International Journal of Technology Engineering Arts Mathematics Science
Vol. 2, No. 1, June 2022, pp. 18-28
ISSN: 2583-1224

Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Application

Ashutosh Kulkarni [1], Onkar Gaikwad [2], Priyank Virkar [3]

[1]AISSMS Institute of Information Technology, Maharashtra, India.

[2]AISSMS Institute of Information Technology, Maharashtra, India.

[3]AISSMS Institute of Information Technology, Maharashtra, India.

Corresponding Author: Ashutosh Kulkarni (ashutoshkulkarniak47@gmail.com)

Article Information

Article history:

Received May 13, 2022
Revised Jun 13, 2022
Accepted Jun 15, 2022

ABSTRACT

Fire is a devastating natural disaster that affects both people and the environment. Recent research has suggested that computer vision could be used to construct a cost-effective automatic fire detection system. This paper describes a unique framework for utilizing CNN to detect fire. Convolution Neural Networks have yielded state-of-art performance in image classification and other computer vision tasks. Their use in fire detection systems will significantly enhance detection accuracy, resulting in fewer fire disasters and less ecological and social consequences. The deployment of CNN-based fire detection in everyday surveillance networks, however, is a severe problem due to the huge memory and processing needs for inference. In this study, we offer an innovative, energy-efficient, and computationally effective CNN model for detection of fire, localisation, and understanding of the fire scenario, based on the SqueezeNet architecture. It makes use of small convolutional kernels and avoids thick, fully connected layers that reduces the computational load. This paper shows how the unique qualities of the problem at hand, as well as a wide range of fire data, can be combined to make a balance of fire detection effectiveness and precision.

KEYWORDS: Convolution Neural Network (CNNs), Deep Learning, Fire detection, Fire Localization, Image classification, Surveillance Network.

1. INTRODUCTION

Recently, a variety of sensors have been introduced for different applications, including setting off a fire alarm, detecting vehicle obstacles, viewing the interior of the human body for diagnosis, monitoring animals and ships, and surveillance. Surveillance has drawn the most attention from academics because of cameras' better embedded processing capabilities. Many unusual events, such as road accidents, fires, emergency situations, and so on, can be detected early and the proper authorities can be notified autonomously utilizing smart monitoring systems.

Rate of forest fires reports have increased yearly due to human causes and dry climate. To avoid the terrible disaster of fire, many detection techniques have been widely studied to apply in practice. Most traditional methods are based on sensors due to its low-cost and simple installation. These systems are not applicable for using outdoor where energy of flame affected by fire materials and the burning process affected by environment that have potential cause of false alarms. Visual-based approach of image or video processing was shown to be a more reliable method to detect the fire since the closed circuit television (CCTV) surveillance systems are now available at many public places, and can help capture the fire scenes. In order to detect fire from scenes of color-videos, various schemes have been studied mainly focusing on the combination of static and dynamic characteristics of fire such as color information, texture and motion orientation, etc.

Two broad categories of approach can be identified for fire detection: 1) traditional fire alarms and 2) vision sensor assisted fire detection. Traditional

fire alarm systems are based on sensors that require close proximity for activation, such as infrared and optical sensors. These sensors are not well suited to critical environments and need human involvement to confirm a fire in the case of an alarm, involving a visit to the location of the fire. Furthermore, such systems cannot usually provide information about the size, location, and burning degree of the fire. To overcome these limitations, many visual sensor-based technologies have been investigated by researchers in this field to address these restrictions; they have the benefits of less human intervention, faster reaction, lower cost, and a greater surveillance coverage. In addition, such systems can confirm a fire without requiring a visit to the fire's location, and can provide detailed information about the fire including its location, size, degree, etc. Despite these advantages, there are still some issues with these systems, e.g., the complexity of the scenes under observation, irregular lighting, and low-quality frames; researchers have made several efforts to address these aspects, taking into consideration both color and motion features.

Recently, Foggia et al. proposed a real-time fire detection algorithm based on color, shape, and motion features, combined in a multi-expert system. The accuracy of this approach is higher than that of other methods; however, the number of false alarms is still high, and the accuracy of fire detection can be further improved. A survey of the existing literature shows that computationally expensive methods have better accuracy, and simpler methods compromise on accuracy and the rate of false positives. Hence, there is a need to find a better tradeoff between these metrics for several application scenarios of practical interest, for which existing computationally expensive methods do not fit well. We examine convolutional neural networks (CNN)-based deep features for early fire detection in surveillance networks to overcome the above issues.

1.1. PROPOSED FRAMEWORK

We examine deep neural networks for potential fire detection at initial stages of surveillance in the proposed system. We examine different deep CNNs for the objective problem, taking into account accuracy, embedded processing capacity of cctv systems, and the rate of false alarms.

Object recognition and localization], picture segmentation, super-resolution, classification, and indexing and retrieval] are just a few of the computer vision issues and applications where CNNs have demonstrated promising results. This broad success can be attributed to their hierarchical system, which learns very powerful features from raw data automatically. Three well-known processing layers make up a typical CNN architecture.

1) When multiple kernels are used to the input data, numerous feature maps are formed as a convolution layer.

2) A pooling layer that selects maximal activation from a limited neighborhood of feature maps acquired from the preceding convolution layer; the purpose of this layer is to create translation invariance and dimensionality reduction to some extent.

3) A completely linked layer that creates a global representation of high-level information from input data. Following a series of convolutional and pooling layers, this layer produces high-level features of the input data.

These layers are organized in a hierarchical architecture, with one layer's output serving as the input for the next. The weights of all neurons in convolutional kernels and fully connected layers are modified and learnt throughout the training phase. These weights can execute the goal classification by modeling the typical qualities of the input training data.

Pre - processing relates to all the modifications performed on the raw data before it has been given to the deep learning or machine learning algorithm in our project. For example, training a CNN model on raw photos will almost certainly result in poor classification results. CNN is a neural network that extracts input image features and another neural network classifies the image features. A feature extraction network uses the input image. The neural network uses the feature extraction signals for classification.

2.2 ALGORITHMS:

Algorithm 1 Feature Map Selection Algorithm for Localization.

Input: Training samples (TS), ground truth (GT), and the proposed deep CNN model (CNN-M)

1. Forward propagate TS through CNN-M
2. Select the feature maps FN from layer L of CNN-M
3. Resize GT and FN to 256×256 pixels
4. Compute mean activations map FMAi for FN

Binarize each feature map Fi as follows

5.

$$F(x,y)_{bin(i)} = \begin{cases} 1, & F(x,y)_i > F_{MA(i)} \\ 0, & Otherwise \end{cases}$$

6. Calculate the hamming distance HDi between GT and each feature map Fbin (i)
7. Calculate the sum of all resultant hamming distances, and shortlist the minimum hamming distances using threshold T

Select appropriate feature maps according to the shortlisted hamming distances **Algorithm 2** Fire Localization Algorithm

Input: Image I of the video sequence and the proposed deep CNN model (CNN-M)

1. Select a frame from the video sequence and forward propagate it through CNN-M
2. **IF** predicted label = non-fire **THEN**

No action

ELSE

1. Extract feature maps 8, 26, and 32 (F8, F26, F32) from the "Fire2/Concat" layer of CNN-M
2. Calculate mean activations map (FMA) for F8, F26, and F32
3. Apply binarization on FMA through threshold T as follows:

$$F_{Localize} = \begin{Bmatrix} 1, & F_{MA} > T \\ 0, & Otherwise \end{Bmatrix}$$

4. Segment fire regions from FMA

Output: Binary image with segmented fire Ilocalize

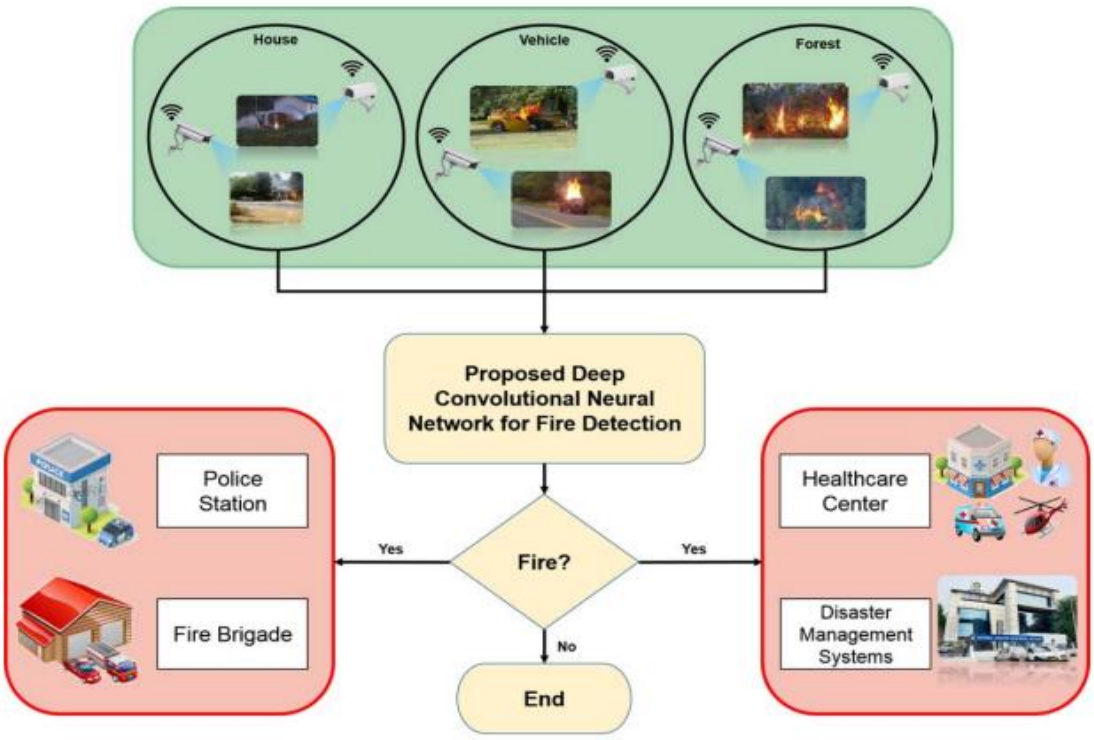


Fig. 1. Overview of the proposed system for fire detection using a deep CNN.

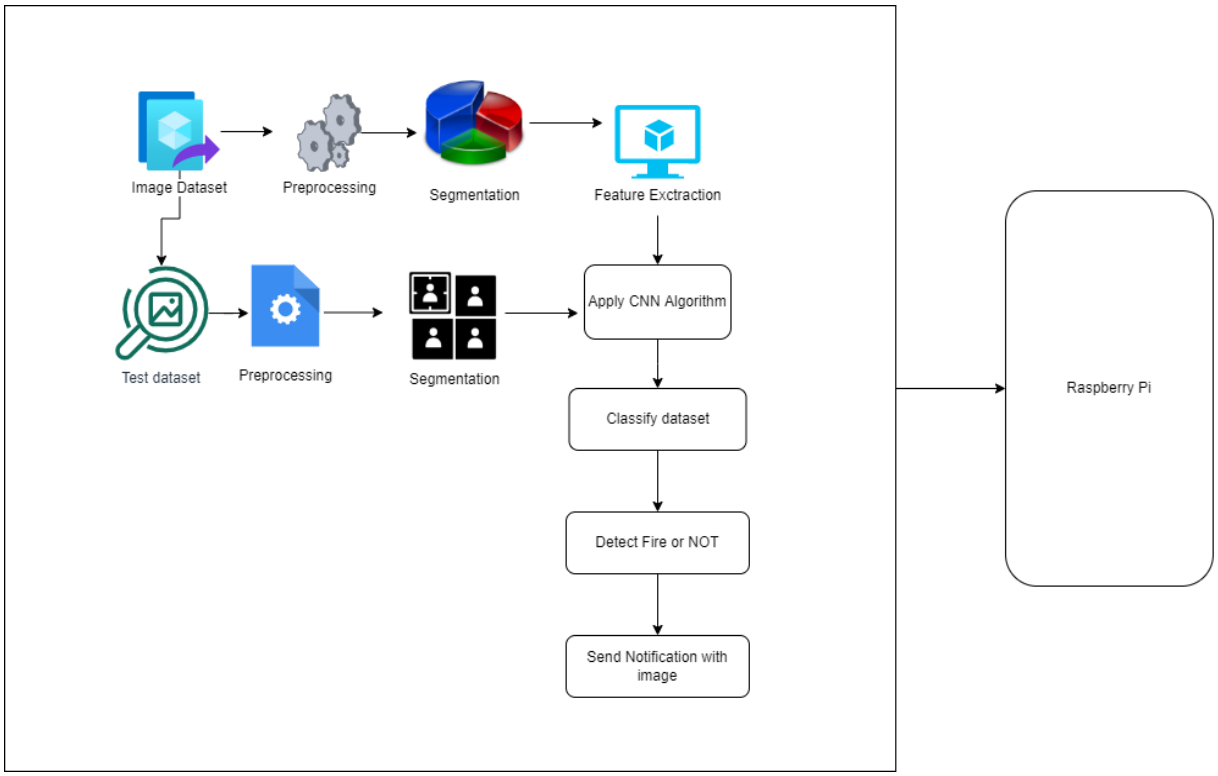


Fig 2. System Architecture.

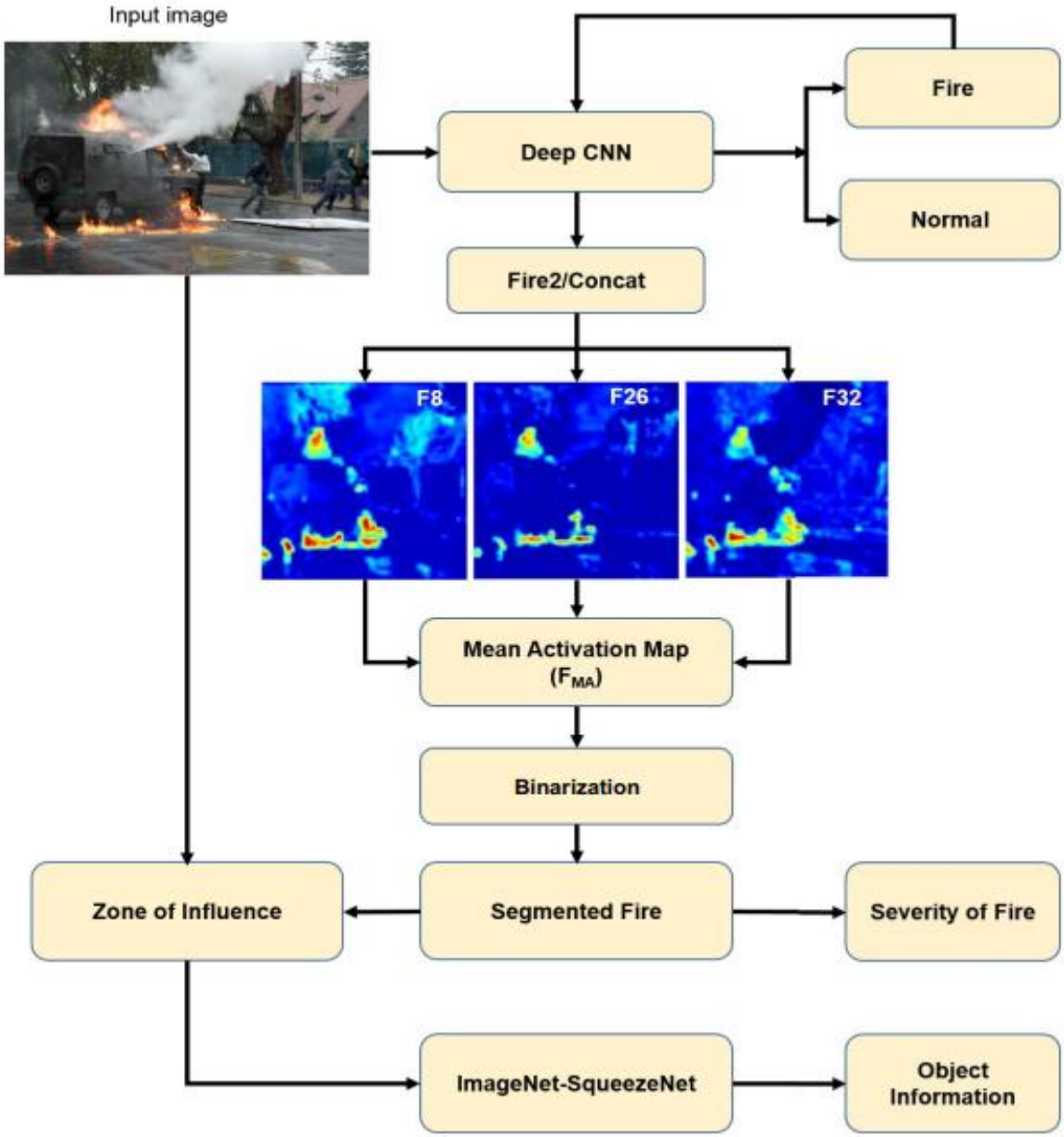


Fig. 3. Fire localization using the proposed deep CNN.

8. :

Output: Feature maps sensitive to fire.

2. METHODOLOGY

We use a model with a similar architecture to SqueezeNet that has been modified to fit our target problem. The original model was trained on the ImageNet dataset and is capable of classifying 1000 different objects. This was achieved by reducing the number of neurons in the final layer from 1000 to 2. By keeping the rest of the architecture similar to the original, we aimed to reuse the parameters to solve the fire detection problem more effectively. Two standard convolutional layers, three maximum pooling layers, one average pooling layer, and eight "fire modules" make up the model.

In the first convolution layer, the input picture is passed through 64 filters of size 3x3, resulting in 64 feature maps. The first max pooling layer, with a stride of two pixels and a neighborhood of 3x3 pixels, selects the maximum activations of these 64 features maps. This decreases the proportions of the feature maps by a factor of two, allowing the most valuable information to be retained while the insignificant particulars are discarded. Following that, we employ two 128-filter fire modules, followed by a 256-filter fire module. Squeezing and expansion are two more convolutions in every firing module. Because each

module has several filter resolutions and the Caffe framework lacks native support for such convolution layers. In each fire module, an expansion layer was added, along with two independent convolution layers. 11 filters make up the first convolution layer, while 33 filters make up the second. In the channel dimension, the output of these two layers is concatenated.

A significant number of weights need to be properly adjusted in CNNs, and a huge amount of training data is usually required for this. Insufficient training data can lead to overfitting of these parameters. The fully connected layers usually contain the most parameters, and these can cause significant overfitting. These problems can be avoided by introducing regularization layers such as dropout, or by replacing dense fully connected layers with convolution layers. We used a pretrained SqueezeNet model and fine-tuned it according to our classification problem with a slower learning rate of 0.001. We also removed the last fully connected layers to make the architecture as efficient as possible in terms of classification accuracy. The process of fine-tuning was executed for 10 epochs; this increased the classification accuracy from 89.8% to 94.50%, thus giving an improvement of 5%.

3. RESULTS AND DISCUSSION

This section describes the tests that were conducted to verify the effectiveness of our method. Beginning with the experimental specifics, we go through the configuration and datasets that were used in the research. The experimental findings for several fire datasets are then provided, followed by a relation to existing fire detection and localisation algorithms. Finally, we discuss tests that demonstrate our method's superiority in terms of resilience. Throughout the experiments, we refer to our method as "CNNFire."

4.1. EXPERIMENTS ON DATASET 1

Our experiments for testing the performance of the proposed framework are mainly based on two datasets: 1) Foggia et al. (Dataset1) and Chino et al. (Dataset2).In the appropriate sections, the reasons for utilizing each of these datasets are explained.Dataset1 contains a total of 31 videos captured in different environments. 14 of the videos feature a fire, while the remaining 17 are normal videos. This dataset is particularly ideal for these investigations due to a number of issues, including its bigger size when compared to other existing datasets. For example, some of the normal videos include fire-like objects; this makes fire detection more challenging, and hence fire detection methods using color features may wrongly classify these frames.

Technique	False Positives	False Negatives	Accuracy
Proposed after FT	**8.87%**	2.12%	**94.50%**
Proposed before FT	9.99%	10.39%	89.80%
AlexNet after FT	9.07%	2.13%	94.39%
AlexNet before FT	9.23%	10.64%	90.07%
Foggia et al.	11.67%	**0%**	93.55%
De Lascio et al.	13.33%	0%	92.86%
Habibuglu et al.	5.88%	14.29%	90.32%

Table 1. Comparison of various fire detection methods for dataset1

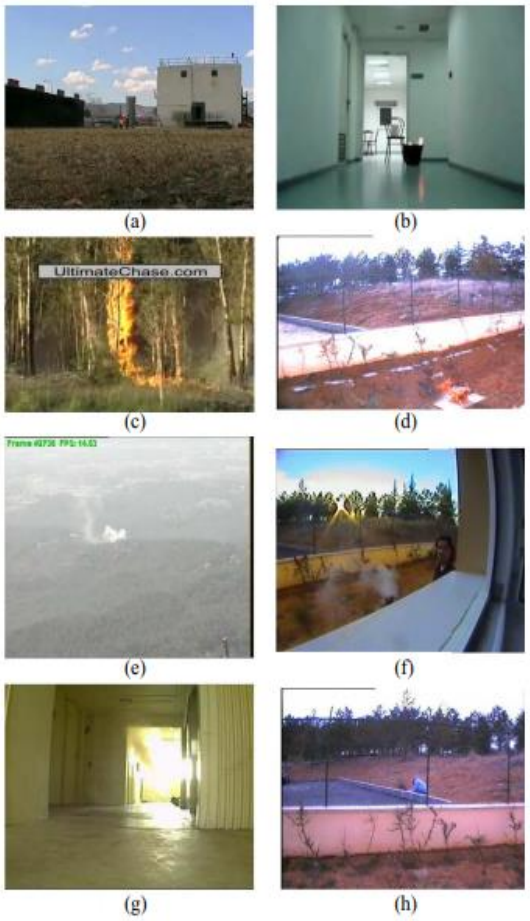


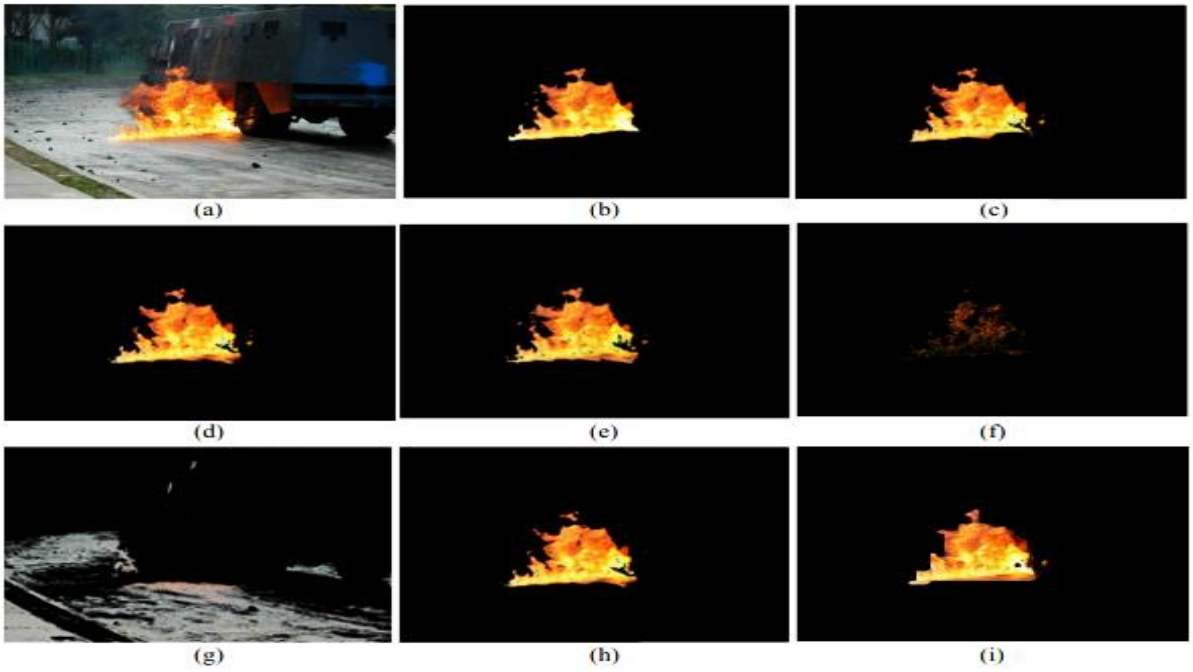
Fig.4. Set of representative images from Dataset1. The top four images are taken from videos of fires, while the remaining four images are from non-fire videos.

The gathered experimental data from Dataset1 are calculated and compared with comparable approaches in Table 1, which includes a set of sample photos from Dataset1.

Although the results of the proposed fine-tuned AlexNet are good compared to other existing methods, there are still certain limitations. First, the size of this model is comparatively large (approx. 238 MB), thereby restricting its implementation in CCTV

networks. Second, the rate of false alarms (false positives) is 9.07%, which is still high and would be problematic for fire brigades and disaster management teams.

For a comparison of our results with state-of-the-art methods for Dataset1, we selected a total of six related works. This selection was based on criteria including the features used in the related works, their year of publication, and the dataset under consideration. We then compared our method with the selected fire detection algorithms, as shown in Table 1. The selected works use various low-level features and different datasets, and their year of publication ranges from 2004 to 2015. The results show that Çelik and Demirel and Foggia et al. are the best algorithms in terms of false negatives. However, their results are not impressive in terms of the other metrics of false positives and accuracy. From the perspective of false positives, the algorithm of Habiboglu ˘ et al performs best, and dominates the other methods. However, its false negative rate is 14.29%, the worst result of all the methods examined. The accuracy of the four other methods is also better than this method, with the most recent method being the best.



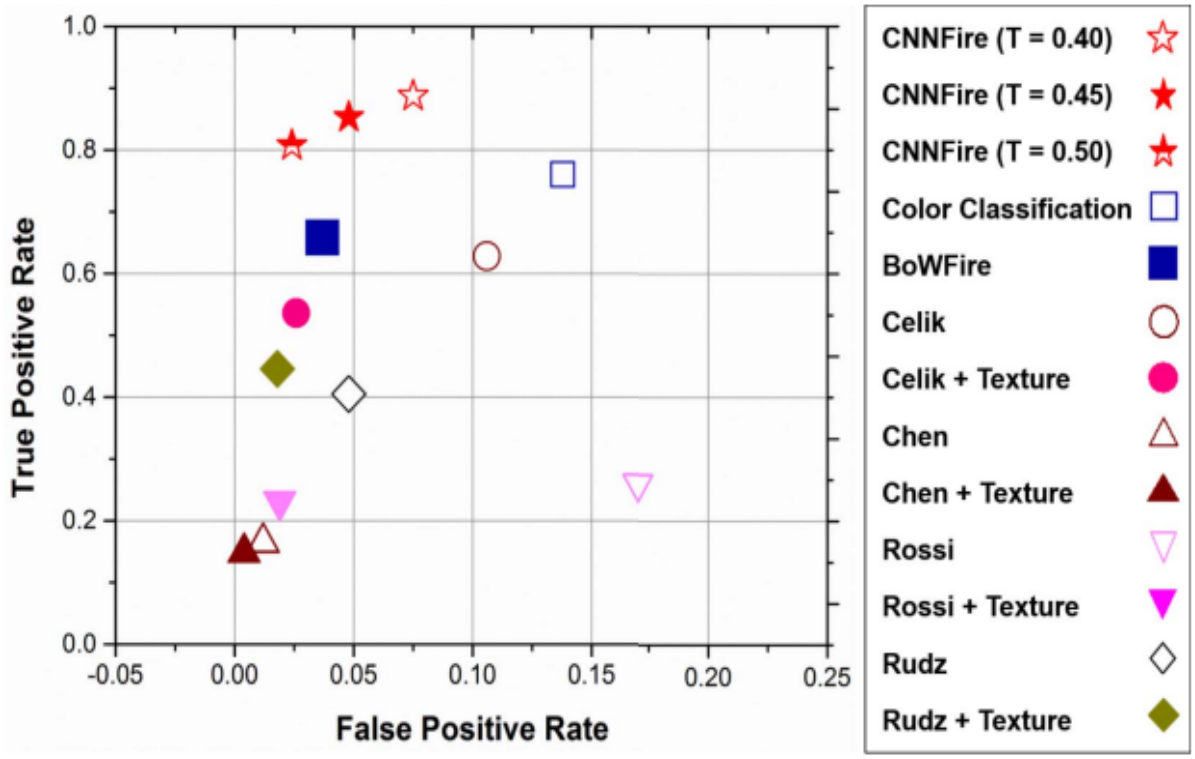


Fig. 5. Comparison of our CNNFire approach with other methods

Fig 6. Visual fire localization results of our CNNFire approach and other fire localization methods. (a) Input image (b) Ground truth. (c) BoWFire. (d) Color classification. (e) Celik. (f) Chen. (g) Rossi. (h) Rudz. (i) CNNFire.

4.2. EXPERIMENTS ON DATASET2

There are 226 photos in Dataset2, containing 119 fire pictures and 107 non-fire shots. The dataset, which was taken from, is modest but offers multiple challenges, including red and fire-colored items, fire-like sunshine, and fire-colored brightness in various structures. Table 2 shows the results derived from Dataset2 utilizing the proposed design. In terms of usefulness, dataset, and publication year, we compared our findings to four existing fire detection methods. To ensure a fair evaluation and a full overview of the performance of our approach, we considered another set of metrics (precision, recall, and F-measure as used by. In a similar way to the experiments on Dataset1, we tested Dataset2 using the fine-tuned AlexNet and our proposed fine-tuned SqueezeNet model.

Technique	**Precision**	**Recall**	**F-Measure**
Proposed method before FT	**0.86**	0.97	**0.91**
Proposed method after FT	0.84	0.87	0.85
AlexNet after FT	0.82	**0.98**	0.89
AlexNet before FT	0.85	0.92	0.88
Chino et al. (BoWFire)	0.51	0.65	0.57
Rudz et al	0.63	0.45	0.52
Rossi et al.	0.39	0.22	0.28
Celik et al.	0.55	0.54	0.54
Chen et al.	0.75	0.15	0.25

Table 2. Comparison of different fire detection methods for dataset2.

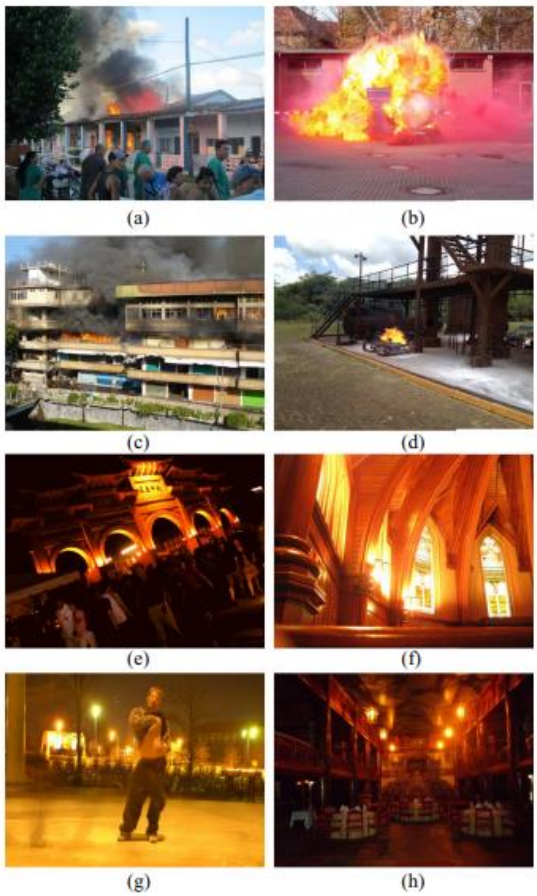


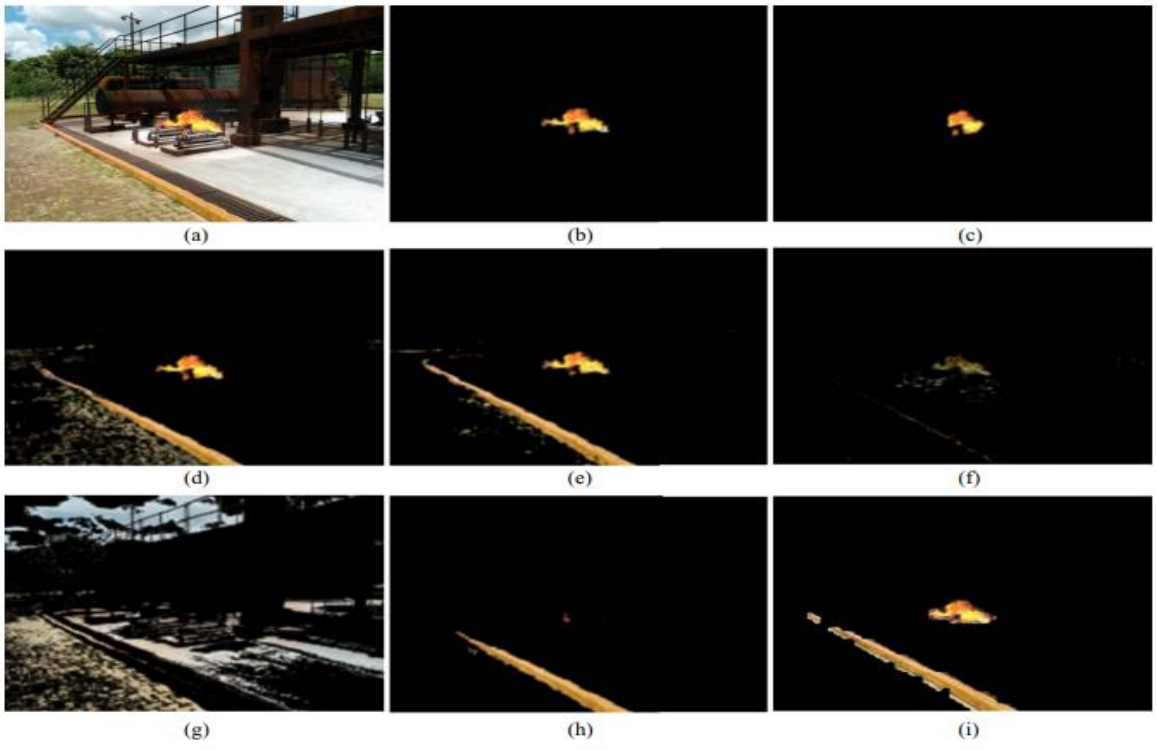
Fig. 7. Representative images from Dataset2. The top four images include fires, while the remaining four images represent fire-like normal images.

In a similar way to the experiments on Dataset1, we tested Dataset2 using the fine-tuned AlexNet and our proposed fine-tuned SqueezeNet model. For the fine-tuned AlexNet, an F-measure score of 0.89 was achieved. Further improvement was achieved using our model, increasing the F-measure score from 0.89 to 0.91 and the precision from 0.82 to 0.86. It is evident from Table III that our method achieved better results than the state-of-the-art methods, confirming the effectiveness of the proposed deep CNN framework.

Fig. 8. Fire localization results from our CNNFire and other schemes with false positives. (a) Input image. (b) Ground truth. (c) BoWFire. (d) Color classification. (e) Celik. (f) Chen. (g) Rossi. (h) Rudz. (i) CNNFire.

4.3. FIRE LOCALIZATION RESULTS AND DISCUSSION

The performance of our technique is evaluated in this part in terms of fire localisation and comprehension of the scene under observation. To examine the performance of fire localisation, true positive and false positive rates were calculated. Because the feature maps we utilized to locate fire were smaller than the ground truth photos, they were scaled to match the ground truth images' dimensions. A reason for choosing SqueezeNet was



the model's ability to provide bigger feature map sizes by using smaller kernels and avoiding pooling layers. When the feature maps were resized to match the ground truth images, this allowed us to execute a more accurate localization.

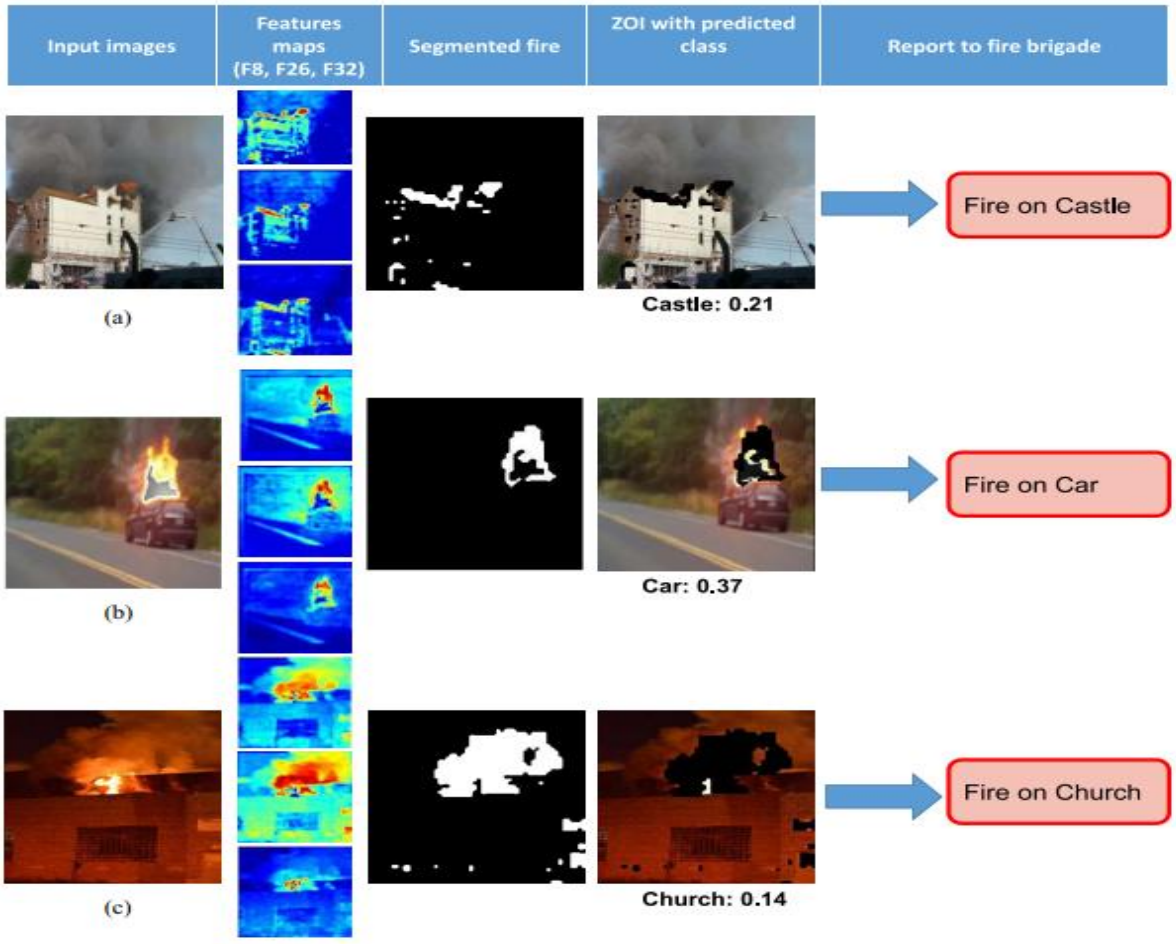
Fig. 5 shows the results of all methods for a sample image from Dataset2. The BoWFire, color categorization, Celik, and Rudz results are nearly identical. In this context, Rossi provides the worst outcomes, while Chen outperforms Rossi. The results from CNNFire are similar to the ground truth. Fig. 7. Highlights the performance of all methods for another sample image, with a higher probability of false positives. Although BoWFire has no false positives for this case, it misses some fire regions, as is evident from its result. Color classification and Celik detect the fire regions correctly, but give larger regions as false positives. Chen fails to detect the fire regions of the ground truth image.

Our system can assess the severity of the observed fire as well as the item under observation, in addition to fire detection and localisation. We retrieved the ZOI from the input image and segmented the fire regions for this. The ZOI image was then fed into the SqueezeNet model, which had been trained on the 1000-class ImageNet dataset. The SqueezeNet model's label for the ZOI image is then paired with the fire's severity for reporting to the fire department. A set of sample cases from this experiment is given in Fig. 8

Fig. 9. Sample outputs from our overall system: the first column shows input images with labels predicted by our CNN model and their probabilities, with the highest probability taken as the final class label; the second column shows three feature maps (F8, F26, and F32) selected by Algorithm 1; the third column highlights the results for each image using Algorithm 2; the fourth column shows the severity of the fire and ZOI images with a label assigned by the SqueezeNet model; and the final column shows the alert that should be sent to emergency services, such as the fire brigade. (a) Fire: 98.76%, normal: 1.24%. (b) Fire: 98.8%, normal: 1.2%. (c) Fire: 99.53%, normal: 0.47%.

4. CONCLUSION

Intelligent CCTV monitoring systems have emerged as a consequence of smart cameras' inherent processing capabilities. Various abnormal events such as accidents, medical emergencies, and fires can be detected using these smart cameras. Fire is the deadliest of all the abnormal events, as failing to manage it at an early stage can result in major disasters, causing human, environmental, and economic losses. We present a SqueezeNet framework based on CNN for detection of fire in CCTV surveillance networks, inspired by the immense potential of CNNs. Our suggested system uses fine-tuning and the SqueezeNet architecture to balance the precision of fire detection with the complexity of the model.



We run tests on two benchmark datasets to ensure that the proposed system is feasible for deployment in real-world CCTV networks. Given the CNN model's reasonable accuracy for fire detection and localization, its size, and the incidence of false alarms, the system can assist disaster management teams in quickly addressing fire disasters and preventing massive losses. SqueezeNet is one of the innovative CNN architectures that we uncovered while investigating the CNN design space. We expect that SqueezeNet will encourage readers to think about and explore the wide range of options in the creative potential of CNN architectures, and to do so more systematically.

REFERENCES

[1] B. C. Ko, K.-H. Cheong, and J.-Y. Nam, “Fire detection based on vision sensor and support vector machines,” Fire Safety J, 2009, Vol. 44, Iss.. 3, pp. 322–329,

[2] Majid, Saima, Fayadh Alenezi, Sarfaraz Masood, Musheer Ahmad, Emine Selda Gündüz, and Kemal Polat. "Attention based CNN model for fire detection and localization in real-world images." Expert Systems with Applications (2021), Vol. 189, p.116114.

[3] M. Aktas, A. Bayramcavus and T. Akgun, "Multiple Instance Learning for CNN Based Fire Detection and Localization," 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2019, pp. 1-8, doi: 10.1109/AVSS.2019.8909842.

[4] 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., Sep. 2015, Vol. 25, Iss.. 9, pp. 1545–1556.

[5] J. Zhang, H. Zhu, P. Wang and X. Ling, "ATT Squeeze U-Net: A Lightweight Network for Forest Fire Detection and Recognition," in IEEE Access, Vol. 9, pp. 10858-10870, 2021, doi: 10.1109/ACCESS.2021.3050628.

[6] Iandola, Forrest N., et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size." arXiv preprint arXiv:1602.07360, 2016.

[7] Lee, H.J., Ullah, I., Wan, W., Gao, Y. and Fang, Z., Real-time vehicle make and model recognition with the residual SqueezeNet architecture. Sensors, 2019, Vol. 19, Iss. 5, p.982.

[8] M. Mueller, P. Karasev, I. Kolesov, and A. Tannenbaum, “Optical flow estimation for flame detection in videos,” IEEE Transactions on Image Processing, 2013, Vol. 22, Iss. 7, pp. 2786– 2797.

[9] Wang, Zhicheng, Zhiheng Wang, Hongwei Zhang, and Xiaopeng Guo. "A novel fire detection approach based on CNN-SVM using tensorflow." In International conference on intelligent computing, 2017, pp. 682-693. Springer, Cham.