Received January 30, 2018, accepted March 3, 2018. Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2018.2812835

Convolutional Neural Networks Based Fire Detection in Surveillance Videos

KHAN MUHAMMAD[1], (Student Member, IEEE), JAMIL AHMAD[1], (Student Member, IEEE), IRFAN MEHMOOD[2], (Member, IEEE), SEUNGMIN RHO[3], (Member, IEEE), AND SUNG WOOK BAIK[1], (Member, IEEE)

[1]Intelligent Media Laboratory, Digital Contents Research Institute, Sejong University, Seoul 143-747, South Korea
[2]Department of Computer Science and Engineering, Sejong University, Seoul 143-747, South Korea
[3]Department of Media Software, Sungkyul University, Anyang 14097, South Korea

Corresponding author: Sung Wook Baik (sbaik@sejong.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No.2016R1A2B4011712).

ABSTRACT The recent advances in embedded processing have enabled the vision based systems to detect fire during surveillance using convolutional neural networks (CNNs). However, such methods generally need more computational time and memory, restricting its implementation in surveillance networks. In this research paper, we propose a cost-effective fire detection CNN architecture for surveillance videos. The model is inspired from GoogleNet architecture, considering its reasonable computational complexity and suitability for the intended problem compared to other computationally expensive networks such as AlexNet. To balance the efficiency and accuracy, the model is fine-tuned considering the nature of the target problem and fire data. Experimental results on benchmark fire datasets reveal the effectiveness of the proposed framework and validate its suitability for fire detection in CCTV surveillance systems compared to state-of-the-art methods.

INDEX TERMS Fire detection, image classification, real-world applications, deep learning, and CCTV video analysis.

I. INTRODUCTION

The increased embedded processing capabilities of smart devices have resulted in smarter surveillance, providing a number of useful applications in different domains such as e-health, autonomous driving, and event monitoring [1]. During surveillance, different abnormal events can occur such as fire, accidents, disaster, medical emergency, fight, and flood about which getting early information is important. This can greatly minimize the chances of big disasters and can control an abnormal event on time with comparatively minimum possible loss. Among such abnormal events, fire is one of the commonly happening events, whose detection at early stages during surveillance can avoid home fires and fire disasters [2]. Besides other fatal factors of home fires, physical disability is the secondly ranked factor which affected 15% of the home fire victims [3]. According to NFPA report 2015, a total of 1345500 fires occurred in only US, resulted in $14.3 billion loss, 15700 civilian fire injuries, and 3280 civilian fire fatalities. In addition, a civilian fire injury and death occurred every 33.5 minutes and 160 minutes, respectively. Among the fire deaths, 78% occurred only due to home fires [4]. One of the main reasons is the delayed escape for disabled people as the traditional fire alarming systems need strong fires or close proximity, failing to generate an alarm on time for such people. This necessitates the existence of effective fire alarming systems for surveillance. To date, most of the fire alarming systems are developed based on vision sensors, considering its affordable cost and installation. As a result, majority of the research is conducted for fire detection using cameras.

The available literature dictates that flame detection using visible light camera is the generally used fire detection method, which has three categories including pixel-level, blob-level, and patch-level methods. The pixel-level methods [5], [6] are fast due to usage of pixel-wise features such as colors and flickers, however, their performance is not attractive as such methods can be easily biased. Compared to pixel-level methods, blob-level flame detection methods [7] show better performance as such methods consider blob-level candidates for features extraction to detect flame. The major problem with such methods is the difficulty in training their classifiers due to numerous shapes of fire blobs. Patch-level algorithms [3], [8] are developed to improve the performance of previous two categories of flame detection algorithms, however, such methods result in many outliers, affecting their accuracy.

To improve the accuracy, researchers attempted to explore color and motion features for flame detection. For instance, Chen *et al.* [6] investigated the dynamic behavior and irregularity of flames in both RGB and HSI color spaces for fire detection. Since, their method considers the frame difference during prediction, hence, it fails to differentiate real fire from fire-like moving outliers and objects. Besides RGB and HSI color models, Marbach *et al.* [9] explored YUV color model in combination with motion features for prediction of fire and non-fire pixels. A similar method is proposed by Töreyin *et al.* [7] by investigating temporal and spatial wavelet analysis, however, the excessive use of parameters by this method limits its usefulness. Another method is presented by Han and Lee [10] by comparing the video frames and their color features for flame detection in tunnels. Continuing the investigation of color models, Celik and Demirel [11] used YCbCr with specific rules of separating chrominance component from luminance. The method has potential to detect flames with good accuracy but at small distance and larger size of fire only. Considering these limitations, Borges and Izquierdo [12] attempted to detect fire using a multimodal framework consisting of color, skewness, and roughness features and Bayes classifier.

In continuation with Borges and Izquierdo [12] work, multi-resolution 2D wavelets combined with energy and shape are explored by Rafiee *et al.* [13] in an attempt to reduce false warnings, however, the false fire alarms still remained significant due to movement of rigid body objects in the scene. An improved version of this approach is presented in [14] using YUC instead of RGB color model, providing better results than [13]. Another color based flame detection method with speed 20 frames/sec is proposed in [15]. This scheme used SVM classifier to detect fire with good accuracy at smaller distance. The method showed poor performance when fire is at larger distance or the amount of fire is comparatively small. Summarizing the color based methods, it is can be noted that such methods are sensitive to brightness and shadows. As a result, the number of false warnings produced by these methods is high. To cope with such issues, the flame's shape and rigid objects movement are investigated by Mueller *et al.* [16]. The presented method uses optical flow information and behavior of flame to intelligently extract a feature vector based on which flame and moving rigid objects can be differentiated. Another related approach consisting of motion and color features, is proposed by [17] for flame detection in surveillance videos. To further improve the accuracy, Foggia *et al.* [14] combined shape, color, and motion properties, resulting in a multi-expert framework for real-time flame detection. Although, the method dominated state-of-the-art flame detection algorithms, yet there is still space for improvement. In addition, the false alarming rate is still high and can be further reduced. From the aforementioned literature, it is observed that fire detection accuracy has inverse relationship to computational complexity. With this motivation, there is a need to develop fire detection algorithms with less computational cost and false warnings, and higher accuracy. Considering the above motivation, we extensively studied convolutional neural networks (CNNs) for flame detection at early stages in CCTV surveillance videos. The main contributions of this article are summarized as follows:

1. Considering the limitations of traditional hand-engineering methods, we extensively studied deep learning (DL) architectures for this problem and propose a cost-effective CNN framework for flame detection in CCTV surveillance videos. Our framework avoids the tedious and time consuming process of feature engineering and automatically learns rich features from raw fire data.
2. Inspired from transfer learning strategies, we trained and fine-tuned a model with architecture similar to GoogleNet [18] for fire detection, which successfully dominated traditional fire detection schemes.
3. The proposed framework balances the fire detection accuracy and computational complexity as well as reduces the number of false warnings compared to state-of-the-art fire detection schemes. Hence, our scheme is more suitable for early flame detection during surveillance to avoid huge fire disasters.

The rest of the paper is organized as follows: In Section 2, we present our proposed architecture for early flame detection in surveillance videos. Experimental results and discussion are given in Section 3. Conclusion and future directions are given in Section 4.

II. THE PROPOSED FRAMEWORK

Majority of the research since the last decade is focused on traditional features extraction methods for flame detection. The major issues with such methods is their time consuming process of features engineering and their low performance for flame detection. Such methods also generate high number of false alarms especially in surveillance with shadows, varying lightings, and fire-colored objects. To cope with such issues, we extensively studied and explored deep learning architectures for early flame detection. Motivated by the recent improvements in embedded processing capabilities and potential of deep features, we investigated numerous CNNs to improve the flame detection accuracy and minimize the false warnings rate. An overview of our framework for flame detection in CCTV surveillance networks is given in Figure 1.

A. CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE

CNN is a deep learning framework which is inspired from the mechanism of visual perception of living creatures.

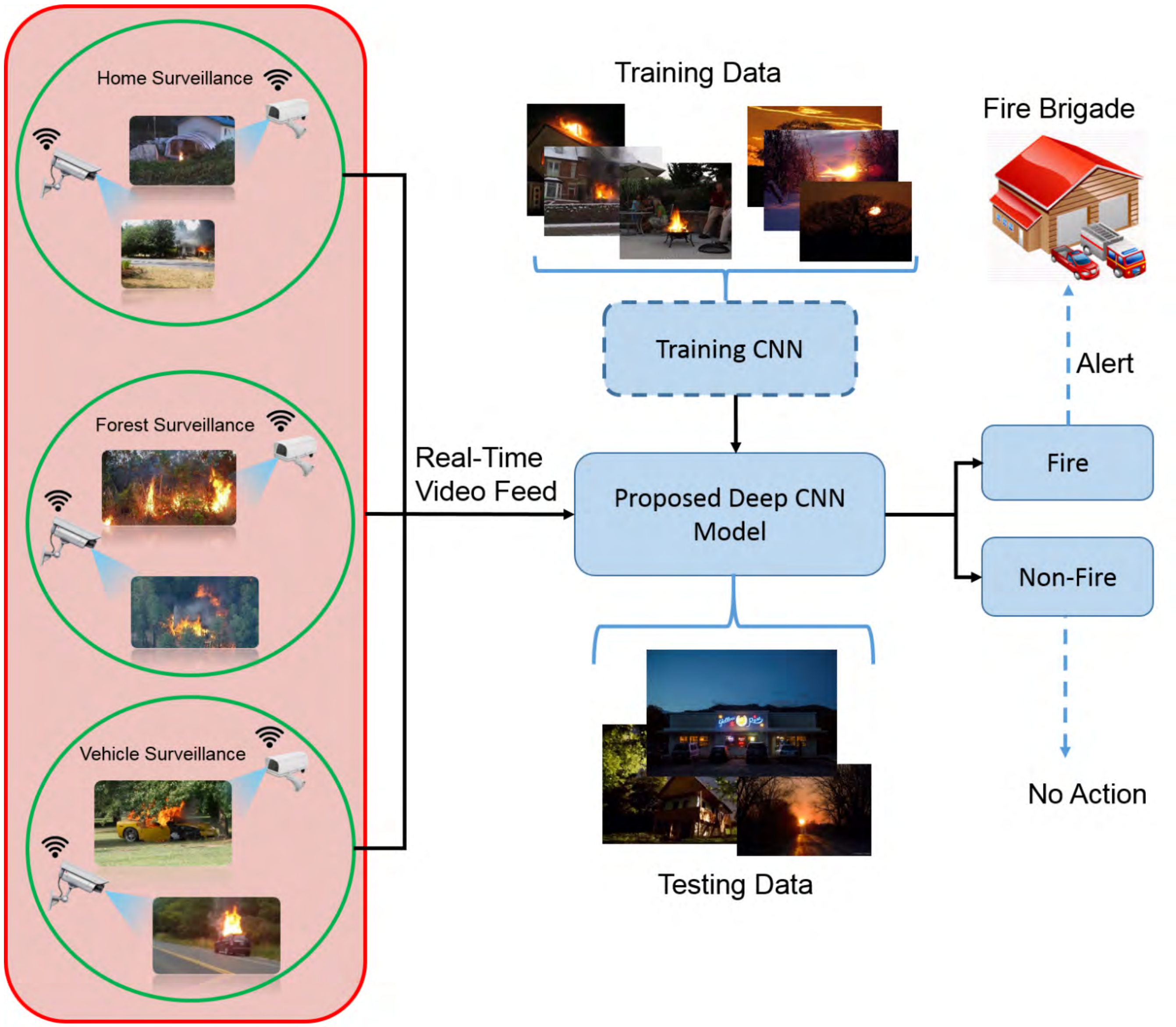

FIGURE 1. **Early flame detection in surveillance videos using deep CNN.**

Since the first well-known DL architecture LeNet [19] for hand-written digits classification, it has shown promising results for combating different problems including action recognition [20], [21], pose estimation, image classification [22]–[26], visual saliency detection, object tracking, image segmentation, scene labeling, object localization, indexing and retrieval [27], [28], and speech processing. Among these application domains, CNNs have extensively been used in image classification, achieving encouraging classification accuracy over large-scale datasets compared to hand-engineered features based methods. The reason is their potential of learning rich features from raw data as well as classifier learning. CNNs generally consist of three main operations as illustrated in Figure 2.

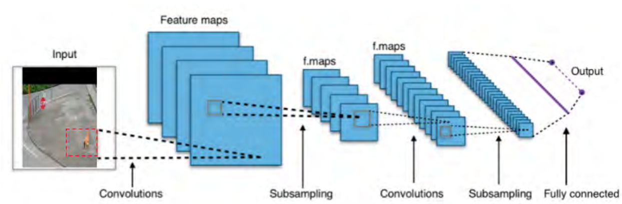

FIGURE 2. **Main operations of a typical CNN architecture.**

In convolution operation, several kernels of different sizes are applied on the input data to generate feature maps. These features maps are input to the next operation known as subsampling or pooling where maximum activations are

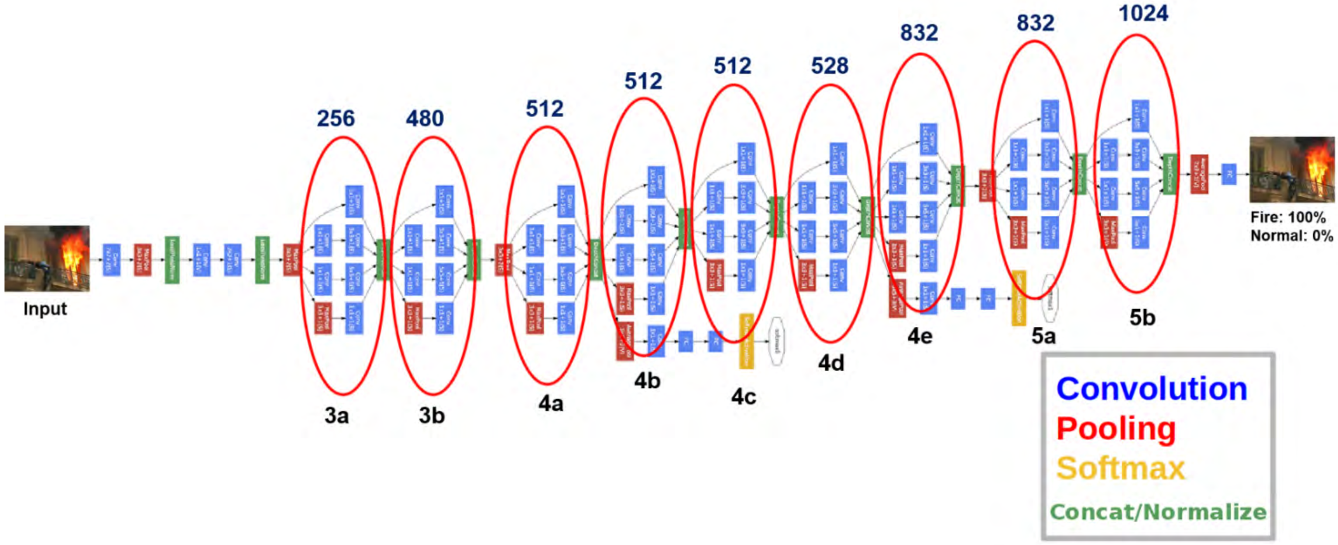

FIGURE 3. Architectural overview of the proposed deep CNN.

FIGURE 4. Probability scores and predicted labels produced by the proposed deep CNN framework for different images from benchmark datasets.

selected from them within small neighborhood. These operations are important for reducing the dimension of feature vectors and achieving translation invariance up to certain degree. Another important layer of the CNN pipeline is fully connected layer, where high-level abstractions are modeled from the input data. Among these three main operations, the convolution and fully connected layers contain neurons whose weights are learnt and adjusted for better representation of the input data during training process.

For the intended classification problem, we used a model similar to GoogleNet [18] with amendments as per our problem. The inspirational reasons of using GoogleNet compared to other models such as AlexNet include its better classification accuracy, small sized model, and suitability of implementation on FPGAs and other hardware architectures having memory constraints. The intended architecture consists of 100 layers with 2 main convolutions, 4 max pooling, one average pooling, and 7 inception modules as given in Figure 3.

FIGURE 5. Sample frames from videos of Dataset1. The top two rows are sample frames from fire videos while the remaining two rows represent sample frames from normal videos.

The size of input image is 224 × 224 × 3 pixels on which 64 kernels of size 7 × 7 are applied with stride 2, resulting in 64 feature maps of size 112 × 112. Then, a max pooling with kernel size 3 × 3 and stride 2 is used to filter out maximum activations from previous 64 feature maps. Next, another convolution with filter size 3 × 3 and stride 1 is applied, resulting in 192 feature maps of size 56 × 56. This is followed by another max pooling layer with kernel size 3 × 3 and stride 2, filtering discriminative rich features from less important ones. Next, the pipeline contains two inception layers (3a) and (3b). The motivational reason of such inception modulus assisted architecture is to avoid uncontrollable increase in the computational complexity and networks' flexibility to significantly increase the number of units at each stage. To achieve this, dimensionality reduction mechanism is applied before computation-hungry convolutions of patches with larger size. The approach used here is to add 1 × 1 convolutions for reducing the dimensions, which in turn minimizes the computations. Such mechanism is used in each inception module for dimensionality reduction. Next, the architecture contains a max pooling layer of kernel size 3×3 with stride 2, followed by four inception modules 4 (a-e). Next, another max pooling layer of same specification is added, followed by two more inception layers (5a and 5b). Then, an average pooling layer with stride 1 and filter size 7×7 is introduced in the pipeline, followed by a dropout layer to avoid overfitting. At this stage, we modified the architecture according to our classification problem by keeping the number of output classes to 2 i.e., fire and non-fire.

B. FIRE DETECTION IN SURVEILLANCE VIDEOS USING DEEP CNN

It is highly agreed among the research community that deep learning architectures automatically learn deep features from

(i) Fire004 (ii) Fire046 (iii) Fire045 (iv) Fire104
(v) not_fire081 (vi) not_fire035 (vii) not_fire088 (viii) not_fire090

FIGURE 6. Sample images from Dataset2. Images in row 1 show fire class and images of row 2 belong to normal class.

raw data, yet some effort is required to train different models with different settings for obtaining the optimal solution of the target problem. For this purpose, we trained numerous models with different parameter settings depending upon the collected training data, its quality, and problem's nature. We also applied transfer learning strategy which tends to solve complex problems by applying the previously learned knowledge. As a result, we successfully improved the flame detection accuracy up to 6% from 88.41% to 94.43% by running the fine-tuning process for 10 epochs. After several experiments on benchmark datasets, we finalized an optimal architecture, having the potential to detect flame in both indoor and outdoor surveillance videos with promising accuracy. For getting inference from the target model, the test image is given as an input and passed through its architecture. The output is probabilities for two classes i.e., fire and non-fire. The maximum probability score between the two classes is taken as the final label of a given test image. To illustrate this procedure, several images from benchmark datasets with their probability scores are given in Figure 4.

III. RESULTS AND DISCUSSION

In this section, all experimental details and comparisons are illustrated. We conducted experiments from different perspectives using images and videos from different sources. All experiments are performed using NVidia GeForce GTX TITAN X with 12 GB onboard memory and deep learning framework [29] and Ubuntu OS installed on Intel Core i5 CPU with 64 GB RAM. The experiments and comparisons are mainly focused on benchmark fire datasets: Dataset1 [14] and Dataset2 [30]. However, we also used data from other two sources [31], [32] for training purposes. The total number of images used in experiments is 68457, out of which 62690 frames are taken from Dataset1 and remaining from other sources. As a principle guideline for training and testing, we followed the experimental strategy of Foggia *et al.* [14] by using 20% data of the whole dataset for training and the remaining 80% for testing. To this end, we used 20% of fire data for training our GoogleNet based flame detection model. Further details about datasets, experiments, and comparisons are illustrated in the following sub-sections.

A. PERFORMANCE ON DATASET1

Dataset1 is collected by Foggia *et al.* [14], containing 31 videos which cover different environments. This dataset has 14 fire videos and 17 normal videos without fire. The dataset is challenging as well as larger in size, making it a better option for experiments. The dataset has been made challenging for both color-based and motion-based fire detection methods by capturing videos of fire-like objects and mountains with smoke and clouds. This is one of the motivations for selection of this dataset for our experiments. Figure 5 shows sample images from this dataset. Table 1 shows the experimental results based on Dataset1 and its comparison with other methods.

The results are compared with other flame detection methods, which are carefully selected using a selection criteria, reflecting the features used for fire detection, time, and dataset. The best results are reported by [14] among the existing recent methods by achieving an accuracy of 93.55% with 11.67% false alarms. The score of false alarms is still high and needs further improvement. Therefore, we explored

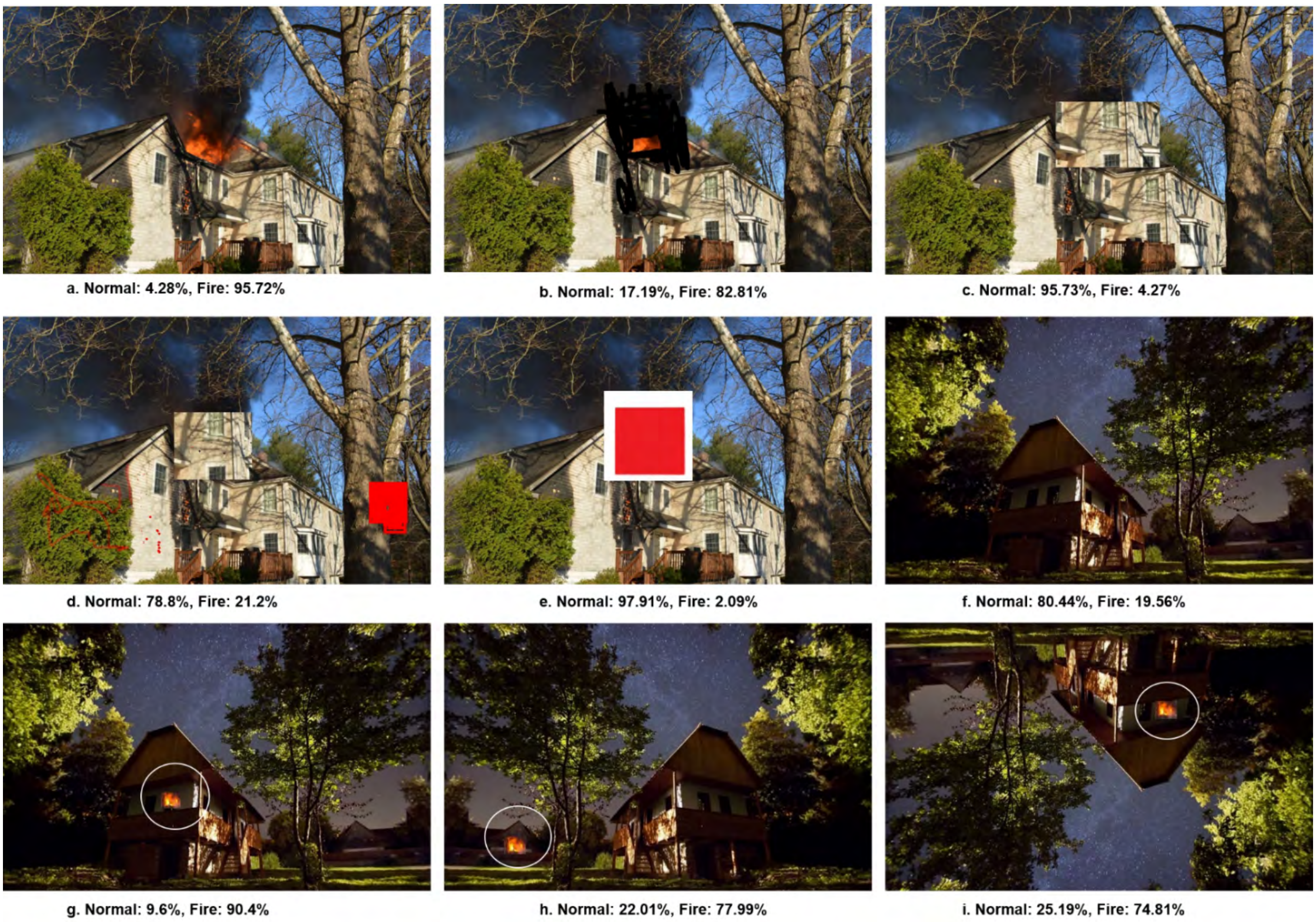

FIGURE 7. **Effect on fire detection accuracy for our proposed method against different attacks. Images with caption "a", "b", and "g-i" are labeled as fire while images (c, d, e, and f) are labeled as normal by our method.**

TABLE 1. **Comparison with different fire detection methods.**

Technique	False Positives (%)	False Negatives (%)	Accuracy (%)
Proposed after fine tuning (FT)	**0.054**	1.5	**94.43**
Proposed before FT	0.11	5.5	88.41
Muhammad et al. [2] (after FT)	9.07	2.13	94.39
Muhammad et al. [2] (before FT)	9.22	10.65	90.06
Foggia et al. [14]	11.67	**0**	93.55
De Lascio et al. [17]	13.33	0	92.86
Habibuglu et al. [15]	5.88	14.29	90.32
Rafiee et al. (RGB) [13]	41.18	7.14	74.20
Rafiee et al. (YUV) [13]	17.65	7.14	87.10
Celik et al. [11]	29.41	**0**	83.87
Chen et al. [6]	11.76	14.29	87.10

deep learning architectures (AlexNet and GoogleNet) for this purpose. The results of AlexNet for fire detection are taken from our recent work [2]. Initially, we trained GoogleNet model with its default kernel weights which resulted in an accuracy of 88.41% with false positives score of 0.11%. The baseline GoogleNet architecture randomly initializes the kernel weights which are tuned according to the accuracy and error rate during the training process. In an attempt to improve the accuracy, we explored transfer learning [33] by initializing the weights from pre-trained GoogleNet model and keep the learning rate threshold to 0.001. Further, we also changed the last fully connected layer as per the nature of the intended problem. With this fine-tuning process, we reduced the false alarms rate from 0.11% to 0.054% and false negatives score from 5.5% to 1.5%, respectively.

B. PERFORMANCE ON DATASET2

Dataset2 was obtained from [30], containing 226 images out of which 119 images belong to fire class and 107 images belong to non-fire class. The dataset is small but very challenging as it contains red-colored and fire-colored objects, fire-like sunlight scenarios, and fire-colored lightings in different buildings. Figure 6 shows sample images from this dataset. It is important to note that no image from Dataset2 was used in training the proposed model for fire detection. The results are compared with five methods including both hand-crafted features based methods and deep learn-

TABLE 2. Results of Dataset2 for the proposed method and other fire detection methods.

Technique	Precision	Recall	F-Measure
Proposed after fine tuning (FT)	0.80	0.93	0.86
Proposed before FT	**0.86**	0.89	0.88
Muhammad et al. [2] (after FT)	0.82	**0.98**	**0.89**
Muhammad et al. [2] (before FT)	0.85	0.92	0.88
Chino et al. [30]	0.4-0.6	0.6-0.8	0.6-0.7
Rudz et al. [36]	0.6-0.7	0.4-0.5	0.5-0.6
Rossi et al. [37]	0.3-0.4	0.2-0.3	0.2-0.3
Celik et al. [11]	0.4-0.6	0.5-0.6	0.5-0.6

ing based method. These papers for comparison were selected based on their relevancy, underlying dataset used for experiments, and year of publication. Unlike experimental metrics of Table 1, we used other metrics (precision, recall, and F-measure [34], [35]) as used by [30] for evaluating the performance of our work from different perspectives. The collected results using Dataset2 for our method and other algorithms are given in Table 2. Although, the overall performance of our method using Dataset2 is not better than our recent work [2], yet it is competing with it and is better than hand-crafted features based fire detection methods.

C. EFFECT ON THE PERFORMANCE AGAINST DIFFERENT ATTACKS

In this section, we tested the effect on performance of our method against different attacks such as noise, cropping, and rotation. For this purpose, we considered two test images: one from fire class and second from normal class. The image from fire class is given in Figure 7 (a), which is predicted as fire by our method with accuracy 95.72%. In Figure 7 (b), the fire region in the image is distorted and the resultant image is passed through our method. Our method still assigned it the label "fire" with accuracy 82.81%. In Figure 7 (c), the fire region is blocked and our method successfully predicted it as normal. To show the effect on performance against images with fire-colored regions, we considered Figure 7 (d) and Figure 7 (e) where red-colored boxes are placed on different parts of the image. Interestingly, we found that the proposed method still recognizes it correctly as "normal". In Figure 7 (f), we considered a normal challenging image which is predicted as normal by our method with accuracy 80.44%. To confirm that our method can detect small amount of fire, we placed small amount of fire on Figure 7 (f) in different regions and investigated the predicted label. As shown in Figure 7 (g, h, and i), our method assigned them the correct label of fire. These tests indicate that the proposed algorithm can detect fire even if the video frames are effected by noise or the amount of fire is small and at a reasonable distance, in real-world surveillance systems, thus, validating its better performance.

IV. CONCLUSION

The recent improved processing capabilities of smart devices have shown promising results in surveillance systems for identification of different abnormal events i.e., fire, accidents, and other emergencies. Fire is one of the dangerous events which can result in great losses if it is not controlled on time. This necessitates the importance of developing early fire detection systems. Therefore, in this research article, we propose a cost-effective fire detection CNN architecture for surveillance videos. The model is inspired from GoogleNet architecture and is fine-tuned with special focus on computational complexity and detection accuracy. Through experiments, it is proved that the proposed architecture dominates the existing hand-crafted features based fire detection methods as well as the AlexNet architecture based fire detection method.

Although, this work improved the flame detection accuracy, yet the number of false alarms is still high and further research is required in this direction. In addition, the current flame detection frameworks can be intelligently tuned for detection of both smoke and fire. This will enable the video surveillance systems to handle more complex situations in real-world.

REFERENCES

[1] K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. H. Ge Wang, and S. W. Baik, "secure surveillance framework for IoT systems using probabilistic image encryption," *IEEE Trans. Ind. Inform.*, to be published, doi: https://doi.org/10.1109/TII.2018.2791944

[2] 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, May 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231217319203

[3] J. Choi and J. Y. Choi, "An integrated framework for 24-hours fire detection," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 463–479.

[4] H. J. G. Haynes. (2016). *Fire Loss in the United States During 2015*. [Online]. Available: http://www.nfpa.org/

[5] C.-B. Liu and N. Ahuja, "Vision based fire detection," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2004, pp. 134–137.

[6] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in *Proc. Int. Conf. Image Process. (ICIP)*, Oct. 2004, pp. 1707–1710.

[7] B. U. Töreyin, Y. Dedeoğlu, U. Güdükbay, and A. E. Çetin, "Computer vision based method for real-time fire and flame detection," *Pattern Recognit. Lett.*, vol. 27, pp. 49–58, Jan. 2006.

[8] J. Choi and J. Y. Choi, "Patch-based fire detection with online outlier learning," in *Proc. 12th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2015, pp. 1–6.

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

[10] D. Han and B. Lee, "Development of early tunnel fire detection algorithm using the image processing," in *Proc. Int. Symp. Vis. Comput.*, 2006, pp. 39–48.

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

[12] P. V. K. Borges and E. Izquierdo, "A Probabilistic approach for vision-based fire detection in videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 5, pp. 721–731, May 2010.

[13] A. Rafiee, R. Dianat, M. Jamshidi, R. Tavakoli, and S. Abbaspour, "Fire and smoke detection using wavelet analysis and disorder characteristics," in *Proc. 3rd Int. Conf. Comput. Res. Develop. (ICCRD)*, Mar. 2011, pp. 262–265.

[14] 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, pp. 1545–1556, Sep. 2015.

[15] Y. H. Habiboğlu, O. Günay, and A. E. Çetin, "Covariance matrix-based fire and flame detection method in video," *Mach. Vis. Appl.*, vol. 23, no. 6, pp. 1103–1113, 2012.

[16] 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, Jul. 2013.

[17] R. Di Lascio, A. Greco, A. Saggese, and M. Vento, "Improving fire detection reliability by a combination of videoanalytics," in *Proc. Int. Conf. Image Anal. Recognit.*, 2014, pp. 477–484.

[18] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[19] Y. Le Cun *et al.*, "Handwritten digit recognition with a back-propagation network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1990, pp. 396–404.

[20] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action recognition in video sequences using deep Bi-directional LSTM with CNN features," *IEEE Access*, vol. 6, pp. 1155–1166, 2017.

[21] A. Ullah *et al.*, "Action recognition in movie scenes using deep features of keyframes," *J. Korean Inst. Next Generation Comput.*, vol. 13, pp. 7–14, 2017.

[22] L. Shao, L. Liu, and X. Li, "Feature learning for image classification via multiobjective genetic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 7, pp. 1359–1371, Jul. 2014.

[23] F. Li, L. Tran, K. H. Thung, S. Ji, D. Shen, and J. Li, "A robust deep model for improved classification of AD/MCI patients," *IEEE J. Biomed. Health Inform.*, vol. 19, no. 5, pp. 1610–1616, Sep. 2015.

[24] R. Zhang, J. Shen, F. Wei, X. Li, and A. K. Sangaiah, "Medical image classification based on multi-scale non-negative sparse coding," *Artif. Intell. Med.*, vol. 83, pp. 44–51, Nov. 2017.

[25] O. W. Samuel *et al.*, "Pattern recognition of electromyography signals based on novel time domain features for amputees' limb motion classification," *Comput. Elect. Eng.*, to be published, doi: https://doi.org/10.1016/j.compeleceng.2017.04.003

[26] Y.-D. Zhang *et al.*, "Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation," *Multimedia Tools Appl.*, pp. 1–20, Sep. 2017. [Online]. Available: https://link.springer.com/article/10.1007/s11042-017-5243-3

[27] J. Yang, B. Jiang, B. Li, K. Tian, and Z. Lv, "A fast image retrieval method designed for network big data," *IEEE Trans. Ind. Inform.*, vol. 13, no. 5, pp. 2350–2359, Oct. 2017.

[28] J. Ahmad, K. Muhammad, and S. W. Baik, "Data augmentation-assisted deep learning of hand-drawn partially colored sketches for visual search," *PLOS ONE*, vol. 12, no. 8, p. e0183838, 2017.

[29] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[30] D. Y. T. Chino, L. P. S. Avalhais, J. F. Rodrigues, and A. J. M. Traina, "BoWFire: Detection of fire in still images by integrating pixel color and texture analysis," in *Proc. 28th SIBGRAPI Conf. Graph., Patterns Images*, Aug. 2015, pp. 95–102.

[31] S. Verstockt *et al.*, "Video driven fire spread forecasting (f) using multi-modal LWIR and visual flame and smoke data," *Pattern Recognit. Lett.*, vol. 34, no. 1, pp. 62–69, 2013.

[32] B. C. Ko, S. J. Ham, and J. Y. Nam, "Modeling and formalization of fuzzy finite automata for detection of irregular fire flames," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 12, pp. 1903–1912, Dec. 2011.

[33] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[34] K. Muhammad, J. Ahmad, M. Sajjad, and S. W. Baik, "Visual saliency models for summarization of diagnostic hysteroscopy videos in healthcare systems," *SpringerPlus*, vol. 5, p. 1495, Dec. 2016.

[35] K. Muhammad, M. Sajjad, M. Y. Lee, and S. W. Baik, "Efficient visual attention driven framework for key frames extraction from hysteroscopy videos," *Biomed. Signal Process. Control*, vol. 33, pp. 161–168, Mar. 2017.

[36] S. Rudz, K. Chetehouna, A. Hafiane, H. Laurent, and O. Séro-Guillaume, "nvestigation of a novel image segmentation method dedicated to forest fire applications," *Meas. Sci. Technol.*, vol. 24, no. 7, p. 075403, 2013.

[37] L. Rossi, M. Akhloufi, and Y. Tison, "On the use of stereovision to develop a novel instrumentation system to extract geometric fire fronts characteristics," *Fire Safety J.*, vol. 46, nos. 1–2, pp. 9–20, 2011.

KHAN MUHAMMAD (S'16) received the bachelor's degree in computer science from the Islamia College, Peshawar, Pakistan, with a focus on information security. He is currently pursuing the M.S. leading to Ph.D. degree in digital contents from Sejong University, Seoul, South Korea. He has been a Research Associate with the Intelligent Media Laboratory since 2015. He has authored over 40 papers in peer-reviewed international journals and conferences, such as the IEEE Transactions on Industrial Informatics, *Future Generation Computer Systems*, *PLOS One*, the IEEE Access, the *Journal of Medical Systems*, *Biomedical Signal Processing and Control*, *Multimedia Tools and Applications*, *Pervasive and Mobile Computing*, *SpringerPlus*, the *KSII Transactions on Internet and Information Systems*, MITA 2015, PlatCon 2016, FIT 2016, Platcon-17, and ICNGC-2017. His research interests include image and video processing, information security, image and video steganography, video summarization, diagnostic hysteroscopy, wireless capsule endoscopy, computer vision, deep learning, and video surveillance. He is an Active Reviewer of over 30 reputed journals and is involved in editing of several special issues.

JAMIL AHMAD (S'16) received the B.C.S. degree (Hons.) in computer science from the University of Peshawar, Pakistan, in 2008, and the master's degree with specialization in image processing from the Islamia College, Peshawar, Pakistan, in 2014, and the Ph.D. degree in digital contents from Sejong University, Seoul, South Korea, in 2017. He is currently a Faculty Member with the Department of Computer Science, Islamia College Peshawar. He has published several journal articles in these areas in reputed journals including the *Journal of Real-Time Image Processing*, *Multimedia Tools and Applications*, the *Journal of Visual Communication and Image Representation*, *PLOS One*, the *Journal of Medical Systems*, *Computers and Electrical Engineering*, *SpringerPlus*, the *Journal of Sensors*, and the *KSII Transactions on Internet and Information Systems*. He is also an Active Reviewer for *IET Image Processing*, *Engineering Applications of Artificial Intelligence*, the *KSII Transactions on Internet and Information Systems, Multimedia Tools and Applications*, the IEEE Transactions on Image Processing, and the IEEE Transactions on Cybernetics. His research interests include deep learning, medical image analysis, content-based multimedia retrieval, and computer vision.

IRFAN MEHMOOD (M'16) received the B.S. degree in computer science from the National University of Computer and Emerging Sciences, Pakistan, and the Ph.D. degree from Sejong University, Seoul, South Korea. He is currently an Assistant Professor with the College of Electronics and Information Engineering, Sejong University, Seoul, South Korea. His research interests include video and medical image processing, big data analysis, and visual information summarization.

SEUNGMIN RHO (M'08) received the B.S. degree in computer science and the M.S. and Ph.D. degrees in information and communication technology from the Graduate School of Information and Communication, Ajou University, South Korea. Before he joined the Computer Sciences Department, Ajou University, he spent two years in industry. He visited the Multimedia Systems and Networking Laboratory, The University of Texas at Dallas, from 2003 to 2004. From 2008 to 2009, he was a Post-Doctoral Research Fellow with the Computer Music Laboratory, School of Computer Science, Carnegie Mellon University. From 2009 to 2011, he was a Research Professor with the School of Electrical Engineering, Korea University. In 2012, he was an Assistant Professor with the Division of Information and Communication, Baekseok University. He is currently a faculty of the Department of Media Software, Sungkyul University, South Korea. His current research interests include database, big data analysis, music retrieval, multimedia systems, machine learning, knowledge management, and computational intelligence.

SUNG WOOK BAIK (M'16) received the B.S. degree in computer science from Seoul National University, Seoul, South Korea, in 1987, the M.S. degree in computer science from Northern Illinois University, Dekalb, in 1992, and the Ph.D. degree in information technology engineering from George Mason University, Fairfax, VA, USA, in 1999. He was with Datamat Systems Research Inc., as a Senior Scientist, the Intelligent Systems Group, from 1997 to 2002. In 2002, he joined the faculty of the College of Electronics and Information Engineering, Sejong University, Seoul, South Korea, where he is currently a Full Professor and a Dean of digital contents. He is also the Head of Intelligent Media Laboratory, Sejong University. His research interests include computer vision, multimedia, pattern recognition, machine learning, data mining, virtual reality, and computer games.