Real-time Building Fire Detection and Segmentation in Video using Convolutional Neural Networks with Gaussian threshold approach

SRIDHAR P
Amrita Vishwa Vidyapeetham

Senthil Kumar (t_senthilkumar@cb.amrita.edu)
Amrita School of Engineering Coimbatore Campus https://orcid.org/0000-0001-8160-7223

Latha Parameswaran
Amrita Vishwa Vidyapeetham

Venkata Ramana Murthy Oruganti
Amrita Vishwa Vidyapeetham

Research Article

Keywords: Fire detection and segmentation, feature maps, area, centroid, growth rate.

Posted Date: April 8th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1524107/v1

Real-time Building Fire Detection and Segmentation in Video using Convolutional Neural Networks with Gaussian threshold approach

Sridhar P[1],
Department of Computer Science and Engineering,
{cb.en.d.cse17011@cb.students.amrita.edu}
Senthil Kumar Thangavel*,
Department of Computer Science and Engineering,
{t_senthilkumar@cb.amrita.edu}
Latha Parameswaran,
Department of Computer Science and Engineering,
{p_latha@cb.amrita.edu}
Dr. Venkata Ramana Murthy Oruganti,
Department of Electrical and Electronics Engineering,
{ovr_murthy@cb.amrita.edu}
Amrita School of Engineering, Coimbatore,
Amrita Vishwa Vidyapeetham
India

Abstract – Computer vision-based fire flame detection system has recently begun to replace traditional fire detection systems. The conventional imaging-based system uses color models and temporal analysis algorithms to detect and segment fire in an image or a frame. Color model algorithms could not effectively distinguish between fire and fire-like objects in real-time videos. Besides, temporal analysis between structures has not yielded higher detection accuracies. This work proposes a lightweight convolutional neural network (CNN) combined with Gaussian distribution threshold probability to detect and segment the fire in a frame sequence. The proposed hybrid work extracted the features and automatically classified the frames using simple network architecture, and the Gaussian probability threshold method improves the detection of fire region localization. Experimental performance of the proposed work increases the detection accuracy and better false alarm performance. This paper can predict the severity of the fire, area, and centroid, fire probability. The proposed work using low-resolution frames will surely help the real-time implementation system.

Keywords—Fire detection and segmentation, feature maps, area, centroid, growth rate.

I. INTRODUCTION

The devastating effects of fire accidents in buildings are increasing day by day. A fire accident takes a toll on human life and harms the economy and the environment. If we can detect fire accidents early, we can avoid or minimize human loss and economic loss. So catching fire at the initial stages is vital. If we fail to control the fire at an initial phase (less than 30 sec), it can cause a great catastrophe. Conventional fire alert systems use multi-sensor such as smoke, heat, and temperature. However, such sensors are affected by other parameters such as threshold delay and transportation delay; to address these limitations, fire detection is still a challenging problem. Vision sensors-based flame detection systems are encouraged by several advantages including.1—reliable fire detection (a conventional sensor may not give reliable detection).2.Large areas can be covered by a single camera.3.Faster detection (light travel much faster than heat or smoke).4.No need to reach threshold and transportation delay.5.According to the available fire scene, the amount of fire spread, direction, centroid can be detected. The vision sensor detection method not only eliminates the setback of traditional methods but also characterizes the fire. These characteristics attracted and motivated the researchers to develop a vision-based fire detection algorithm. Although researchers have developed several fire detection approaches using computer vision approaches, a few challenges still exist, such as illumination changes, flickering, low-resolution frames, and color shifts of fire-like objects. This is the current concern of the vision-based fire detection problem [1]–[10]

To overcome the problems mentioned earlier, we propose Hybrid fire detection and segmentation framework using convolutional neural networks (CNNs) and the Gaussian threshold method. The several vision-based fire detection approaches use color models, moving object detection, and probability-based approaches. The color model and probability techniques differentiate fire regions from non-fire by setting rules or parameter values. It is not suitable for several nested ring colors of fire.

The proposed hybrid work incorporates an automatically extracted deep feature vector of the network, which can discriminate the fire pixels at the initial stage under various illumination levels. The Gaussian threshold method improves the detection accuracy of the fire pixel at low-resolution frames/images. For this motivation, the hybrid work used CNN as a lightweight architecture and Gaussian probability threshold for this fire detection and localization problem. Another general problem addressing is the camera resolution. The proposed work ensures fire detection with low-resolution surveillance cameras [11]–[16]

<u>The contributions of this paper are as follows</u>

1. We train lightweight deep learning architectures on a diverse dataset. It allows the detection of fire accidents at an early stage. It can minimize damage.
2. Gaussian threshold method improves the Localization of the fire region after classifying the fire region in lightweight CNN.
3. Our hybrid work can detect and localize the fire region then it can be helpful to predict the characteristics of the localized fire region.
4. The proposed model yields better accuracy and lower false alarm rates than State-of-the-art approaches.
5. The more compact architecture and simple Gaussian thresholding have embedded processing capabilities.

The remaining of this article is arranged as follows. We review the literature on the traditional and latest image-based detection algorithms in Section II. Our proposed architecture is introduced in Section III. The data sets with experimental results and performance evaluations are yielded in Section IV. Lastly, the work is concluded in Section V

II. Related work

Recently, several flame pixel detection methods have been proposed by researchers [17]–[19]. We now present different literature works along with their strengths and limitations. Sowah et al.[20] used fire, heat, and temperature sensors. The sensors which are placed in very close proximity to fire accident areas only can detect. Though the sensor provides high detection accuracy, it may not produce reliable detection for various environmental conditions.

Celik et al.[21] proposed rule-based fire pixel classification using a generic YCbCr color model. The fire pixel was identified by novel rules. The rules were formed by experimental analysis of the fire region. However, their work detects fire-like objects also and produces a high false alarm rate. Giuseppe Marbach et al. [22] proposed an image processing technique based on real-time fire detection.YUV color space was used in work. They used two phases for identifying the candidate fire region. In the first phase, flickering and luminance properties were used in the temporal domain. In the second phase, six fire features were used for evaluating fire patterns. However, the method yields an even less false alarm rate in all environmental conditions per week, in specific environmental condition and certain position of the camera it gives the high false alarm rate. Chen et al. [2] proposed Color and time analysis between frames for early flame detection and Localization in video sequences. It is shown that the low-temperature flame has the Color red to yellow spectrum range. The

flame spectrum change when the temperature increases. The fire may be white Color if the temperature is high. It indicated that the high-temperature flame has low saturation, and low temperature produces high saturation. In the chromatic study, this work proposed three decision rules for detecting the fire pixel. Unfortunately, few fire-like object colors in an image are distinguished as fire. The frame disorder characteristic is measured to enhance the fire detection system reliability. This dynamic feature is used to explore if the flame area will spread or not based on the frame threshold value of two subsequent frames. This method provides reliable flame pixel detection, reduces the false alarm rate, reliable flame segmentation during various illumination conditions, is affordable cost, and gives additional information of flame if the fire is growing or not. This early fire detection approach is not only for forest fire but also for explosive fire. This method has irregular rules for fire detection. Wen-Bing Horng et al.[3] applied a simple image processing work for the flame detection system. This algorithm used HSI color space and 70 training images to construct the fire feature vector. This predictor is used for creating the rules. This ruled-based work has not entirely removed the non-fire pixel. This method could not remove the spurious region of fire. This approach with one more way is applied as subsequent frame difference and color making. This complete work provides high detection accuracy and removes spurious flame regions. Though this approach provides good detection accuracy compared to the other works, the false alarm rate is very high. The experimental result shows 35% of false-negative and 81% false-positive. Adnan Khalil et al. [23] proposed a multi-space color model and motion detection for fire objects detection with a reduced number of parameters. The work explored RGB and LAB color space for segmenting fire and fire-like regions. To remove the fire-like regions, the Gaussian mixture model (GMM) was used to detect the moving (fire) objects. GMM models have given high performance than other models. The work provides high detection accuracy but false positive rate is still high (88.81%).

Kwang-Ho Cheong et al.[24] proposed Support Vector Machine (SVM) based fire detection and Localization. Their work is detected the fire pixel by combining four methods probability, hybrid background subtraction, temporal luminance, and SVM correspondingly. In addition to these methods, anyone method which fails others can lift the work. However, the work has consumed an extended processing time and produced high false alarms. Walter Philips et al. [25] presented two methods to detect fire. They computed Color and motion information in the video frames to detect the fire region. They used the color histogram method and temporal variation of the pixel which was used to localize the fire region. The spurious regions are removed by using erode operation. This work specifically is used for camera motion. This work yields a high false-negative alarm.

Amin Khatami et al. [26], [27] proposed an early and accurate automated fire detection algorithm based on a computer-vision approach. This work has addressed early fire detection by using the image processing technique. This method provides a new fire color space through the linear conversion of the fire and non-fire samples multiplication using a weight matrix. The feature vectors contain the fire and non-fire pixels, which are labeled manually. Particle Swarm Optimization (PSO) technique uses the conversion weight matrix that can migrate this feature vector into a new fire color space. The fire color space feature vector is labeled using K-medoids clustering or Fuzzy C-means clustering. The changes in the labels are identified as an error. The conversion matrix updates the weight value by the proper PSO approach. The final updated weight values differentiate the fire and non-fire pixels. This algorithm is fast with high detection accuracy of fire and background; fire-like objects can be differentiated. The method detects the flame color from red to yellow; it fails when a high similarity between fire and fire-like color regions exists. Qiu et al. [28] applied an auto-adaptive edge detection algorithm to detect the edge of the fire region. The frame/image has been initially

enhanced by histogram equalization and noise removal. The edges are detected by using the Sobel operator. The detected edges are discontinuous. The work handles two threshold values T_H and T_L, for removing the discontinuity. The LMS (Least mean square) algorithm chose the appropriate threshold values. The work has a few limitations: (1) it does not describe the fire region characteristics. (2) It does not work correctly in fire-like objects.

Many deep learning models have been proposed for fire detection and Localization. Qingjie Zhang et al. [29] introduced a cascaded fashion of fire detection. The author presented two methods, namely patch classifier, and deep CNN. The patch classifier learns from annotated patches. The patch size is 32x32, and the patch classifier uses CNN from the Caffe framework [30]. The patch classifier is a cascade with deep CNN, which is 8- a layer CNN model. The full image is given to the deep CNN classifier. This work yielded better accuracy and a minimal number of false alarms. Khan Muhammad et al. proposed various variants of deep learning frameworks, AlexNet [31], GoogleNet [32], and SqueezeNet [33]. AlexNet [31] is outperforming traditional hand-crafted features fire detection methods. It detects fire in both indoor and outdoor environments well.

When compared with other methods such as color model-based methods, machine learning methods, and a few deep learning models, the approach detects fire quickly with reasonable accuracy. The architecture has a 5-convolution layer, 3-Maxpooling layer, and 2-Fully connected layer. Since this work successfully faced many challenges, not localized the fire region and did not characterize the fire. AlexNet model size is 238 MB. The SqueezeNet architecture has two convolution layers, three max-pooling layers, one average pooling layer, and eight fire modules. This model detects and localizes the fire region in various resolution images. Khan Muhammad et al. [33] extracted the feature maps from the 8th, $26^{th,}$ and 32nd from Fire2 modules of SqueezeNet, then find the Mean activation map and finally localized the fire region. The model size is 3MB. Compared with AlexNet [31], the Squeeze Net model makes its hardware implementation more feasible in surveillance network cameras. Khan Muhammad et al. [32] proposed a model similar to GoogleNet. The model compared with AlexNet yields better detection accuracy, and the model size is less. The model consists of 100 layers with two convolution layers, four max-pooling layers, and seven inception modules. This model uses different kernel size1x1, 3x3, and 5x5. The model extracts different scales of features. The purpose of the kernel 1x1 is to reduces the parameters numbers. This model is trained by inception modules which add the loss with the main loss. The model also provides very low false alarm rate. Arpit Jadon et al. [34] proposed like a proposed model kind light weight architecture which is called FireNet. The architecture has 3 convolution layers, Average pooling layers and dense layer. The total no of parameters have 7.45MB only. But it could not localize the fire region. The work used low cost embedded board to detect the fire. The survey of all these deep learning models shows computationally expensive model yields better accuracy low false alarm. In this few survey works have feasibility of real time implementation also. Even though, the proposed work provides the same with lesser computations and early detection.

III. Proposed Framework for Surveillance Video Fire Detection

The flow diagram for the proposed hybrid fire detection and localization system can be seen in Fig.1.First, we get the fire detection probability from light CNN. In the real-time videos, if non-fire frames are present, no more operation is performed; if fire-frames are present, the identified fire-sensitive layer gives the feature map that helps localize the fire region. Furthermore, the Gaussian probability-based threshold logically comines with the feature map to improve the Localization of fire pixels in low-resolution frames/images.

A.Light CNN model

The proposed architecture related to the CNN architecture with light modification is presented now. The architecture contains five convolution layers, two max-pooling layers, and two fully connected layers, as shown in Fig.2, and weight parameters are shown in Table.1.The input frame size of this model is 64x64x3. The input size implies a reasonable resolution for detecting the fire in a short time [35][36]. The first convolution layer has 16 filters with the size of 3x3 of stride 1. Because of the very low frame resolution, we chose stride as 1. This feature map is given to the input of the second convolution layer. The filters are convolved on input images to generate the 16 feature maps of size 62x62. There are two reasons for choosing the minimum number of filters. The survey paper [26] uses only a simple (3x3) weight filter, and it gives good accuracy. The minimum number of parameters was to avoid overfitting. Here we used filter size 3x3. This filter size is suitable for capturing fine details in a frame widely used for detection problems. The feature maps are moved into the second and third convolution layers. This layer also uses 16 filters with stride 1. This feature The stacked convolution layer allows hierarchical decomposition of the input. The maximum activation values are obtained after two convolution operations by using the max-pooling layers with stride 2. The stride value of 2 reduces the feature maps size by 2. In the end, two fully connected layers are used. We have tested the output with more than two connected layers. But the results have not changed significantly. We select the Leaky ReLu activation function for five convolution layers and also to this first fully connected layer. To avoid that deactivation of neurons, we chose Leaky ReLu activation instead of the ReLu activation function. This last layer (fully connected layer) is fed into Softmax. It computes the fire detection probabilities for binary classes.

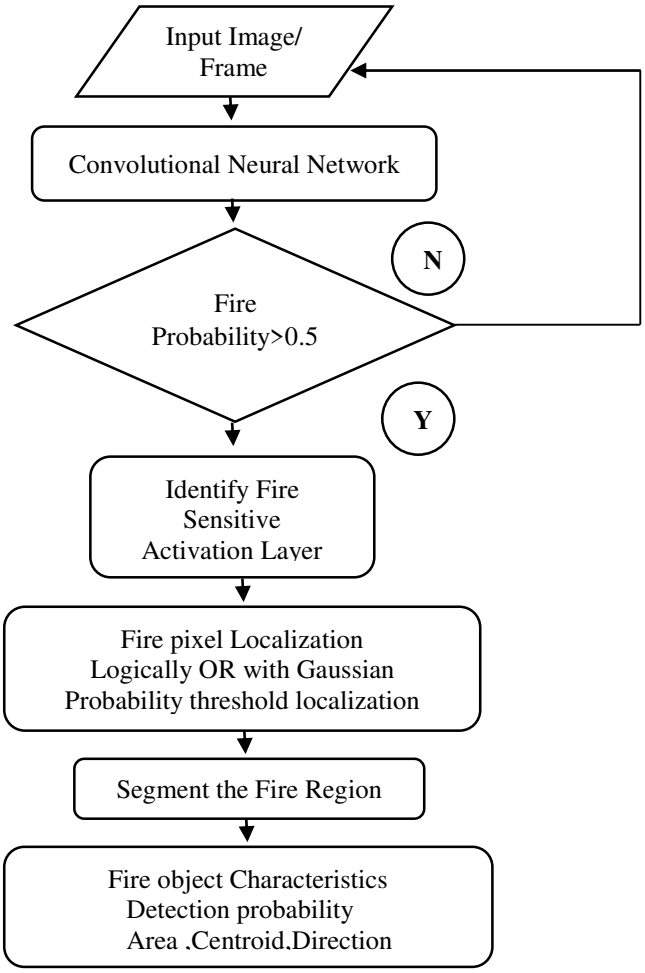


Fig.1 Flow diagram of light CNN for fire localization

Table 1 Light CNN architecture

S No	Layer	Filters	Size	Input	Output	Padding
1	Convolutional Layer 1	16	3x3/1	64x64x3	62x62x16	0
2	Convolutional Layer 2	16	3x3/1	62x62x16	60x60x16	0
3	Convolutional Layer 3	16	3x3/1	60x60x16	60x60x16	1
3	Max pooling Layer 1		3x3/2	60x60x16	29x29x16	0
4	Convolutional Layer 4	16	3x3/1	29x29x16	27x27x16	0
5	Convolutional Layer 5	1	3x3/1	27x27x16	25x25x1	0
6	Max pooling Layer 2	1	3x3/2	25x25x1	12x12x1	0
7	Flat				144	
8	Fully connected Layer 1			144	100	
9	Fully connected Layer 2			100	2	

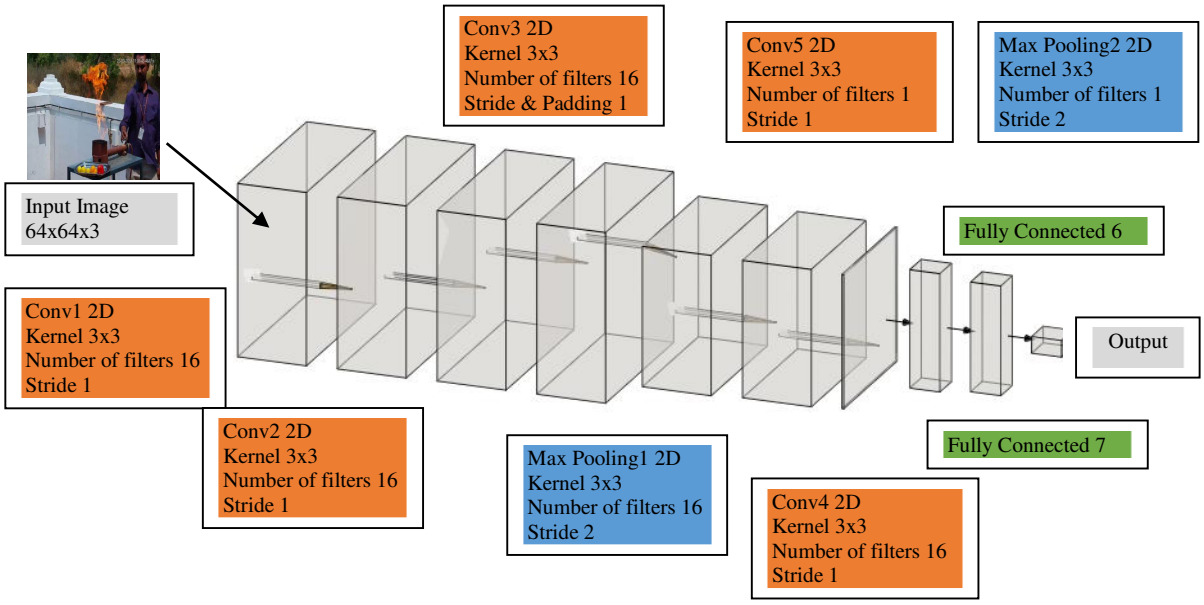


Fig.2.Proposed Deep learning framework architecture for fire detection

B. Fire pixel Localization by Gaussian probability threshold:

Chen et al. [2] used the early fire detection method on videos. This method uses a red plane threshold which is vital in flame pixel detection and for saturation threshold computation. Even though this approach needs some thresholds for fire pixel detection, it cannot detect all varying illumination of the fire environment or when burning material is changed. Hence, in this method, fire pixel candidates are detected using RGB probability distribution. Each channel uses a pixel distribution independently. This distribution is fit into the Gaussian probability distribution of each channel alone which is estimated as:

$$p_c = \frac{1}{\sqrt{2\pi\pi_c}} \exp - \left(\frac{(I_c(x,y) - \mu_c)^2}{2\sigma^2{}_c} \right), \qquad c \in \{R, G, B\} \tag{1}$$

Where Ic (x, y) indicates the pixel intensity of channel c in a frame, μc is the mean value of the color space and σ_c is the standard deviation of the fire region of the normal distribution. Hand segmentation technique has been app1ied to 120 RGB fire images, and 1,20,269 fire pixels are obtained for each channel, as shown in Fig 3 and Fig.4 (a). Fire pixel distribution is checked for each channel; if the distribution is not Gaussian or normal, the model is fit into the normal distribution, as shown in Fig 4 (b). The probability density function value for the Red channel is shown in Fig 5. Similarly, the pdf of green and blue channels is computed.

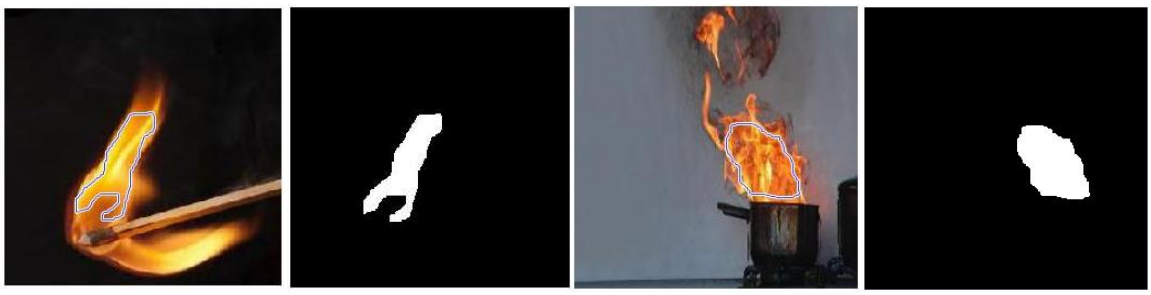
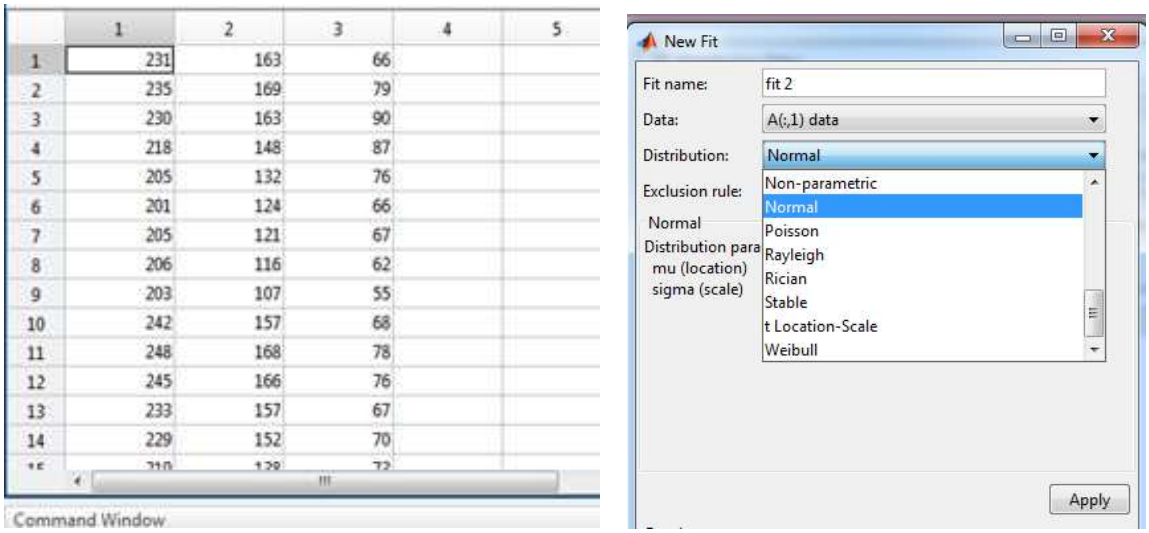


Fig.3. Hand segmentation is done manually in two sample images and its monochrome image



	1	2	3	4	5
1	231	163	66		
2	235	169	79		
3	230	163	90		
4	218	148	87		
5	205	132	76		
6	201	124	66		
7	205	121	67		
8	206	116	62		
9	203	107	55		
10	242	157	68		
11	248	168	78		
12	245	166	76		
13	233	157	67		
14	229	152	70		

(a) (b)

Fig.4. (a) fire region pixel values of 120 images (each channel) (b) fitting into Normal distribution to get the pdf values

The global probability threshold value of a fire pixel is computed as the product of each channel probability. The threshold value is set as 2.469x10-6

$$p(I(x,y)) = \prod_{c\in\{R,G,B\}} p_c(I(x,y))$$

$$\begin{cases} \text{if } \; p(I(x,y)) > \tau & \text{Fire pixel} \\ \text{else} & \text{Non - Fire pixel} \end{cases} \tag{2}$$

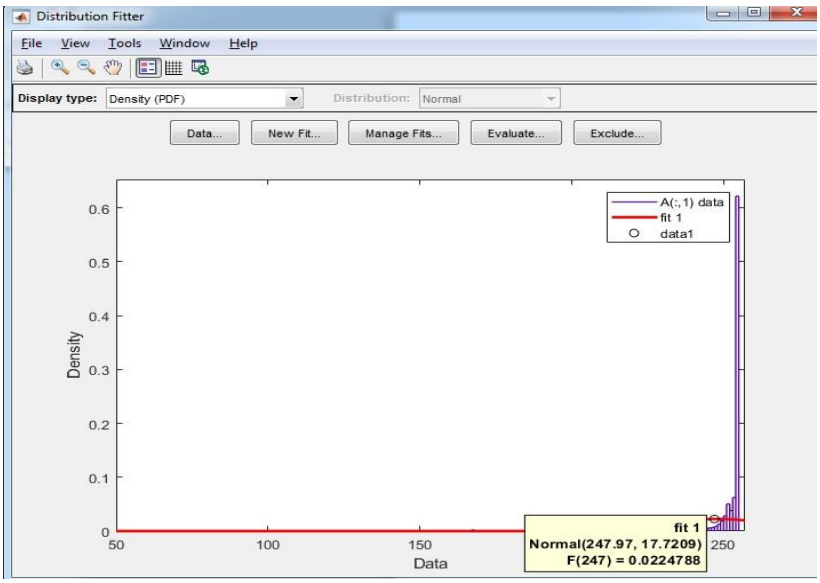
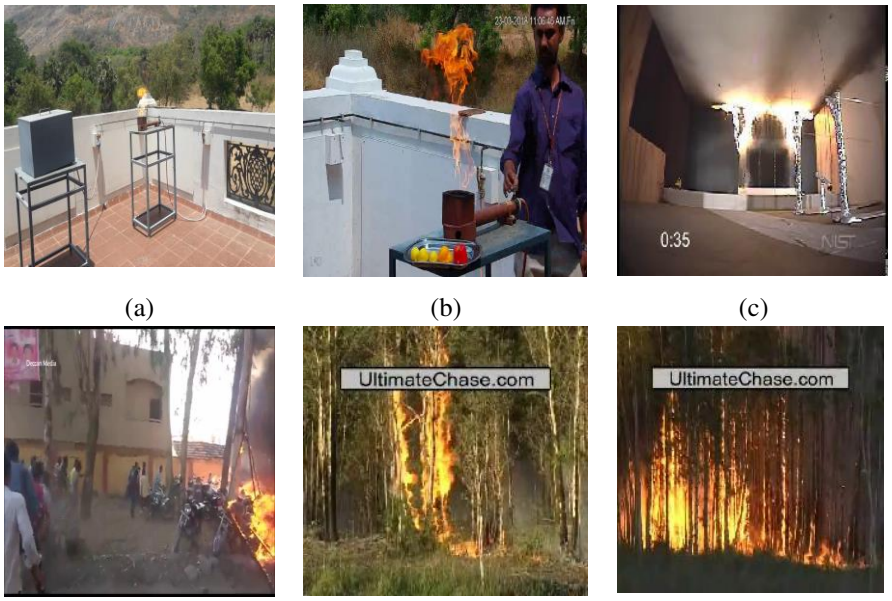


Fig.5. Graph showing the red channel fire pixel distribution.

IV. Experimental Results and discussion

The section describes data sets in detail with the detection performance of our method are compared it with other fire detection approaches. We have been trained our system with a machine with the following specifications: NVidia GeForce GTX 1050 4GB onboard memory and Tensor flow deep learning framework and Windows 10 OS platform on an Intel Core i5 CPU with 8GB RAM. The Gaussian probability threshold and remaining simulation work have been developed using MATLAB R2019b for fire region characteristics. There were a total of 25 fire and 26 non-fire videos used. Our model was trained, validated, and tested by the following datasets as shown in Fig.6: (1) Amrita object Library [37] (2) NIST [38] (3) YouTube (4) Visifire [39] (5) mivia [40] and Downloaded images. Amrita objects Library videos [37] contains fire-like objects such as red cloth, apple, orange, banana, and corn, and few datasets have been night environment videos. YouTube videos support real-world fire accident locations and fire-like objects such as sun, fire color lights, etc. The remaining Dataset contains a standard fire detection dataset (NIST [38], Visifire [39], mivia[40]) and downloaded images. Muhammad et al.[31]–[33] they have used Foggia's Dataset [41], containing 14 fire videos and 17 non-fire videos. It contains many similar frames, which affects the performance while training the model. As we examined the datasets of the past approaches, there is a lack of a diverse fire dataset. Accordingly, we apply the data augmentation process to our Dataset. This strategy has been used to increase the diversity of datasets for training. We have applied a few augmentation techniques such as flipping, frame rotation is specified degrees, cropping, and illumination changes. The data augmentation technique has been done with Keras deep learning library. Although the Dataset appears small, it is diverse. The video dataset contains 9118 fire images and 6200 non-fire images. These videos were captured by various illuminations, indoor and outdoor environments, various resolutions, different frame rates, and nested rings of fire color. This is the main reason for choosing the Dataset for training and evaluating our work. The detail of the Dataset is shown in Table 2. We used 20% images for training and 80% images for testing from this dataset. The proposed lightweight CNN was trained by1541 fire and1522 non-fire images; there were used 3063 images for training. Moreover, there are 20% validation images used from the training dataset. We evaluated the performance of our proposed work by testing datasets. We selected unused normal frames for this evaluation and generated challenging frames in the standard dataset, different environment (indoor and outdoor) frames in the YouTube videos, and Downloaded images. In the proposed light CNN model, with the Leaky ReLu activation, we use a coefficient value of 0.33. For solving the problem, the model requires a lot of labeled datasets for learning. The stochastic gradient descent (SGD) with a 0.9 momentum value is set. The initial learning rate is 0.01. 0.25x decreases it for every ten iterations. We trained the proposed model for 200 iterations. The initial weights in the proposed model are assigned randomly. We dropout 0.5 in the fully connected layers to avoid the over fitting.



(a) (b) (c)

(d) (e) (f)

Fig.6 representative images of test dataset (a-b) Amrita object Library (c) NIST (d) YouTube (e) Visifire (f) mivia

Table 2 Details of Dataset

Dataset Name	Total No of Frames	Resolution	Frame rate/ sec	Environment	Color	Image/ Video Format
NIST						
NIST 1, NIST 2	2873	492x360	29	Indoor	White	.jpg
Visifire						
controlled2, controlled3	245,207	400x256	15	Outdoor	Yellow	.jpg
FbackYardFire	1200	320x240	15	Outdoor	Orange	.jpg
fire1,forest1, forest2,forest3, forest4,forest5, ForestFire1	707,199 244,254 218,215 217	320x240	15	Outdoor	Orange	.jpg
Mivia						
fire2, fire13, fire14	139,300,600	320x240	29,25,25	Indoor, Indoor Outdoor	Yellow & Red, White, Orange	.jpg
Amrita Object Library						
9.00 am to 7pm	500	1920x1080	30	Outdoor	Yellow, White	.jpg
Amrita Object Library (Fire with Fire Like Objects)						
All items	500	1920x1080	30	Outdoor	Yellow	.jpg
Youtube Videos (Building)						
Downloaded Images						
Different resolution				All Kinds	All Color	.jpg

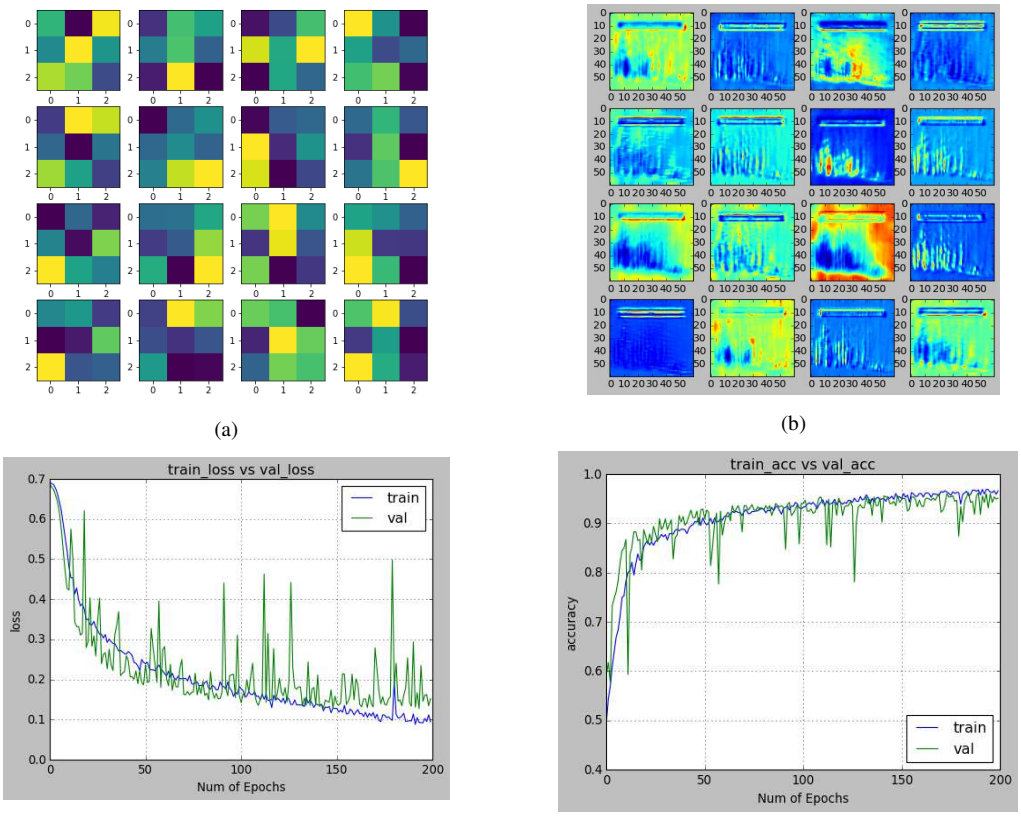


Fig.7 (a-b) 16 color mapping for 16weight filter of convolution layer three and corresponding feature map (c) compare training loss and validation loss vs. number of epochs of the model (d) compare training accuracy and validation accuracy vs. number of epochs of the model

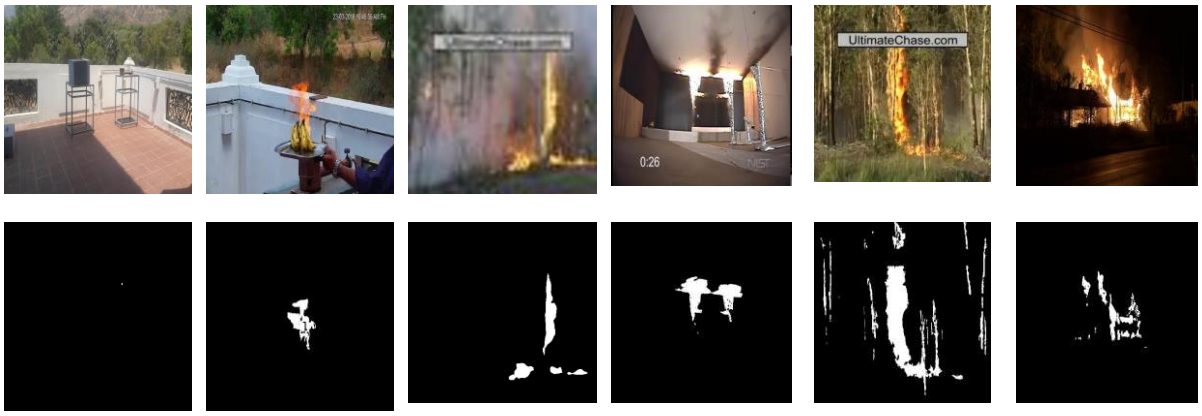


Fig.8 Test images of the dataset and its corresponding localized fire region using the proposed approach

(a) (b) (c) (d) (e) (f)

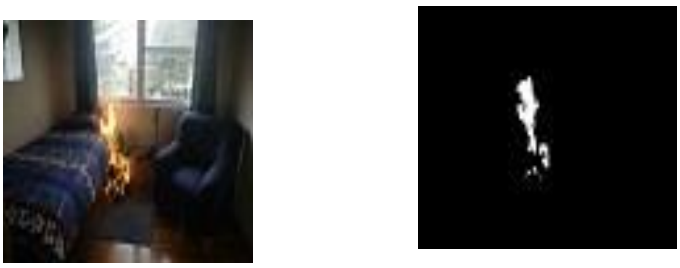
Fig.9 Localization of fire frames for challenging dataset (a) Flipped frame (b) noisy Gaussian frame (c) Fire and Fire-like object frames (d) illumination changes (e) fire-color like cloth (f) fire-color like light

Table 3

Comparison of different fire detection and localization approach for above mentioned dataset

Approach	False positives	False Negatives	Accuracy	Precision	Recall	F1-measure
Celik et al.[21]	25.00	0.00	90.00	85.71	100	92.31
Kwang-Ho Cheong et al. [24]	17.86	1.32	92.20	89.55	98.68	93.90
Khatami et al.[26]	27.50	4.67	86.20	83.87	95.33	89.24
Khatami et al.[27]	34.50	7.67	81.60	80.06	92.33	85.76
Khan Muhammad et al.[32]	5.0	0.05	95.00	98.70	95.00	96.82
Proposed CNN approach	11.50	1.00	94.80	92.81	99.00	95.81

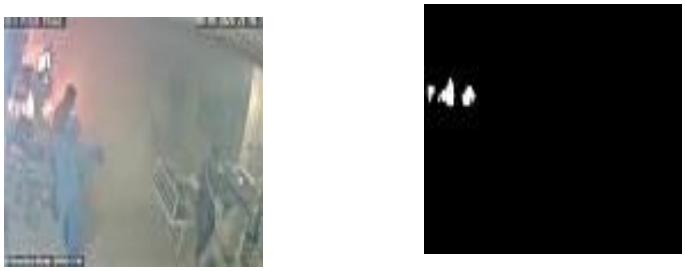
Home



Electrical Fire

Car Parking

Hospital



School

Fig.10 Fire detection and Localization in various types of building environments

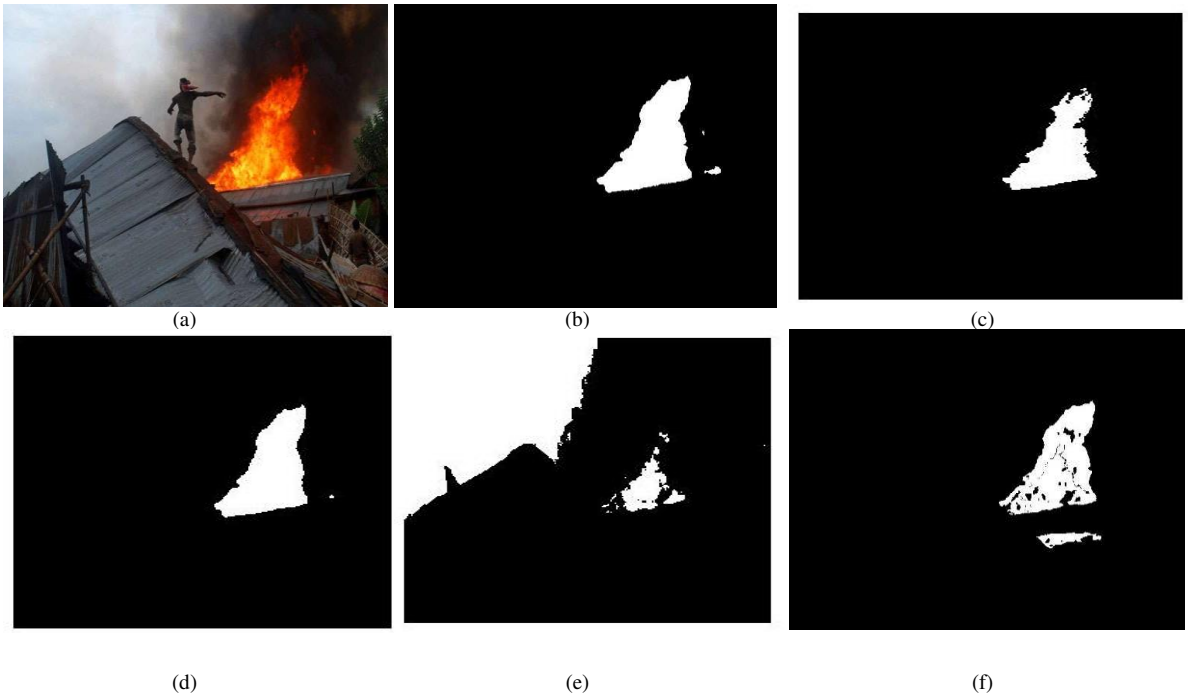


Fig.11 Proposed fire region localization approach compared with other systems with false alarms (a) original frame (b) proposed method(c) Celik (d) Khatami(2015a) (e) Khatami (2015b) (f) Kwang-Ho Cheong

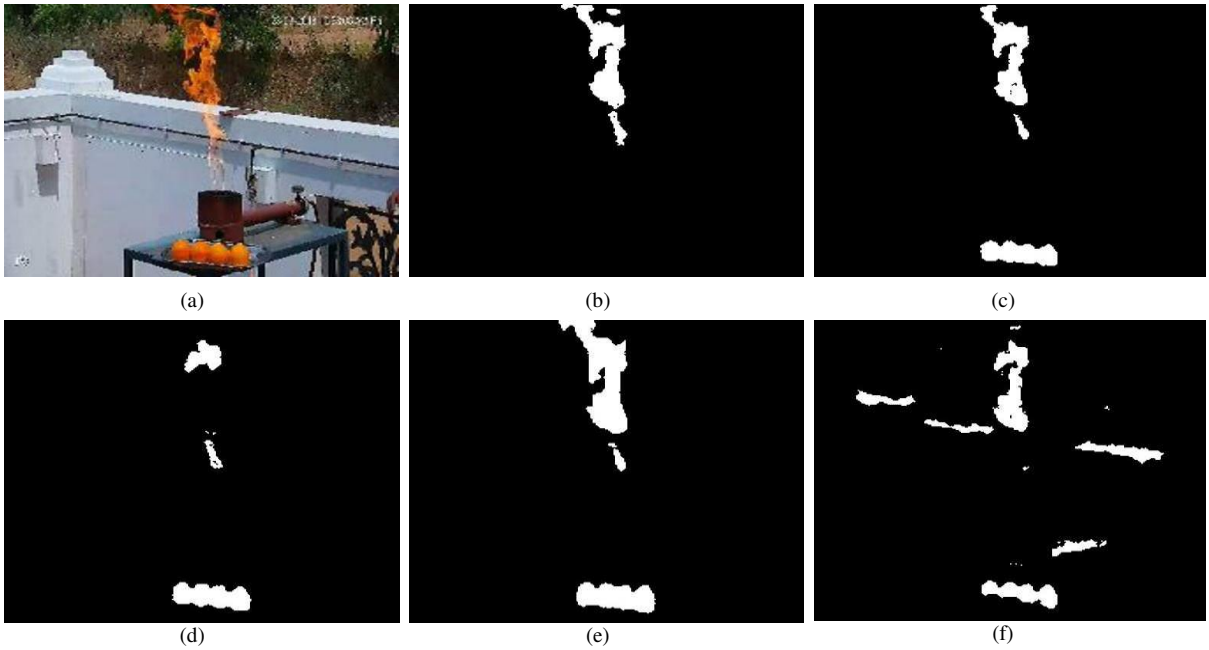
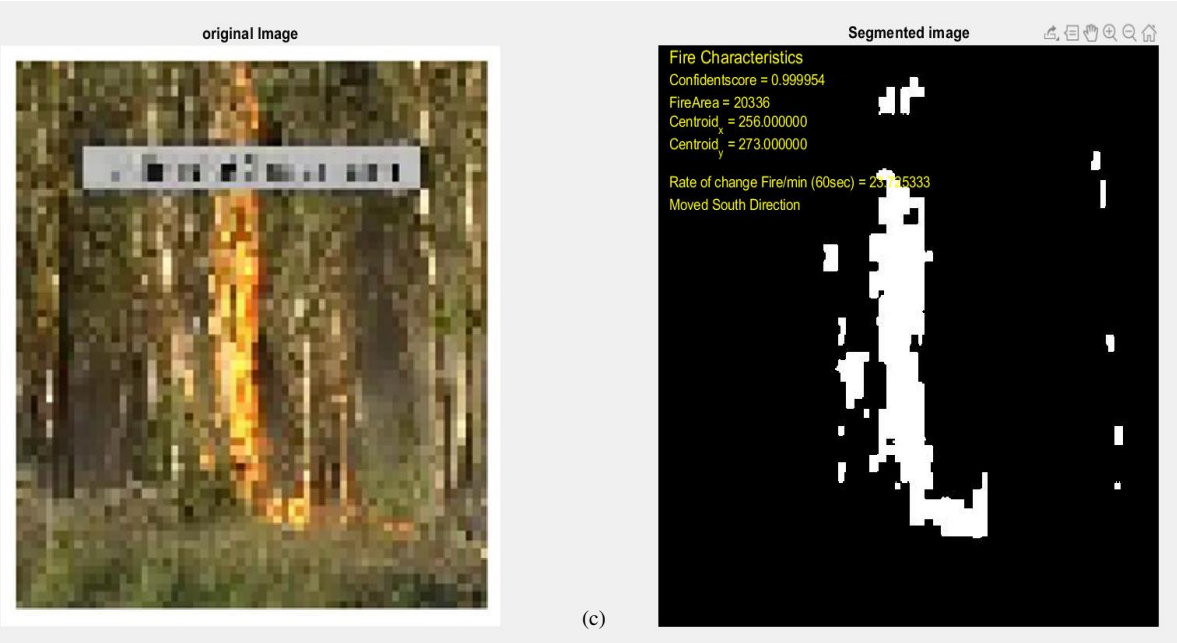


Fig.12 Localized the fire region with fire-like objects using the proposed CNN approach and other approaches (a) original image (b) proposed method (c) Celik (d) Khatami (2015a) (e) Khatami (2015b) (f) Kwang-Ho Cheong

In the convolution layer 3 of the proposed model, the weight filters and the feature maps are shown in Fig.7 (a-b). We identified feature map 12th in the convolution layer 3, which was very sensitive to the fire region for the entire dataset. During the training, we checked the loss and accuracy of our model for each epoch. The training and validation performance graph is shown in Fig.7 (c-d). The purpose of the checking is to determine whether our model fits in the right way.

The training loss and validation loss are fairly approximately fit, showing our proposed model fits properly for fire detection. Fig.8 shows the simulation output representative frame from Dataset. The tested frames contain different challenges such as Color, indoor and outdoor, the scale of the fire, and various illumination. Fig.9 shows the Localization of the fire region in other sets of challenging frames. This set of representative frames contains flipped frames, Gaussian noise, fire-like objects, and lights. This type of challenging dataset evaluates the performance of the fire region localization. Fig.10 shows the Localization of fire regions in various building environments. We have developed a hybrid method for fire detection on images or videos. In this work, the proposed lightweight CNN detects fire and is sensitive to fire pixels in 3rd layer. However, the layer is not sensitive to all images due to low-resolution frames. The Gaussian probability threshold method improves the fire pixel localization.

(a) (b)



(c)

Fig.13 Fire region characteristics of 61st frame of mivia dataset (a) original image (b) Feature map of convolution layer 2 (c)Fire region characteristics of localized fire region

We have selected five related approaches to compare our simulation results with conventional and recent camera vision approaches for our challenging dataset. The selection is based on their works and complexity. The proposed work is compared with selected standard approaches, and the performance parameter was tabulated in Table 3. The Celik et al. [21] method yielded high detection accuracy and zero false negatives. However, the false-positive rate is 25%, clearly shown in Fig.12 (c). Kwang-Ho Cheong et al. [24] proposed algorithm is best for detection accuracy and 1.32% false negatives.

However, the false-positive rate is still high in this approach. Khatami et al. [26], [27] are the most straightforward image processing algorithms for localized fire regions. The algorithm was created a fire color space using a weighting filter. Even though the algorithm detected a narrow range of fire colors. The detection algorithm is very sensitive to the fire-like color region and bright region. Thus, the approach yields a high false alarm rate compared to all other selected methods. The simulation results of this method are shown in Fig.11 (e) and Fig.12 (d), which is the evidence for the above statement.

Finally, we compared our results with K. Muhammad et al. [32]. Table 3 shows that K. Muhammad's works give high detection accuracy and low false alarm rate among all other works. Our light CNN model approach provides nearby performance for this model with low computation cost. K. Muhammad et al. [32] method only detects the fire probability and does not localize the fire region. The proposed CNN architecture is modified from conventional CNN. The input image resolution is very low, 64x64. The proposed model can localize the fire region at very low-resolution frames also. The weight file size is approximately (502KB). These advantages of our model will apply to hardware implementation for real-time applications. Furthermore, the proposed work has high detection accuracy and a very low false alarm compared with other approaches. This can be seen in Fig.11 (b) and Fig.12 (b). The proposed work detects fire region characteristics like confidence score, area, centroid, and direction using Region Prop in MATLAB. These fire characteristics provide the time analysis of the fire region for preserving living [42]. This can be shown in Fig.13

V. Conclusion

Vision-sensor-based fire region localization was presented. The algorithm applies a deep learning framework based on light CNN architecture. The evaluation parameters and simulation results interpret the high detection accuracy and low false alarm. The proposed method detects the fire region and provides the fire region characteristics, which creates a new direction for fire detection automation. The model and weight file size are much lesser than any other standard model, which is very helpful for embedding capabilities in real-time applications.

Acknowledgments

The proposed method is the part of the project work entitled "A framework for event modeling and detection for Smart Buildings using Vision Systems."The work is supported by DST (DST/TWF Division/AFW for EM/C/2017/121)

Compliance with Ethical standards:

1. All Authors state that there is no conflict of interest.
2. Humans and animals are not involved in this research work.
3. We used our data.

REFERENCES

[1] S. G. Kong, D. Jin, S. Li, and H. Kim, "Fast fire flame detection in surveillance video using logistic regression and temporal smoothing," *Fire Saf. J.*, vol. 79, pp. 37–43, Jan. 2016.

[2] T. H. Chen, P. H. Wu, and Y. C. Chiou, "An early fire-detection method based on image processing," in *Proceedings - International Conference on Image Processing, ICIP*, 2004, vol. 3, pp. 1707–1710.

[3] W. B. Horng, J. W. Peng, and C. Y. Chen, "A new image-based real-time flame detection method using color analysis," in *2005 IEEE Networking, Sensing and Control, ICNSC2005 - Proceedings*, 2005, vol. 2005, pp. 100–105.

[4] T. Celik, H. Demirel, H. Ozkaramanli, and M. Uyguroglu, "Fire detection using statistical color model in video sequences," *J. Vis. Commun. Image Represent.*, vol. 18, no. 2, pp. 176–185, Apr. 2007.

[5] M. Mueller, P. Karasev, I. Kolesov, and A. Tannenbaum, "Optical flow estimation for flame detection in videos," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2786–2797, 2013.

[6] K. Dimitropoulos, P. Barmpoutis, and N. Grammalidis, "Spatio-temporal flame modeling and dynamic texture analysis for automatic video-based fire detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 2, pp. 339–351, 2015.

[7] T. Toulouse, L. Rossi, M. Akhloufi, T. Celik, and X. Maldague, "Benchmarking of wildland fire colour segmentation algorithms; Benchmarking of wildland fire colour segmentation algorithms," *IET*, vol. 9, no. 12, pp. 1064–1072, Dec. 2015.

[8] R. Chi, Z.-M. Lu, and Q.-G. Ji, "IET Image Processing Real-time multi-feature based fire flame detection in video," *Wiley Online Libr.*, vol. 11, no. 1, pp. 31–37, Jan. 2016.

[9] P. Sridhar, S. K. Thangavel, and L. Parameswaran, "A New Approach for Fire Pixel Detection in Building

Environment Using Vision Sensor," in *Advances in Intelligent Systems and Computing*, 2021, vol. 1189, pp. 392–400.

[10] P. Sridhar, L. Parameswaran, and S. K. Thangavel, "An efficient rule based algorithm for fire detection on real time videos," *J. Comput. Theor. Nanosci.*, vol. 17, no. 1, pp. 308–315, 2020.

[11] M. Dua, M. Kumar, G. Singh Charan, and P. Sagar Ravi, "An Improved Approach for Fire Detection using Deep Learning Models," in *2020 International Conference on Industry 4.0 Technology, I4Tech 2020*, 2020, pp. 171–175.

[12] J. Gotthans, T. Gotthans, and R. Marsalek, "Deep convolutional neural network for fire detection," in *Proceedings of the 2020 30th International Conference Radioelektronika, RADIOELEKTRONIKA 2020*, 2020, pp. 1–6.

[13] A. J. Dunnings and T. P. Breckon, "Experimentally defined convolutional neural network architecture variants for non-temporal real-time fire detection," in *Proceedings - International Conference on Image Processing, ICIP*, 2018, pp. 1358–1362.

[14] G. S. Ca, N. Bhowmik, and T. P. Breckon, "Experimental exploration of compact convolutional neural network architectures for non-temporal real-time fire detection," in *Proceedings - 18th IEEE International Conference on Machine Learning and Applications, ICMLA 2019*, Dec. 2019, pp. 653–658.

[15] S. Saponara, A. Elhanashi, and A. Gagliardi, "Exploiting R-CNN for video smoke/fire sensing in antifire surveillance indoor and outdoor systems for smart cities," in *Proceedings - 2020 IEEE International Conference on Smart Computing, SMARTCOMP 2020*, 2020, pp. 392–397.

[16] K. T. Abraham, M. Ashwin, D. Sundar, T. Ashoor, and G. Jeyakumar, "An evolutionary computing approach for solving key frame extraction problem in video analytics," in *Proceedings of the 2017 IEEE International Conference on Communication and Signal Processing, ICCSP 2017*, Feb. 2018, vol. 2018-Janua, pp. 1615–1619.

[17] K. S. Gautam and S. K. Thangavel, "Video analytics-based intelligent surveillance system for smart buildings," *Soft Comput.*, vol. 23, no. 8, pp. 2813–2837, Apr. 2019.

[18] K. S. Gautam and S. K. Thangavel, "Hidden object detection for classification of threat," in *International Journal of Computer Aided Engineering and Technology*, 2020, vol. 13, no. 1–2, pp. 217–238.

[19] K. S. Gautam and S. K. Thangavel, "Video analytics-based facial emotion recognition system for smart buildings," *Int. J. Comput. Appl.*, pp. 1–10, 2019.

[20] R. A. Sowah, A. R. Ofoli, S. N. Krakani, and S. Y. Fiawoo, "Hardware design and web-based communication modules of a real-time multisensor fire detection and notification system using fuzzy logic," *IEEE Trans. Ind. Appl.*, vol. 53, no. 1, pp. 559–566, 2017.

[21] T. Çelik and H. Demirel, "Fire detection in video sequences using a generic color model," *Fire Saf. J.*, vol. 44, no. 2, pp. 147–158, 2009.

[22] G. Marbach, M. Loepfe, and T. Brupbacher, "An image processing technique for fire detection in video images," *Fire Saf. J.*, vol. 41, no. 4, pp. 285–289, 2006.

[23] A. Khalil, S. U. Rahman, F. Alam, I. Ahmad, and I. Khalil, "Fire Detection Using Multi Color Space and Background Modeling," *Fire Technol.*, vol. 57, pp. 1221–1239, May 2021.

[24] B. C. Ko, K. H. Cheong, and J. Y. Nam, "Fire detection based on vision sensor and support vector machines," *Fire Saf. J.*, vol. 44, no. 3, pp. 322–329, 2009.

[25] W. Phillips, M. Shah, and N. Da Vitoria Lobo, "Flame recognition in video," *Proc. IEEE Work. Appl. Comput. Vis.*, pp. 224–229, 2000.

[26] A. Khatami, S. Mirghasemi, A. Khosravi, C. P. Lim, and S. Nahavandi, "A new PSO-based approach to fire flame detection using K-Medoids clustering," *Expert Syst. Appl.*, vol. 68, pp. 69–80, 2017.

[27] A. Khatami, S. Mirghasemi, A. Khosravi, and S. Nahavandi, "An efficient hybrid algorithm for fire flame detection," in *Proceedings of the International Joint Conference on Neural Networks*, 2015, vol. 2015-Septe, pp. 1–6.

[28] T. Qiu, Y. Yan, and G. Lu, "An autoadaptive edge-detection algorithm for flame and fire image processing," in *IEEE Transactions on Instrumentation and Measurement*, 2012, vol. 61, no. 5, pp. 1486–1493.

[29] Q. Zhang, J. Xu, L. Xu, and H. Guo, "Deep Convolutional Neural Networks for Forest Fire Detection," 2016.

[30] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, Nov. 2014, pp. 675–678.

[31] K. Muhammad, J. Ahmad, and S. W. Baik, "Early fire detection using convolutional neural networks during surveillance for effective disaster management," *Neurocomputing*, vol. 288, pp. 30–42, 2018.

[32] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, "Convolutional Neural Networks Based Fire Detection in Surveillance Videos," *IEEE Access*, vol. 6, pp. 18174–18183, 2018.

[33] K. Muhammad, J. Ahmad, Z. Lv, P. Bellavista, P. Yang, and S. W. Baik, "Efficient Deep CNN-Based Fire

Detection and Localization in Video Surveillance Applications," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 49, no. 7, pp. 1419–1434, 2019.

[34] A. Jadon, M. Omama, A. Varshney, M. S. Ansari, and R. Sharma, "FireNet: A specialized lightweight fire And smoke detection model for real-time iot applications," *arXiv*. 2019.

[35] M. M. Kabir, A. Q. Ohi, M. S. Rahman, and M. F. Mridha, "An Evolution of CNN object classifiers on low-resolution images," in *HONET 2020 - IEEE 17th International Conference on Smart Communities: Improving Quality of Life using ICT, IoT and AI*, Dec. 2020, pp. 209–213.

[36] Y. V. Venkatesh, A. K. Kassim, and O. V. R. Murthy, "Resampling approach to facial expression recognition using 3D meshes," in *Proceedings - International Conference on Pattern Recognition*, 2010, pp. 3772–3775.

[37] "AOL DATASET:" https://intranet.cb.amrita.edu/sslab/imagedatasets/index.html.

[38] "NIST DATASET." https://www.nist.gov/video-category/fire?page=0.

[39] "BILKENT DATASET:" http://signal.ee.bilkent.edu.tr/VisiFire/Demo/FireClips/.

[40] "mivia DATASET:" https://mivia.unisa.it/datasets/video-analysis-datasets/fire-detection-dataset/.

[41] 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 Trans. CIRCUITS Syst. VIDEO Technol.*, vol. 25, no. 9, 2015.

[42] A. Dhall, O. V. Ramana Murthy, R. Goecke, J. Joshi, and T. Gedeon, "Video and image-based Emotion recognition challenges in the wild: EmotiW 2015," in *ICMI 2015 - Proceedings of the 2015 ACM International Conference on Multimodal Interaction*, Nov. 2015, pp. 423–426.