

## Research Article

# RGB Color Model Based the Fire Detection Algorithm in Video Sequences on Wireless Sensor Network

**Yoon-Ho Kim,<sup>1</sup> Alla Kim,<sup>1</sup> and Hwa-Young Jeong<sup>2</sup>**

<sup>1</sup> Division of Computer Engineering, Mokwon University, Daejeon, Republic of Korea

<sup>2</sup> Humanitas College, Kyunghee University, Seoul, Republic of Korea

Correspondence should be addressed to Hwa-Young Jeong; [hyjeong@khu.ac.kr](mailto:hyjeong@khu.ac.kr)

Received 30 August 2013; Accepted 6 March 2014; Published 9 April 2014

Academic Editor: Yuh-Shyan Chen

Copyright © 2014 Yoon-Ho Kim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Since the fire causes serious disasters, fire detection has been an important study to protect human life. Based on the deficiencies of existing fire detection on real-time and monitoring accuracy, the wireless sensor network technique for fire detection was introduced and needed. In this paper, we proposed the fire detection algorithm in video sequences on wireless sensor network. The proposed fire detection algorithm processes visual information acquired through static camera that lets us incorporate the algorithm to CCTV surveillance system, and therefore does not require an additional expanses on conventional fire sensors. Fire detection method based on motion information, any input image, is compared to background in order to identify foreground. Background is statistically modeled by mixture of Gaussians. To detect the foreground of video sequences, the proposed color detection algorithm was performed in RGB space. The procedure of algorithm eliminates all objects that do not fulfill color requirements without fire-like objects. And the change map and blob's area are computed, change map shows temporal variation of pixel between two consecutive binary frames, and percentage area increase or decrease characterizes a fire property for swinging.

## 1. Introduction

The fire is considered as one of the important and indispensable resources. In view of the conventional fire detection, it is necessary to bring in a new method for effective fire detection with higher false rates. One of the most important tasks for applications based on visual information, such as video surveillance systems, intrusion detection system, activity monitoring, and object tracking, aims to identify events of fire occurrence in a variety of situations [1–5]. In addition, it is important to inform the related department of the actual conditions of fire outbreak immediately on wireless sensor network. Fire detection applications based on CCTV cameras exploit a surveillance network that is already widely used in public areas. This is particularly useful in areas such as smart home network and forest fire prevention, where conventional fire detectors cannot be used [1, 3, 6].

The rest of the paper is organized as follows. The following section gives the related works of wireless sensor network and fire detection. Section 3 describes the proposed fire detection algorithms. It discusses the method and subalgorithms used

for fire detection in a video sequence with details. Section 4 presents the result of experiments and algorithm's comparison with algorithm proposed by [5]. Section 5 is devoted to conclusion and pointing on some further issues.

## 2. Related Works

In this section, we enumerate the various problems and challenges related to the task of fire detection on the wireless sensor network. Most of the proposed techniques for fire-like object detection rely on RGB color information [1–5, 7–9] to detect the fire, while calculating the changed map detection.

**2.1. Wireless Sensor Network.** Wireless sensor network [10–13] is emerging field and now becoming very attractive. Authors in [14] have focused on human activity recognition. The human activity recognition is a component which is used to provide real-time data management, to human activities are detected and to manipulate the detected activities. Authors in [15, 16] have presented an integration of

wireless sensor networks and cloud computing for U-life care. And the University of California at Berkeley took the lead research on wireless sensor networks for the application of monitoring forest fires, and developed the fire bug system [17]. Washington University scholars also launched a study in wireless sensor networks and used the mobile agents to find the fire source, which made the network with greater flexibility [18].

As shown in Figure 1, application layer gateway is needed to cost effectively deliver the sensor data from the link layer of wireless sensor network protocol to the network layer of IP protocol over the Internet.

**2.2. Fire Detection.** There are a lot of fire detection systems that used different rules to define fire objects in video sequence.

An overview of fire detection algorithms is given in [1], and the different concepts have been introduced to combat lightening conditions, image quality, and scene complexity. Moreover, the number of solution based on different fire features has been introduced recently.

Color information was previously used in many fire detection algorithms [2–5, 7, 19–22]. Motion information is also used to classify fire regions. A moving object detection algorithm in the preprocessing phase was employed to raise an alarm for possible fire in video [3]. Authors in [5] suggested another flame and smoke detection method in tunnel environments based on image difference and elimination stationary and moving lightening areas. In order to distinguish fire-like objects from true fire, a flame flicker identification technique based on statistical frequency counting was proposed [19]. Additional procedure that registers permanent changes was employed to overcome long-time changes [22]. Also, wavelet based fire detection method in video was developed by [8, 9]. Figure 2 shows the general flow for fire detection.

### 3. The Proposed Fire Detection Algorithm

The proposed fire detection algorithm consists of three subalgorithms: detection of moving pixels or regions in the current frame of a video, color detection of moving pixels, and blob analysis. The outline of proposed method is given below in Figure 3.

**3.1. Background/Foreground Modeling.** The raw input video is changed in a form that can be used and conformed to any algorithm's goal and capability of acquisition device. For foreground detection we have to choose a proper algorithm of background subtraction; a background model is a crucial task in the algorithm. Background modeling is a statistical description of the current background scene. In the algorithm, each background pixel was modeled by using a mixture of three Gaussians (MoG). MoG is parametric method and its parameters are adaptively updated that makes it more and more popular. Nonadaptive methods accumulated errors over the time without manual reinitialization and that makes nonadaptive methods useful only in scenes with no

significant changes. Besides MoG can maintain multimodal changes, which is particularly useful in outdoor scenes. A background pixel in a video frame is modeled as a random variable that follows the Gaussian distribution as in the following:

$$P(I_t) = \sum_{i=1}^K \omega_{i,t} \eta(I_t | \mu_{i,t}, \Sigma), \quad (1)$$

where  $K$  is the number of distributions and determined by the available memory and computational power,  $\omega_{i,t}$  is a weight associated with the  $i$ th Gaussian at time  $t$  with mean  $\mu_{i,t}$  and standard deviation is the covariance matrix of the  $i$ th Gaussian in the mixture at time  $t$ , and  $\eta$  is a Gaussian probability density function. There are typically 3–5 distributions per pixel as in the following:

$$\eta(I_t | \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} e^{-gpd} \quad (2)$$

here,  $gpd = -\frac{1}{2}(I_t - \mu_t)^T \Sigma^{-1} (I_t - \mu_t)$ .

And (3) assumes for computational reasons that the covariance matrix is to be of the form

$$\Sigma = \delta_{i,t}^2 I. \quad (3)$$

The distribution of recently observed values of each pixel in the scene is modeled by a MoG. That assumption lets us avoid some costly processing data at the expense of accuracy. Then each input pixel  $I_t$  is compared to the means  $\mu_{i,t-1}$  to find their associated component  $\hat{i}$  as in (4). If the absolute difference between the pixel and mean less than the component's standard deviation scaled by factor  $D$ , the component  $\hat{i}$  is considered as matched, otherwise unmatched:

$$\hat{i} = \begin{cases} \text{matched}, & |I_t - \omega_{\hat{i},t-1}| \leq D \cdot \delta_{\hat{i},t-1} \\ \text{unmatched}, & \text{otherwise.} \end{cases} \quad (4)$$

$D$  defines a small positive deviation threshold.  $D$  is a constant and is equal to the scalar 2.5 [6, 7, 23]. If none of the  $K$  distributions match the current pixel value, the least probable distribution is replaced with a distribution with the current value as its mean value and initially high variance ( $\mu$  and  $\delta$  are unchanged) and low prior weight (the weights decrease exponentially). The weights are replaced with the following:

$$\omega_{i,t+1} = (1 - \alpha) \omega_{i,t} + \alpha, \quad (5)$$

where  $\alpha$  is constant learning rate of the background. The learning rates handle the adaptation rate with a compromise between being fast enough to adapt to changes and slow enough to store useful temporal history. At fast adaptation rate, the distribution quickly becomes dominated by a single Gaussian.  $\alpha$  is a user-defined learning rated with  $0 \leq \alpha \leq 1$  [24]. In other papers  $\rho = \alpha$  or set manually [6, 23, 25],

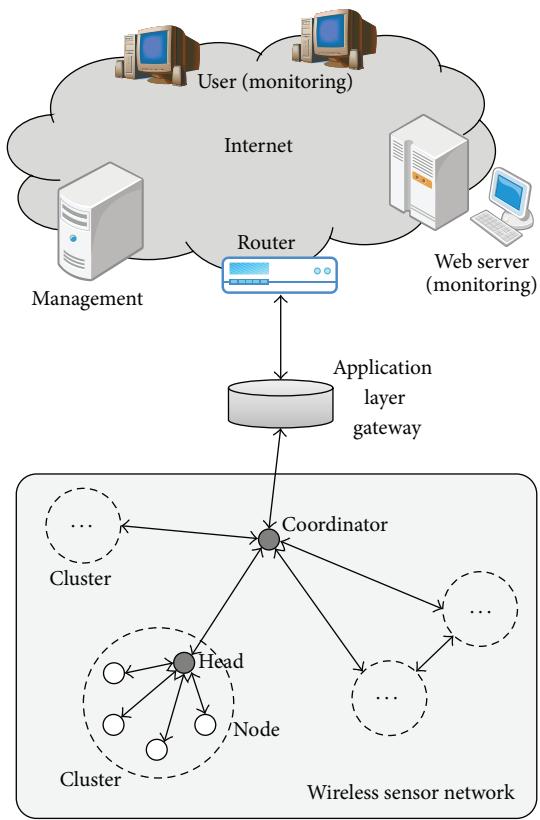


FIGURE 1: Wireless sensor network for fire detection.

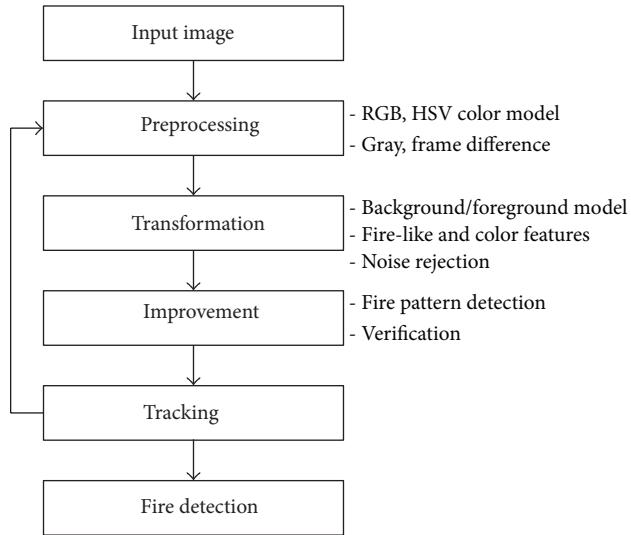


FIGURE 2: The general processes for fire detection.

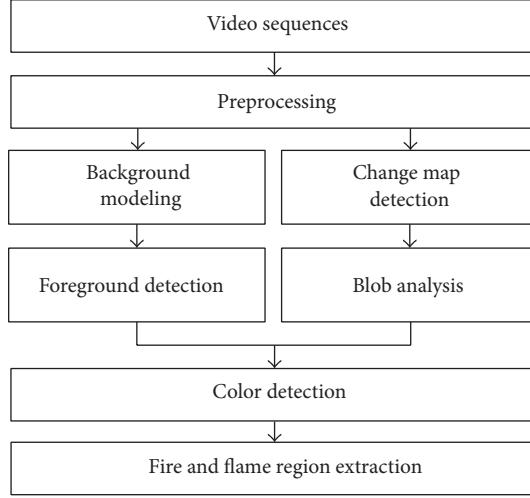


FIGURE 3: The diagram of the proposed fire detection.

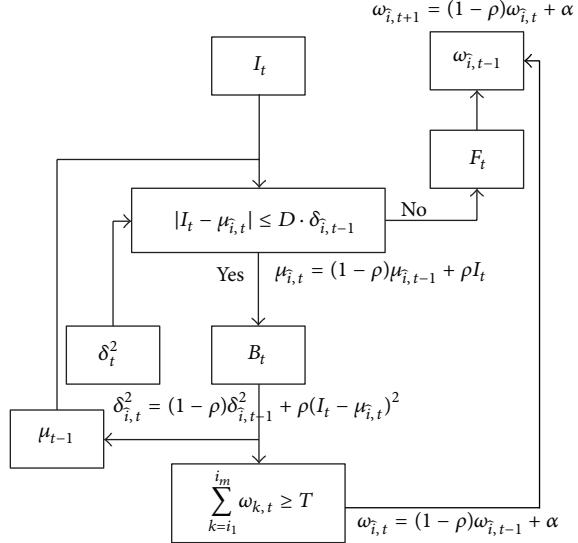


FIGURE 4: The MoG based BG/FG detection.

where  $\rho$  is the learning rate for the parameters and can be approximated as follows:

$$\rho \approx \frac{\alpha}{\omega_{i,t}}. \quad (6)$$

The parameters of the distribution which matches the new observation are updated as follows:

$$\begin{aligned} \delta_{i,t}^2 &= (1 - \rho) \delta_{i,t-1}^2 + \rho (I_t - \mu_{i,t})^2 \\ \omega_{i,t+1} &= (1 - \rho) \omega_{i,t} + \alpha \\ \mu_{i,t} &= (1 - \rho) \mu_{i,t-1} + \rho (I_t) \\ \sum_{k=i_1}^{i_m} \omega_{k,t} &\geq T. \end{aligned} \quad (7)$$

Once the parameters maintenance is made, foreground/background detection can be made and so on. The background model is then the first  $M$  components (in order of highest to lowest  $\omega_{i,t}/\delta_{i,t}$ ), whose weight is above the threshold. Higher-rank components have low variances and high probabilities, which are typical characteristics of background. The component  $M$  is the maximum number of components in the background model and reflects the number of modes we expect in the background probability distribution function. Pixels that do not match any components determined to be in the background model are considered to be foreground [23]. Figure 4 shows the process of background and foreground detection using MoG.

Fire detection algorithm requires defining and segmenting with maximum accuracy motion information, caused by fire. Table 1 is the list of parameters used in Figure 4.

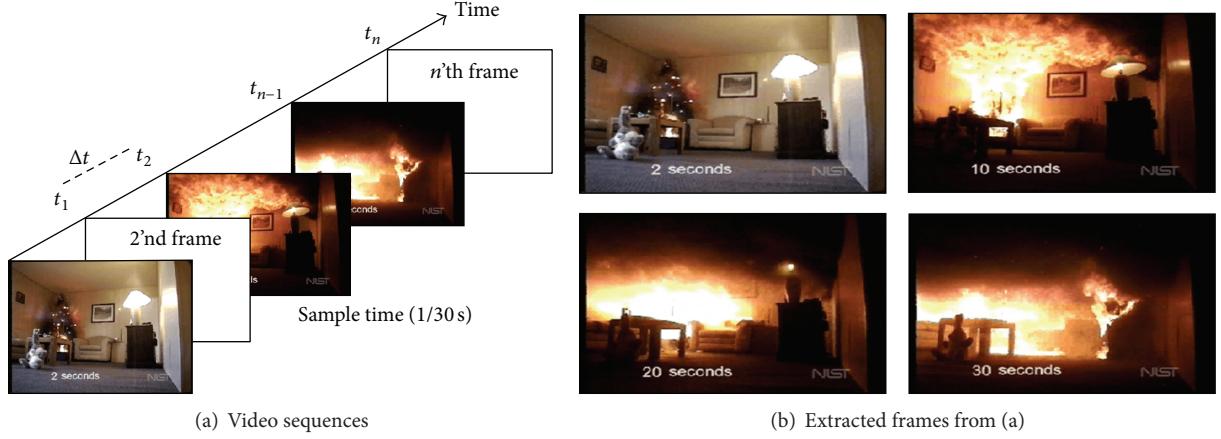


FIGURE 5: Video sequences for test.

TABLE 1: The list of parameters used in background/foreground detection algorithm.

Parameter	Value
$K$ (number of Gaussian components)	2
$D$	2.5
$\alpha$	0.01
$T$	0.75
$\delta$	10
$\rho$	$\alpha$

Fire detection algorithm requires defining and segmenting with maximum accuracy motion information, caused by fire. In this paper, we employ only color video input. The underlying foundation algorithm uses Mixture of Gaussian for background subtraction and background modeling.

**3.2. Color Detection.** Color information is often used in visual applications (face detection, food industry application, and number of quality control applications) in order to define objects more precisely: color information gives robust information that cannot be distorted by rotations and scaling and therefore does not require complex or time-consuming preprocessing. With complimentary features, color can be a powerful tool in image processing applications.

To identify fire-like colored objects, we developed color model similar to [20, 22]. First rule, pixel values in a red channel are greater than mean of pixels of red channel as shown in (8):

$$R_t(x, y) > R_{\text{mean}} \quad (8)$$

$$R_{\text{mean}} = \frac{1}{N} \sum_{i=1}^N R(x_i, y_i), \quad (9)$$

where  $R(x, y)$  is the pixel intensity value at time  $t$  in spatial location of  $(x, y)$ . In (9),  $R_{\text{mean}}$  is the mean of red channel pixels intensity values.

Second rule, pixels in red channel are supposed to be greater than pixels in green channel, and pixels in green

channel are supposed to be greater than pixels in blue channel as follows:

$$R_t(x, y) > G_t(x, y) > B_t(x, y). \quad (10)$$

Pixels which match the first rule and second rule are considered to be fire-like colored as shown in the following:

$$L_t(x, y) = \begin{cases} 1, & \text{if rule 1 \& rule 2 is true} \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

In order to perform fire-like region detection, a database of different resolutions images with the fire was collected and then color detection algorithm was implemented with (9). Formula (11) shows that the pixel has likely fire-like color if its intensity value in a red channel is greater than the mean of red channel pixel intensity values.

**3.3. Blob Analysis.** A blob is a region of connected pixels that are consolidated into a structure. In a binary image, pixels in the background have values equal to zero and every nonzero pixel is part of a binary object. Blob analysis aims to detect blobs and make measurements that give information about object in an image. The goal of the proposed algorithm is to make a decision whether each visible object is fire or not. For this one the proposed algorithm uses blob analysis for calculating statistics for labeled regions in a binary image.

Fires with changes (foreground objects) detected by background/foreground model in Section 3.1 are converted to binary image, where “0” represents background. Then dilation and opening with structural element were performed as in (12) and (13):

$$F \oplus S = \{\omega \mid \hat{S}\} \cap F \neq \phi, \quad (12)$$

where  $F$  is the image with detected changes,  $\phi$  is the empty state,  $S$  is the structuring element

$$F \circ S = \{F \ominus S\} \oplus S, \quad (13)$$

where  $F$  is the foreground image after dilation and  $S$  is structural element 3 by 3. After all, the hole-filling algorithm

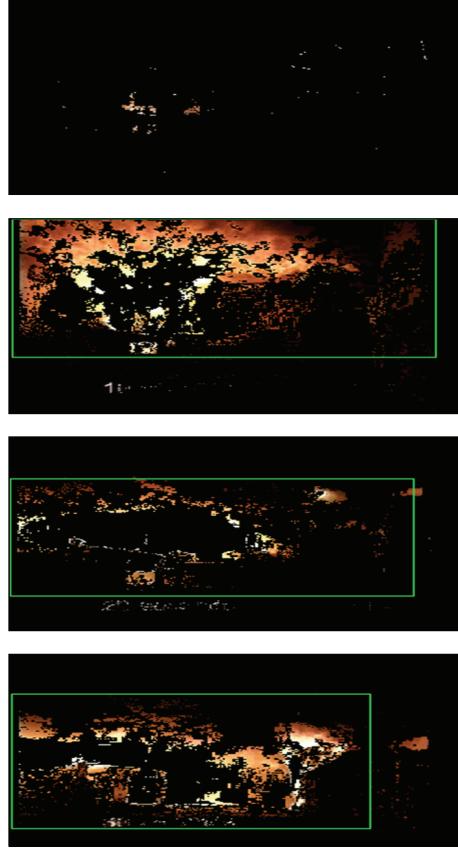


FIGURE 6: Detected foreground by GMM algorithm.

based on morphological reconstruction was used to fill all holes in detected foreground. The fire has a property to swing; its area changes significantly in consecutive frames.

To determine the percentage of object's area increase, area of binary blobs is detected for all input images by (14)–(16):

$$Tv = \frac{\text{area}(BI_{t+1}) - \text{area}(BI_t)}{\text{area}(BI_t)}, \quad (14)$$

where  $BI_t$ ,  $BI_{t+1}$  are input images converted to binary representation with threshold value  $T (= 230)$ ;

$$CM(x, y) = \begin{cases} 1, & \text{if } Tv > T \& \text{diff}(x, y) > 0 \\ 0, & \text{otherwise,} \end{cases} \quad (15)$$

where CM is change map, and  $T$  is threshold value that equal to 30. diff is a binary image difference between consecutive frames. Finally, it is possible to detect the fire by the following:

$$\text{Fire}(x, y) = \begin{cases} 1, & \text{if } CM > 0 \& L > 0 \\ 0, & \text{otherwise.} \end{cases} \quad (16)$$

In order to improve output, morphological operations (dilation, removing all isolated pixels, performing an interior fill operation, setting all pixel to 1 if all four-connected pixels are 1) are applied to  $\text{Fire}(x, y)$ .

## 4. Experimental Results

To evaluate fire detection algorithm, several tests on still images and video sequences were performed. In order to demonstrate algorithm's performance, indoors video sequence with ignited New Year tree with 122 frames and human motion in-doors 150 frames were chosen.

The proposed algorithm was implemented by using Matlab7; the test was performed on Intel Processor 2.83 HGz, 1.98 Gb of RAM. We acquired RGB video sequence with fire indoors as shown Figure 5.

The proposed algorithm was implemented as described in (1)–(15). The detected foreground pixels can have any values. Therefore, pixels are validated through color detection procedure as described in (8)–(10) in order to define fire-like objects. The proposed algorithm eliminates all objects that do not fulfill color requirements by [19–22].

The foreground area is detected, and then boundaries were constructed as shown in Figures 6 and 7. All fire-like areas are considered to be fire candidates. Here, algorithm detects not only fire, but also fire reflections on the wall from explosion, because they have motion and fire-like color; also false positives can give standard lamp which has bright regions and fire explosion. To overcome this, all objects, whose connected area in pixels is less than some predefined threshold, are removed; threshold is defined experimentally. That helps to improve output result.

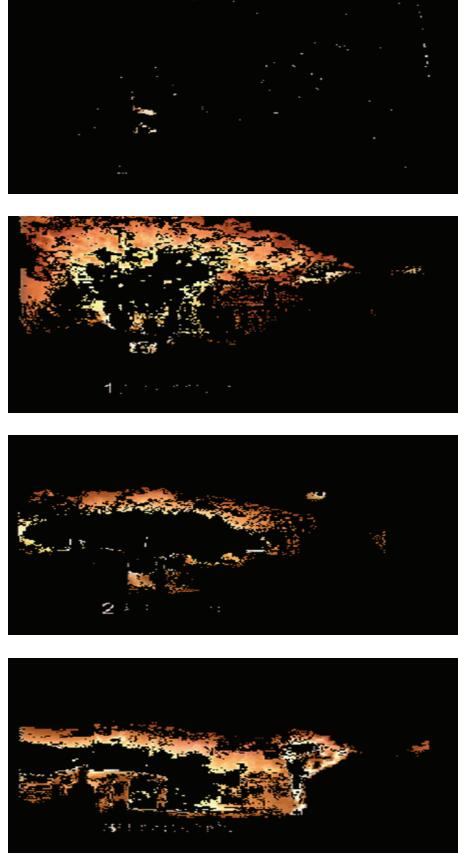


FIGURE 7: Detected the fire-like region by Section 3.2.

For performance comparison, algorithm proposed by [5] was chosen. For verification of lights during fire pattern verification, authors in [5] have to select the reference image of normal state. They make binary image which represents the region of pixel values which are bigger than the pre-determined threshold of RGB for consecutive input image and reference image. Then, they calculate differences between those input image and reference image by the following:

$$F_b = \begin{cases} 1, & R \geq T_R, G \geq T_R, B \geq T_R \\ 0, & \text{otherwise,} \end{cases} \quad (17)$$

$$N_b = \begin{cases} 1, & R \geq T_R, G \geq T_R, B \geq T_R \\ 0, & \text{otherwise.} \end{cases}$$

Equation (17) shows the binary image of the normal state image  $F(x, y)$  and the input image  $N(x, y)$  with the threshold value  $T$ . The difference of obtained binary image through (17) is calculated by (18). Figure 8 shows its implementation and approach of [5]

$$D_T(x, y) = N_b(x, y) - F_b(x, y)$$

$$\text{if } D_T(x, y) < 0 \text{ then } D(x, y) = 0 \quad (18)$$

$$\text{else } D(x, y) = D_T(x, y).$$

By (12)–(16), we calculate the change map between frames as shown in Figure 9. The change map indicates area changes and binary changes between consecutive frames. Then Figure 10 is made the fire object after morphological operations by the blob analysis.

Figure 11 shows the results of fire detection which is detected by the method of [5] and proposed algorithm. Figure 11(a) presents the result of fire detection algorithm that was introduced by [5], and Figure 11(b) is the output of proposed fire detection algorithm.

In order to have a quantitative evaluation of the performance, we have selected frames with no fire (Human motion, 150 frames) and within fire (Ignited New Year Tree, 122 frames) and evaluated next characteristics:

- (i) TP (true positives): frames that positively detected,
- (ii) FP (false positives): frames, where algorithm detected fire, while there is no fire,
- (iii) FN (false negatives): frames, where algorithm does not detect fire, while there is fire,
- (iv) TN (true negatives): frames, where algorithm does not detect fire, while there is no fire,
- (v) FD (fire detection): percentage of detected frames with fire:

$$FD = \frac{TP \times 100}{\text{fire frames}}. \quad (19)$$

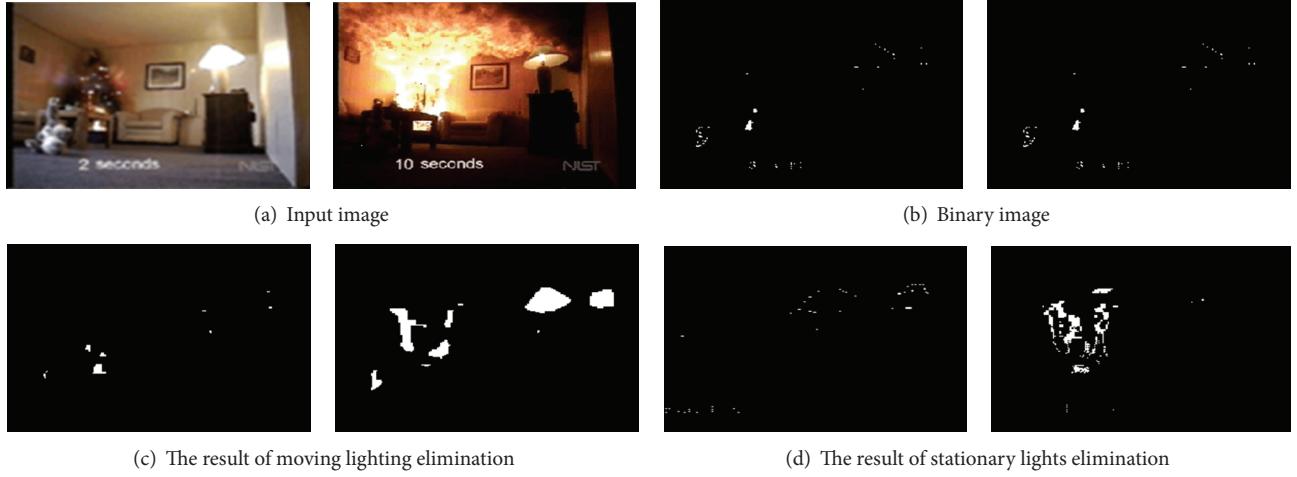


FIGURE 8: Fire detection in algorithm proposed by [5].

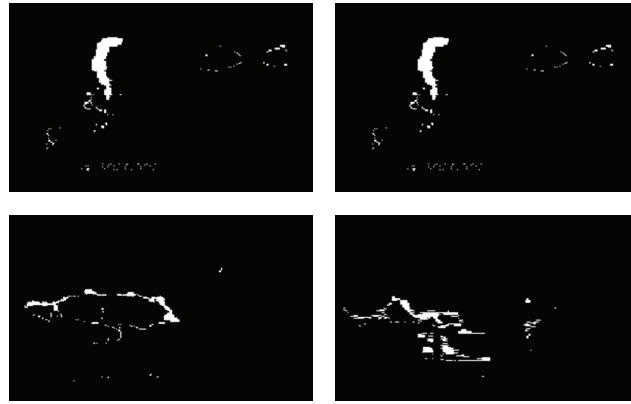


FIGURE 9: The results of the change map.

(vi) NFD (nonfire detection): percentage of frames with no fire that were correctly undetected:

$$NFD = \frac{TN \times 100}{\text{non-fire frames}}. \quad (20)$$

(vii) Overall: percentage of frames that was correctly detected by algorithm:

$$\text{Overall} = \frac{(TN + TP) \times 100}{\text{total frames}}. \quad (21)$$

The performance of algorithm is shown in Tables 2 and 3. It shows the percentage of detected frames with fire, nonfire, and overall performance of algorithm.

## 5. Conclusions

Wireless sensor network technology has greater advantage, and there are prospects for fire monitoring. The fire detection on wireless sensor networks, also, is still in the laboratory stage. In this paper, we proposed the color model that based the fire detection algorithm in video sequences on wireless

TABLE 2: The analysis of proposed fire detection algorithm.

Video sequences	Number 1	Number 2
Total frames	122	150
Fire frames	84	0
Nonfire frames	38	150
TP	78	0
FP	0	0
FN	6	0
TN	35	150
FD (%)	92.8	—
NFD (%)	92	100
Overall (%)	92.6	100

TABLE 3: The comparison between the proposed algorithm and [5].

	TP (%)	FP (%)	FN (%)
Proposed	92.8	0	7
[5]	88.3	2.7	9

sensor network. The proposed algorithm uses Gaussian mixture model for background modeling and background subtraction for detecting foregrounds. Fire has many different



FIGURE 10: The morphological operation results of Figure 9.



(a) Fire detection by [5]

(b) Fire detection by proposed algorithm

FIGURE 11: The results of [5] and proposed fire detection.

features that researchers use in their detection system, and color information gives useful data for video based fire detection. We have implemented color detection algorithm in RGB color space. Because color alone is not enough, we consider object's area growth in consecutive. Temporal variation for each pixel is estimated, for each fire-like colored object whose motion significantly differs from background and whose area changes frame-by-frame boundary box is constructed. The method can detect fire in video sequences that were acquired by static camera and can be useful in surveillance visual systems based on CCTV cameras in order to prevent fire damages on wireless sensor network.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] S. Verstockt, P. Lambert, and R. van de Walle, "State of the art in vision-based fire and smoke detection," in *Proceedings of the 14th International Conference on Automatic Fire Detection*, vol. 2, pp. 285–292, AUBE, Duisburg, Germany, September 2009.
- [2] W. Phillips III, M. Shah, and N. V. Lobo, "Flame recognition in video," in *Proceedings of the 5th IEEE Workshop on Applications of Computer Vision*, vol. 23, no. 1–3, pp. 224–229, December 2002.
- [3] T. Celik, H. Ozkamanli, and H. Demirel, "Fire and smoke detection without sensors: image processing based approach," in *Proceedings of the 15th European Signal Processing Conference (EUSIPCO '07)*, pp. 1794–1798, Poznan, Poland, 2007.
- [4] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in *Proceedings of the International Conference on Image Processing (ICIP '04)*, vol. 3, pp. 1707–1710, October 2004.
- [5] B. M. Lee and D. G. Han, "Real-time fire detection using camera sequence image in tunnel environment," in *Advanced Intelligent Computing Theories and Applications*, vol. 4681 of *Lecture Notes in Computer Science*, pp. 1209–1220, Springer, Berlin, Germany, 2007.
- [6] A. Kim and Y. H. Kim, "Intrusion detection algorithm based on motion information in video sequence," *Journal of Korea Navigation Institute*, vol. 17, no. 3, pp. 284–288, 2010.
- [7] M. Piccardi, "Background subtraction techniques: a review," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '04)*, pp. 3099–3104, October 2004.
- [8] B. U. Töreyin, Y. Dedeoğlu, U. Gudukbay, and A. E. Çetin, "Computer vision based method for real-time fire and flame detection," *Pattern Recognition Letters*, vol. 27, pp. 49–58, 2005.

- [9] B. U. Töreyin, Y. Dedeoğlu, and A. E. Çetin, "Wavelet based real-time smoke detection in video," in *Proceedings of the 13th European Signal Process Conference (EUSIPCO '05)*, Antalya, Turkey, September 2005.
- [10] M. Yoon, Y.-K. Kim, and J.-W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, no. 1, pp. 15–22, 2013.
- [11] J. H. Chong, C. K. Ng, N. K. Noordin, and B. M. Ali, "Dynamic transmit antenna shuffling scheme for MIMO wireless communication systems," *Journal of Convergence*, vol. 4, no. 1, pp. 7–14, 2013.
- [12] B. Singh and D. Lobiyal, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," *Human-Centric Computing and Information Sciences*, vol. 2, article 13, 2012.
- [13] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, article 13, 2013.
- [14] A. M. Khattak, L. T. Vinh, D. V. Hung et al., "Context-aware human activity recognition and decision making," in *Proceedings of the 12th IEEE International Conference on e-Health Networking, Application and Services (Healthcom '10)*, pp. 112–118, Lyon, France, July 2010.
- [15] X. H. Le, S. Lee, P. T. H. Truc et al., "Secured WSN-integrated cloud computing for u-life care," in *Proceedings of the 7th IEEE Consumer Communications and Networking Conference (CCNC '10)*, pp. 1–2, Las Vegas, Nev, USA, January 2010.
- [16] H. Jameel, R. A. Shaikh, H. Lee, and S. Lee, "Human identification through image evaluation using secret predicates," in *Topics in Cryptology—CT-RSA 2007*, vol. 4377 of *Lecture Notes in Computer Science*, pp. 67–84, Springer, Berlin, Germany, 2007.
- [17] "FireBug," <http://firebug.sourceforge.net>.
- [18] C. L. Fok, G. C. Roman, and C. Lu, "Tracking fires using mobile agents in a wireless sensor network," in *Proceedings of the 4th International Conference on Information Processing in Sensor Networks*, Los Angeles, Calif, USA, April 2005.
- [19] J. Chen, Y. He, and J. Wang, "Multi-feature fusion based fast video flame detection," *Building and Environment*, vol. 45, no. 5, pp. 1113–1122, 2010.
- [20] T. Celik and K.-K. Ma, "Computer vision based fire detection in color images," in *Proceedings of the IEEE Conference on Soft Computing on Industrial Applications (SMCia '08)*, pp. 258–263, June 2008.
- [21] G. Marbach, M. Loepfe, and T. Bruppacher, "An image processing technique for fire detection in video images," *Fire Safety Journal*, vol. 41, no. 4, pp. 285–289, 2006.
- [22] T. Celik, H. Demirel, H. Ozkaramanli, and M. Uyguroglu, "Fire detection using statistical color model in video sequences," *Journal of Visual Communication and Image Representation*, vol. 18, no. 2, pp. 176–185, 2007.
- [23] S.-C. S. Cheung and C. Kamath, "Robust background subtraction with foreground validation for urban traffic video," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 14, pp. 2330–2340, 2005.
- [24] A. McIvora, Q. Zangb, and R. Kletteb, "The background subtraction problem for video surveillance systems," in *Proceedings of the International Workshop on Robot Vision*, vol. 1998 of *Lecture Notes In Computer Science*, pp. 176–183, February 2001.
- [25] C. Peng, "Video background/foreground detection implementation on TMS320C64/64x+ DSP," Application Report, Texas Instruments, 2007.