H. Wang, "A Novel Smoke Detection Algorithm based on MSER Tracking," Proceedings of Chinese Control and Decision Conference , Qinqdao, China, 2015.
- [5] T. S. Surya and M. S. Suchithra, "Survey on Different Smoke Detection Techniques Using Image Processing," International Journal of Research in Computer and Communication Technology, vol. III, no. 11, pp. 16-19, 2014.
- [6] J. Chen, Y. Wang, Y. Tian and T. Huang, "Wavelet Based Smoke Detection Method with RGB Contrast-Image and Shape Constrain," Proceedings of Visual Communications and Image Processing Workshop, Kuchin, Malaysia, 2013.
- [7] J. A. Ojo and J. A. Oladosu, "Video-based Smoke Detection Algorithms : A Chronological Survey," Journal of Computer Engineering and Intelligent Systems , vol. 5, no. 7, pp. 38-50, 2014.
- [8] A. Filonenko, D. C. Hernandez and K.-H. Jo, "Smoke Detection for Static Cameras," Proceedings of Korean-Japan Joint Workshop on Frontiers of Computer Vision, Mokpo, South Korea, 2015.
- [9] R. D. Labati, A. Genovese, V. Piuri and F. Scotti, "Wildfire Smoke Detection Using Computational Intelligence Techniques Enhanced With Synthetic Smoke Plume Generation," IEEE Transaction on Systems, Man., and Cybernetics, vol. 43, no. 4, pp. 1003-1012, 2013.
- [10] Y. Wang, A. Wu, Z. Jie, M. Zhao, W. Li and N. Dong, "Fire Smoke Detection Based on Texture Features and Optical Flow Vector of Contour," Proceedings of World Congress on Intelligent Control and Automation, Guilin, China, 2016.
- [11] L. Jinghong, Z. Xiaohui and W. Lu, "The Design and Implementation of Fire Smoke Detection System Based on FPGA," Proceedings of 24th Chinese Control and Decision Conference, Taiyuan, 2012.
- [12] I. Grubisic, D. Kolaric and K. Skala, "Intelligent algorithm for smoke extraction in autonomous forest fire detection," Proc. Microelectron. Electron. Technol./GRID Visual Syst., 2009
- [13] P. Foggia, A. Saggese and M. Vento, "Real-Time Fire Detection for Video-Surveillance Applications Using a Combination of Experts Based on Color, Shape and Motion," IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 9, pp. 1545-1556, 2015.