Computers, Materials & Continua
DOI: 10.32604/cmc.2022.030963
Article

Tech Science Press

A Deep Real-Time Fire Prediction Parallel D-CNN Model on UDOO BOLT V8

Amal H. Alharbi and Hanan A. Hosni Mahmoud*

Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
*Corresponding Author: Hanan A. Hosni Mahmoud. Email: hahosni@pnu.edu.sa

Received: 07 April 2022; Accepted: 29 May 2022

Abstract: Hazardous incidences have significant influences on human life, and fire is one of the foremost causes of such hazard in most nations. Fire prediction and classification model from a set of fire images can decrease the risk of losing human lives and assets. Timely promotion of fire emergency can be of great aid. Therefore, construction of these prediction models is relevant and critical. This article proposes an operative fire prediction model that depends on a prediction unit embedded in the processor UDOO BOLT V8 hardware to predict fires in real time. A fire image database is improved to enhance the images quality prior to classify them as either fire or non-fire. Our proposed deep learning–based Very Deep Convolutional Networks Visual Geometry Group (VGG-16) model (Parallel VGG-16) is an enhanced version of the VGG-16 model, by incorporating parallel convolution layers and a fusion module for better accuracy. The experimental results validate the performance of the Parallel VGG-16 which achieves an accuracy of 97%, compared to the compared state-of-the-art models. Moreover, we integrate the prediction module into a UDOO BOLT V8 computer, which precisely controlled the fire alarm so that it can cautious people from fire in real time. In this paper we propose a complete fire prediction model using a camera and the UDOO BOLT V8 embedded system. Our experiments validate the effectiveness and applicability of the proposed fire model.

Keywords: Fire prediction; UDOO BOLT V8; deep learning

1 Introduction

We live in a contemporary dynamic era with a lot of technology related devices. The volume of fire related incidence is increasing due to demand for dangerous products, which can originate life threatening hazards leading to loss of lives and assets [1]. Many countries such as the United states and Australia suffer from outdoor huge fires and usually seek well-developed fire detection automated models [2–5].

Fires, if not detected in an early phase, will lead to human life and monetary losses. Forest and outdoor fires are very dangerous and can highly spread fires that will need several years for loss restoration. Therefore, it is vital to precisely and timely detect such fires [1–3].

Local sensors [3–6] are the frequently utilized fire detecting methods for screening temperature and smoke and other vital fire features. In temperature sensors, the authors in [7] presented a sensor set to rate the fire phases by measuring the heat variance between the different surfaces internally and externally. The authors in [8] studied the close and distant field heat sensor set to identify fire phases. Authors in [9] utilized electronic and heat-couple temperature sensors to detect heated surface in link to fire phases. Gas sensors are described in [9] where gas and smoke detection are performed by the output disparity of semiconductor devices, infrared and calorimetric devices. Using fire blob sensing, both emission and hue features are used to form heat and spatial fire sensing methods [8–10]. Smoke is considered a nonvisual measure with pyrolysis, and blobs visual features are captured by cameras to detect blob movement. Smoke detection systems that identify fire phases are discussed in [11–14]. Usually, the flame sensing models can identify early-stage fires with fewer false negatives. Nevertheless, as characteristic founded models, the current local sensors yield great false negatives for complicated fire situations.

Paralleled to local sensors, image models for fire prediction and classification could efficiently decrease the impact of the external environment. At the breakout of a fire, smoke delivers well-timed prospective data in contrary to fire flames. Consequently, operative smoke detection models have a significant part in fire classification and detection. Computerized fire detection models are mainly machine learning models that use mined features for classifiers learning phase such as random forest model, and support vector machine system [15–17]. Nevertheless, most current research on intelligent fire detection models usually uses single type of smoke or blob extraction from captured videos. It is frequently hard to detect extra complex fire circumstances. Also, many fire detection models only utilize images that include smokes only in an idyllic background with no noises. Video frame segmentation is the main algorithm in image and video processing, which is of high importance to automated vision procedures. Indeed, frame segmentation was comprehensively investigated in visual recognition processes, such as I-ray image segmentation, surveillance imagery, etc. Authors in [18] joined frame segmentation and regression to identify fluctuations in radar videos. Authors in [19], employed fuzzy C-means clustering technique for magnetic resonance classification, and then mined feature from the Magnetic resonance (MR) images to for classification. C-means algorithm was also used to extract pedestrian edge as an objects in a complicated scenario and then predictors are used for pedestrian identification [20]. A variation detection intelligent models, for aerial vehicle movement detection, were created using fuzzy c-means algorithm to choose training features [21]. Smoke detection displayed a prominence part in fire identification. Authors in [22] proposed a smoke identification model using an iterative classification support vector machine (SVM) model by extracting spatial features of fire regions. Nevertheless, the images segmentation technique, they used, mixed and distorted the detection accuracy. To extract spectral features from fire areas, the authors in [23], segmented fire areas by Slice-Spectral model, then computed the local binary features such as variant features. Though, the extracted features were found to be not robust for moveable fire situations. The authors in [24], established a Gaussian feature mixture algorithm using hue-intensity-pair segmentation for the preprocessing of fire videos with dynamic progress features of various fire phases. Authors in [23] defined warm air imaging by fuzzy C-means algorithm for early stage detection. In [24], they employed motion data and analyzed it using Markov random model for fire segmented images.

With the speedy advance of deep learning models [25–28], many studies investigated deep learning models for classification for various applications [28–30]. Lately, the authors in [25] presented a smoke concentration prediction neural network. The abundant semantic data needed by intelligent encoder-decoder networks are used to compute smoke concentration from fire videos [26]. In [27], the authors used an energy- efficient neural network for fire and flame identification, and semantic considerate computation of the fire scenario. The authors in [28] joined the pixel and object-level prominent neural architectures to mine the smoke feature maps utilizing video frame sequence. High efficient region convolutional neural network (CNN). Long short-term memory neural network are utilized to identify the alleged areas of fire and predict a Fire or Non-fire in a short-time [26]. In the meantime, deep learning models was used for fire prediction modules using generic algorithms [17]. The authors in [18] presented a video smoke identification model by incorporating Kalman coefficient, hue analysis and image integration. Flame labeling using shape and color features were used for early stage alarm in fire detection models. The authors in [28] established a bag-of-features function to produce a random forest prediction model for high accuracy classification of the plane features to identify fire candidate regions.

Therefore, a novel fire detection and classification deep learning model is proposed by integrating a deep real time fire prediction model on UDOO BOLT V8 hardware. The experimental results prove that our model outperforms several state of the art models in accuracy and time.

The rest of this article is organized as follows: Section 2 depicts the problem statement and the methodology details of the proposed model. Experimental are depicted in Section 3. Conclusion are depicted in Section 4.

2 Problem Statement

A deep real time fire prediction model on UDOO BOLT V8 should satisfy these requirements:

- A fire video frame from the dataset captured in real time should be detected as Fire or Non-fire.
- For real-time fire alarm system, an alarm of fire should be generated automatically and transferred to the appropriate authority.
- The model should be accurate and with high practical value, especially to forest fires. These fires can cause environmental hazards where life losses can reach an extraordinary threatening level. Fire detection is one of the key element in receiving government venture.

Our research faced these challenges by:

- Presenting a real time accurate fire detection model to classify the fires in video frames with high precision reducing false positives and negatives.
- Proposing an embedded hardware module in the high-performance UDOO BOLT V8 hardware to efficiently detect fires and send alarms in real time.
- Testing the fire prediction model and comparing its accuracy to state-of-the-art models.

2.1 The Automatic Fire Prediction Model

Previous research studies, in the last years, are used feature extraction models for fire classification and detection. The main issues with these models are their high time requirements for feature extraction and their low accuracy for fire detection. These models also produce high number of false positives and negatives particularly in images with shadows, noises and fluctuating illuminations, and fire same-hue objects. To face these matters, we comprehensively investigated deep learning models for early stage fire detection. Inspired by the current advances in embedded systems processing aptitudes and ability of deep feature extraction, we studied many deep neural networks to enhance the fire detection performance and optimize the false positives and negatives rate. An outline of our platform for fire detection in surveillance cameras architectures is depicted in Fig. 1.

Our proposed fire prediction model is intended to perform the following processes:

- Capture an image every few seconds and submit it to the model.
- Detection of fire or no-fire in real time.
- Delete the image of non-fire or send an alarm if the image is classified

To recognize these processes, the proposed model will have two central modules:

- Fire prediction model on UDOO BOLT V8 hardware
- Surveillance module and servo motor hardware

In this paper, a proposed parallel deep neural network presented for Fire/Non-fire classification and a 16 GB UDOO BOLT V8 model is embedded to control the surveillance camera, which captures videos, and the servo motor. In Section 3 the dataset description and implementation details are described. In Section 4, experimental results are demonstrated. Conclusions and limitations are depicted in Section 5.

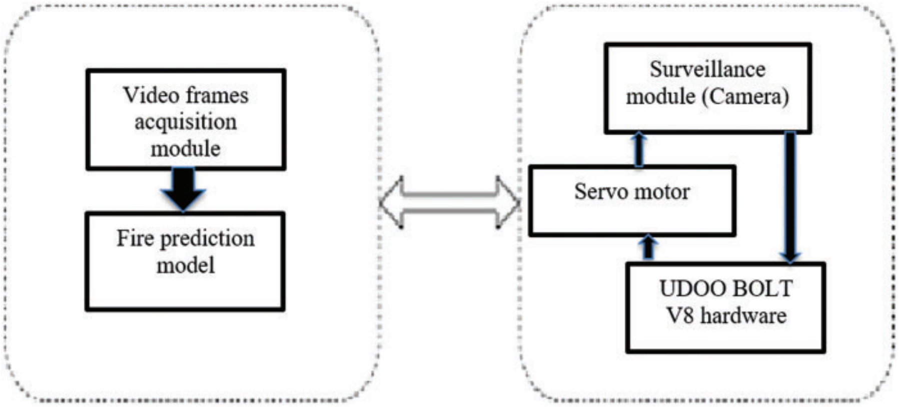

Figure 1: Framework of the proposed model

2.2 Fire Detection Module

Fire detection module is a key constituent of the whole fire detection model which regulates the effectiveness of the model. We present a deep learning CNN utilizing VGG-16. Features of the VGG-16, the utilized datasets, and the experimental setting are described in the following sub-sections.

2.2.1 VGG-16 Deep Convolutional Neural Network

Very Deep Convolutional Networks for Large-Scale Image Recognition (VGG-16) [16] were introduced in 2015 using ImageNet dataset detection and segmentation. There are various types of VGG-16 networks with diverse layer structures such as VGG-21, VGG-34, VGG-64. In which the network VGG is trailed by the layers count. VGG-16 is a deep CNN that has many convolutional/pooling and output layers. A challenge has been encountered when designing a deep CNN with many layers is the converging Gradient which restricts the training process as depicted in Tab. 1.

VGG-16 is a collection of blocks, built with many convolutional/pooling and output layers. VGG-16 network is similar to ResNet model [26] with multiple stack convolutions layers. The main concept is to avoid the current layer by connecting it to the preceding layer. The concept of the feedback block is to feed the layer input to the convolutional-pooling-convolutional layers to compute $G(I)$ and $Z(I)$ such that $Z(I) = G(I) + I$ (Eqs. 1 and 2). The training process is superior if features extracted from the preceding layers are summed. VGG-16 utilizes a feed forward network to go through several convolutional layers. As depicted in Tab. 1.

Table 1: VGG-16 network layers

Layer #	Layer type	Properties
1	Input Layer	256, 256, 3 images
2	First Convolutional Layer	1:64 filters
3	First Pooling Layer	Max pooling
4	Second Convolutional Layer	Input Frame: (112,112) 1:128 filters
5	Second Pooling Layer	3 × 3 Maxpooling
6	Third Convolutional Layer	Input Frame: (56,56) 1: 256 filters
7	Third Pooling Layer	Maxpooling
8	Fourth Convolutional Layer	Input Frame: (28, 28) 1: 512 filters
9	Fourth Pooling Layer	Maxpooling
10	Fifth Convolutional Layer	Input Frame: (14, 14) 1: 512 filters
11	Fifth Pooling Layer	Maxpooling
12	Fusion module	
13	First Fully Connected (FC1) Layer	2048 neurons, producing a (1, 2048) feature vector
14	Second Fully Connected (FC2) Layer	2048 neurons producing a (1, 2048) feature vector
15	Third Fully Connected (FC3) Layer	2048 neurons producing a (1, 2048) feature vector
16	Classifier	Softmax
17	Output Layer	Two output classes: 1. Fire 2. Non-Fire

The value *Z(I)* is the predicted solution, while, the value of *G(I)* is the ground truth (the labeled data item). *Z(I)* is probably equal to or nearly equal to *G(I)*. *G(I)* is computed from the input *I* by performing the following equation:

$$I \rightarrow W1 \rightarrow Relu \rightarrow W2 \qquad (1)$$

where W1 and W2 are specified weights

and *Z(I)* is computed by:

$$Z\,(I) = G(I) + I \rightarrow ReLU \qquad (2)$$

The VGG-16 is designed on the network depicted in Fig. 2.

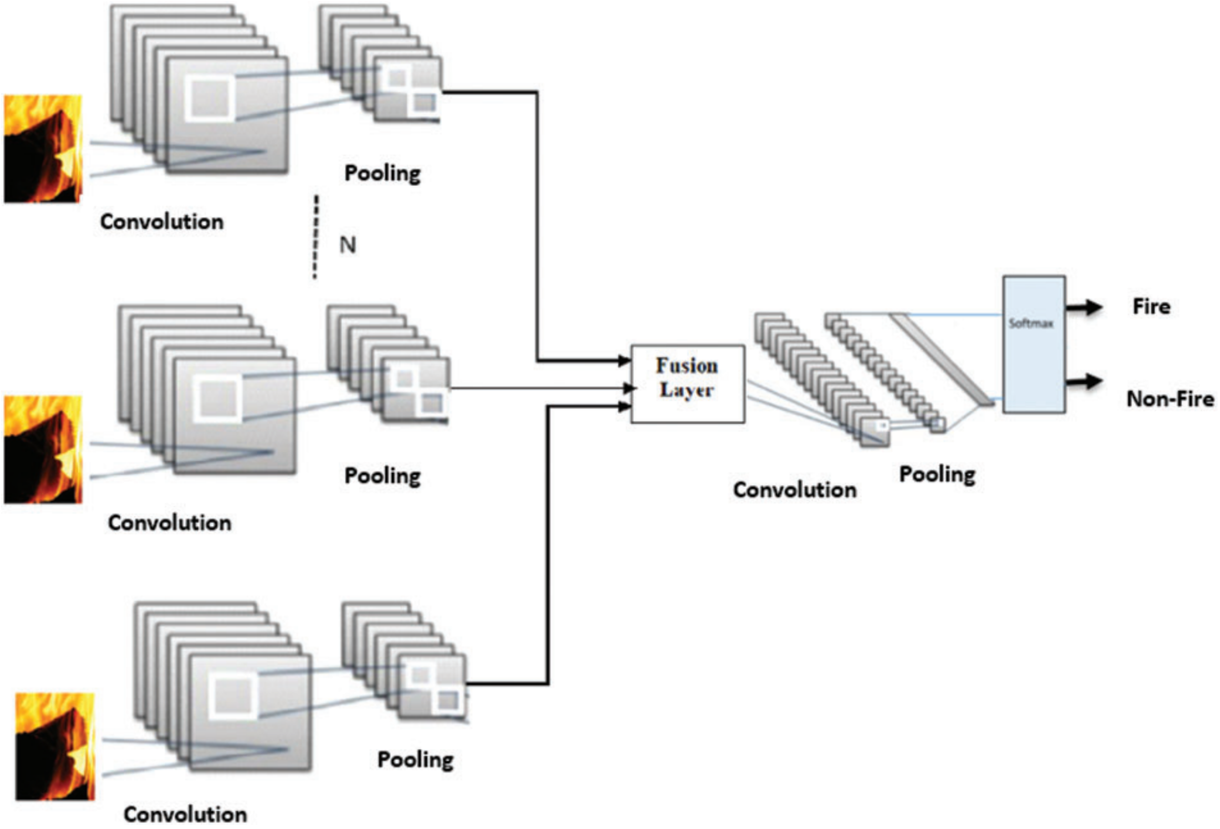

Figure 2: The parallel VGG-16 network with parallel convolution and pooling layers with fusion layer

The Parallel VGG-16 CNN network has zero-padding layers, five convolution layers, 2 MaxPooling layers, and several fully connected (FC) layer.

The FC layers employ Softmax classifier for input normalization into a probability distribution function and then perform the prediction and classification the input frame into a labelled data item.

3 Dataset Description and Implementation Details

3.1 Dataset Description

In the model, we use a public dataset (Fire-Dat) that is depicted in [15]. It is composed of videos of indoor fires, outdoor fires and no-fire at different day time periods (day, night, dark and semi dark).

The dataset contains 6200 frame sequence in set of 5, of several sizes. The frames are labelled as Fire or Non-fire by three specialists. To compute the agreement consensus value between the three labels, we use kappa coefficient [13] as depicted in Eq. (3).

$$kappa = \frac{Cs\,(a) - Prob\,(e)}{1 - Prob\,(e)} \tag{3}$$

where, $Cs\,(a)$ is the consensus agreement value computed from the input expert values, and $Prob\,(e)$ is the expected consensus likelihood. Then kappa coefficient of the used public dataset Fire-Dat labeling is equal to 0.97 which depicts a high value of agreement consensus for the specialist labeling of the three labels for the same fire frame-set.

In the following phase, the Fire-Dat dataset is divided into three subsets 70% for training, 15% for testing and 15% for validation.

These frame-sets are transformed into 256 × 256 × 3 frames, and are associated with the defined label in the corresponding folders. Fig. 3 shows some video frames in fire training and evaluation folders with size 256 × 256 × 3 pixels.

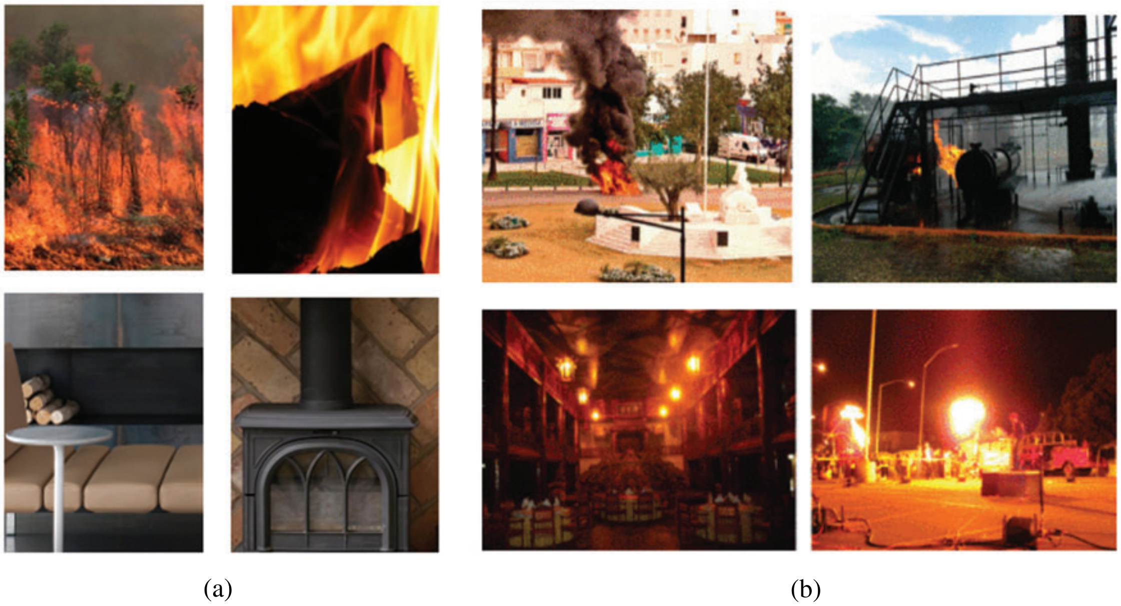

Figure 3: Video frames from the Fire-Dat dataset, the first rows in Figs. 3(a, b) are frames including fires, the last rows in Figs. 3(a, b) do not include fires

The distribution statistics of the dataset, according to Fire and Non-fire, and the distribution of the training, validation and testing datasets, are depicted in Tabs. 2 and 3, respectively. Distribution of fire sizes in the Fire-Dat dataset is depicted in Fig. 4.

Table 2: Statistics of dataset according to fire/non-fire distribution

Type of frame	Number of frame	%
Fire	3235	52,19%
Non-Fire	2965	47.81%
	Total 6200 frames	

Table 3: Distribution of subsets

Subset	Number of frames	%
Training subset	4340	70%
Validation subset	930	15%
Testing subset	930	15%

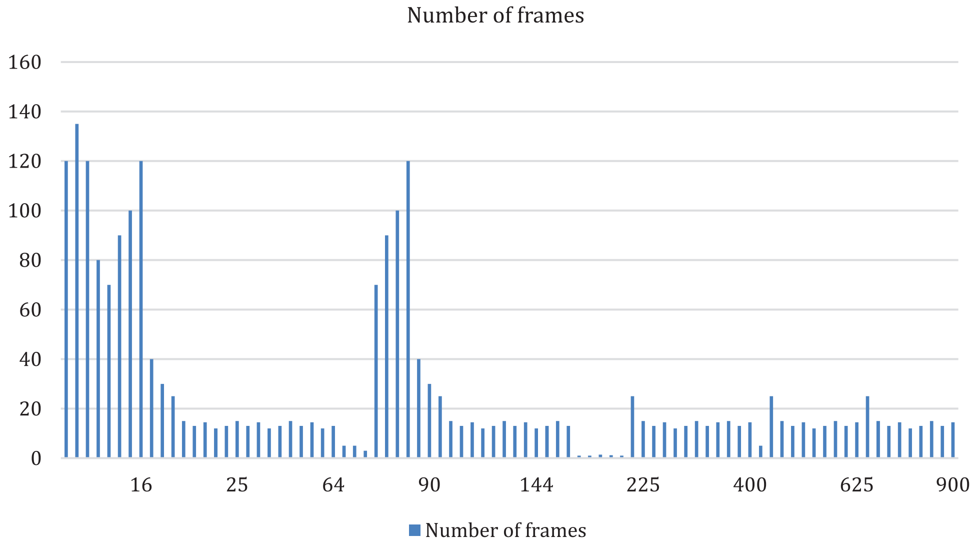

Figure 4: Distribution of fire sizes in the Fire-Dat dataset

3.2 *Experiment Setting*

The proposed fire detection model is executed as an embedded UDOO BOLT V8 hardware using the HP Pavilion 27-d1000, CPU: i7 (3.80 GHz), GPU: Pavilion GTX 1050 8 GB, RAM: 32 GB. The simulation factors of the VGG-16 model are depicted in Tab. 4.

Table 4: The simulation factors of the proposed fire detection model

Parameter	Description
Input image size	256 x 256 x 3
Batch	64
Number of Epochs	200
Activation	Rectified linear activation function (Relu)
Classifier	Softmax

3.3 UDOO BOLT V8 Model Implementation

The model uses UDOO BOLT V8 on a 16 Gigabyte board with 64-bit Operating system. UDOO BOLT V8 board is united with UDOO BOLT V8 Camera and Node-RED service. The camera has a microcontroller which connects with the UDOO BOLT V8 through the MQTT controller. The model will capture images every 5 s which are fed to the processing and prediction modules. Based on the classification, UDOO BOLT V8 will rotate the servo motor to capture more images from different directions to be a set of 5 images and will repeat the classification by changing the utilized parameters weights.

UDOO BOLT V8 is employed in our model because of its benefits, such as:

General Features	UDOO BOLT V8 is a high performance configuration. Four cores each of 64 bits, Broadcom BCM2711, 64-bit SoC @ 1.5 GHz, Memory (RAM) 16 GB, 40 pins input/output (GPIO) with easy integration
CPU	AMD V1605B Four Core @ 3.0 ghz (3.6 ghz Boost)
Operating System	Windows 10 X86 64bit
Microcontroller	XT32U4

Fig. 5. Depicts the parts of the parallel fire detection model:

- UDOO BOLT V8 Unit: comprises of a Camera with image procurement module; UDOO BOLT V8 board and Servo motor. This unit obtains the classification results from the deep learning unit and if the result is Fire, the servo motor will rotate to take images from different directions and return them back to the deep learning module.
- Deep learning module: is implemented in Python 3, liable for acquiring, analysing and classifying fire images.

4 Experimental Results

We performed experiments with the proposed Parallel VGG-16 and performed comparison with the start-of-the-art models, such as Long Short Term Memory Networks, Recurrent Neural Networks, Generative Adversarial Networks, Radial Basis Function Networks and Multilayer Perceptrons.

Experiment results employing Parallel VGG-16 are depicted in Tabs. 5 and 6.

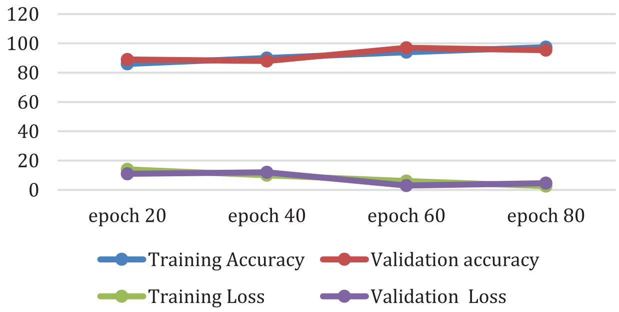

Figure 5: Training and validation using Parallel VGG-16

Table 5: Experiment results using Parallel VGG-16

Parameter	Description
Experiment result accuracy	97%
Test loss	4%
Deep learning Parallel VGG-16 Training time	640 min
epochs	80
Average time per epoch	8 min

Table 6: Performance evaluation of fire detection

Model	Mean average precision mAP %	mAP % for fire	mAP % for no-fire	Run time (seconds)
Long Short Term Memory Networks [12]	75.6	71.6	77.9	23.6
Recurrent Neural Networks (RNNs) [15]	73.9	69.7	76.8	33.5
Generative Adversarial Networks [23]	74.7	70.2	79	18.9
Radial Basis Function Networks [20]	80.5	75	82	26.5
Multilayer Perceptrons [13]	84.7	77	87	22.4
Our proposed model (3 parallel blocks)	95.9	93.3	97	13.5

The training and validation accuracy and loss using VGG-16 are shown in Fig. 5.

The deep learning training model takes 640 min with 80 epochs, the mean training time for each epoch takes 8 min. The accuracy attained is high with 97.4% accuracy.

We employed our Parallel VGG-16 with different number of parallel blocks for evaluation of accuracy and time comparisons. The parallel VGG-16 underwent deep training on a the mentioned public dataset [26]. Different parallel blocks models contributing to the experiment are:

- Serial 1-block VGG model with a single block of convolutional/Maxpooling layers followed by FC/Softmax classifier layers (no fusion layer)
- The 3-parallel block parallel VGG model encompasses two parallel convolutional/Maxpooling layers blocks followed by fusion layer and then FC/Softmax classifier layers.
- The 5-parallel block parallel VGG model encompasses two parallel convolutional/Maxpooling layers blocks followed by fusion layer and then FC/Softmax classifier layers.
- The 7-parallel block parallel VGG model encompasses two parallel convolutional/Maxpooling layers blocks followed by fusion layer and then FC/Softmax classifier layers.
- The 10-parallel block parallel VGG model encompasses two parallel convolutional/Maxpooling layers blocks followed by fusion layer and then FC/Softmax classifier layers.

Experimental results, of different parallel blocks configuration, are shown in Fig. 6. The results display the compromise between accuracy and time complexity using a weighted metric Acc_T as depicted in Tab. 7.

Acc_T is computed as follows:

$$Acc_T = \frac{\frac{Accuracy}{100} + Fusiontime}{\text{Classification time}} \tag{4}$$

From the results in Tab. 5 it is seen that with 5 blocks there is a compromise between accuracy and classification time. Fire detection experiment results are averaged over 200 runs for various blocks and are depicted in Tabs. 7 and 8.

Table 7: The comparison of proposed model and nominate models

Model	Epochs	Prediction accuracy	Fusion time (s)	Classification time	Acc_T
1-Block serial VGG	80	83%	0	8	0.104
3-Block Parallel VGG	80	89%	1.5	13.7	0.174
5-Block Parallel VGG	80	92%	3.2	20.5	0.201
7-Block Parallel VGG	80	97%	4.7	23.5	0.241
10-Block Parallel VGG	80	98.9%	8.7	29.9	0.326

Table 8: Detection experiment results averaged over 200 runs for our proposed model with different number of parallel blocks

Model	Accuracy %	Sensitivity %	Specificity %	Fire detection accuracy %	No-fire detection accuracy %
Our proposed model: one block	89.67	87.37	87.92	90.17	87.78
Our proposed model: 3 parallel blocks	91.32	91.62	91.18	92.59	92.47
Our proposed model: 5 parallel blocks	93.41	91.33	91.93	94.91	91.91
Our proposed model: 7 parallel blocks	95.42	93.32	93.88	96.99	93.87
Our proposed model: 10 parallel blocks	98.8	95.4	96.2	97.7	96.8

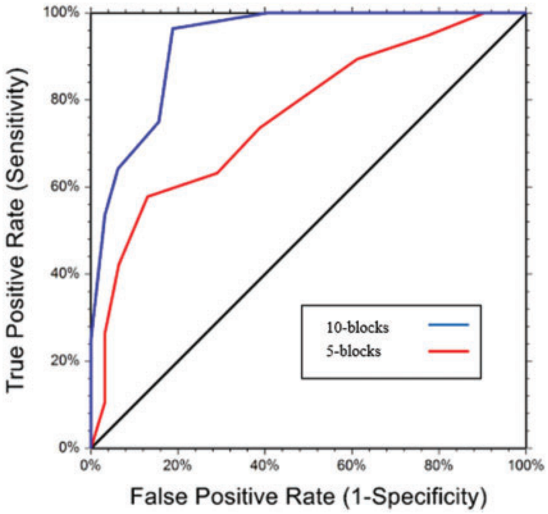

Figure 6: Area under the curve comparison

Some real-time images on fire prediction are shown in Fig. 7.

We investigated the models with the 5, 7 and 10 parallel blocks with the greatest accuracy, including 5-block parallel VGG, 7-block parallel VGG-16, and 10-block parallel VGG-16. Fig. 8 depicts the confusion matrices showing the false positives and negatives which are decreased highly using our proposed model.

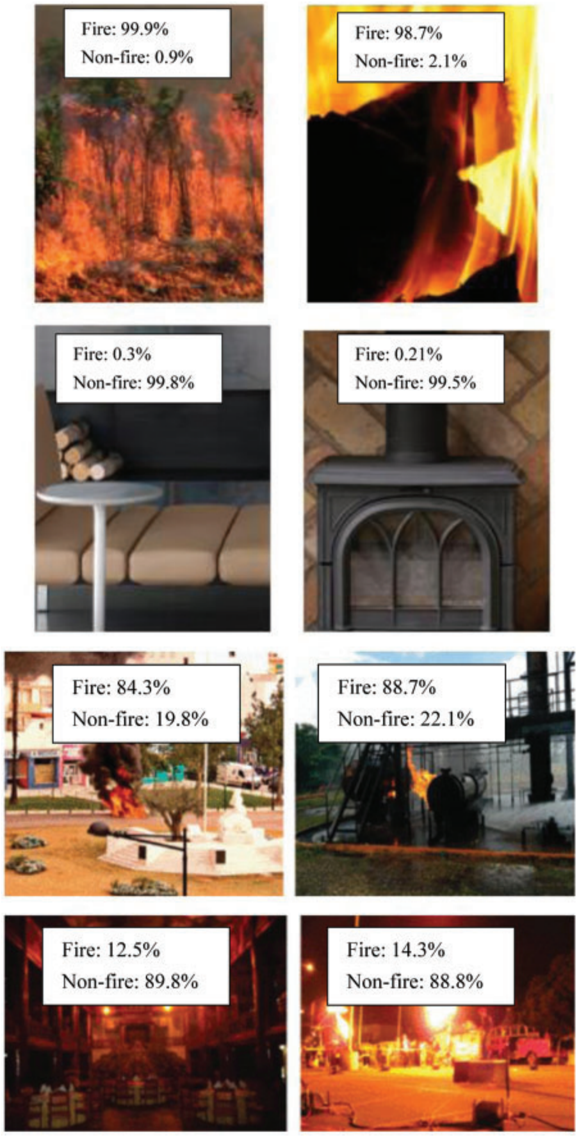

Figure 7: Experimental results on real images in the dataset with accuracy percentage

		Predicted cases	
		Positive	Negative
Actual	Positive	482	9
	Negative	10	382

(a) 5-Block Parallel VGG

		Predicted cases	
		Positive	Negative
Actual	Positive	487	4
	Negative	6	386

(b) 7-Block Parallel VGG

		Predicted cases	
		Positive	Negative
Actual	Positive	489	2
	Negative	3	389

(c) 10-Block Parallel VGG-16

Figure 8: Confusion matrices using 7% of the images in the dataset for 5, 7 and 10 block parallel VGG-16 models

5 Conclusion

Fires, if not detected in an early phase, will lead to human life and monetary losses. Forest and outdoor fires are very dangerous and highly spread fires that will need several years for loss restoration. Therefore, it is vital to precisely and timely detect such fires. This research proposed an intelligent fire prediction model that combined deep learning with embedded system. This article proposes an operative fire prediction model that depend on a prediction unit embedded in a UDOO BOLT V8 hardware to predict fires in real time. A fire image database is improved to enhance the images quality prior to classify them as either fire or non-fire. Our proposed deep learning–based VGG-16 model (Parallel VGG-16) is an enhanced version of the VGG-16 model, by incorporating parallel convolution layers and a fusion module for better accuracy. The experimental results validate the performance of the Parallel VGG-16. Our model achieves an accuracy of 97%, as compared to the state-of-art models. Moreover, integrating the prediction module into a UDOO BOLT V8 computer, precisely controls the fire alarm so that it can cautious people from fire in real time. In this paper, we proposed a complete fire prediction model using a camera and the UDOO BOLT V8 embedded system. We employed our Parallel VGG-16 with different number of parallel blocks for evaluation of accuracy and time comparisons. The parallel VGG-16 CNN underwent deep training on the mentioned public dataset. Our experiments validated the effectiveness and applicability of the proposed fire model. We investigated models with the 5, 7 and 10 parallel blocks with the greatest accuracy, including 5-block parallel VGG, 7-block parallel VGG-16, and 10-block parallel VGG-16. The results depicted the confusion matrices showing the false positives and negatives which are decreased highly using our proposed model.

Limitations of the proposed model is the cost of the UDOO BOLT V8 processor as it is expensive and might not be appropriate to be installed in households. Other embedded system can be employed and validated by experiments. In future extension we can use multiple cameras instead of one rotating camera for better performance. Although, this can be suitable for big incorporations but it will be costly for typical households.

Acknowledgement: We would like to thank Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R120), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R120), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] 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 Transaction on Circuits Systems and Video Technology*, vol. 25, no. 1, pp. 1545–1556, 2019.

[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, no. 2, pp. 30–42, 2019.

[3] J. Choi and Y. Choi, "An integrated framework for 24-hours fire detection," in *Proc. of European Conf. of Computer Vision*, Paris, France, pp. 463–479, 2020.

[4] X. Zhang, Z. Zhao and J. Zhang, "Contour based forest fire detection using FFT and wavelet," in *Proc. of the Int. Conf. on Computer Science and Software Engineering*, Hubei, China, pp. 760–763, 2019.

[5] C. Liu and N. Ahuja, "Vision based fire detection," in *Proc. 17th Int. Conf. of Pattern Recognition*, Athens, Greece, pp. 134–137, 2019.

[6] T. Chen, P. Wu and Y. Chiou, "An early fire-detection method based on image processing," in *Proc. Int. Conf. of Image Processing (ICIP)*, Cleveland, Ohio, pp. 1707–1710, 2020.

[7] B. Töreyin, Y. Dedeoğlu, U. Güdükbay and A. Çetin, "Computer vision-based method for real-time fire and flame detection," *Pattern Recognition Letter*, vol. 27, no. 1, pp. 49–58, 2021.

[8] J. Choi and Y. Choi, "Patch-based fire detection with online outlier learning," in *Proc. 12th IEEE Int. Conf. of Advanced Video Signal Based Surveillance. (AVSS)*, London, England, pp. 1–6, 2019.

[9] G. Marbach, M. Loepfe and T. Brupbacher, "An image processing technique for fire detection in video images," *Fire Safety Journal*, vol. 41, no. 2, pp. 285–289, 2021.

[10] D. Han and B. Lee, "Development of early tunnel fire detection algorithm using the image processing," in *Proc. Int. Sym. of Visual Computing*, pp. 39–48, 2020.

[11] T. Çelik and H. Demirel, "Fire detection in video sequences using a generic color model," *Fire Safety Journal*, vol. 44, no. 2, pp. 147–158, 2019.

[12] P. Borges and E. Izquierdo, "A probabilistic approach for vision-based fire detection in videos," *IEEE Trans. Circuits Systems Video Technology*, vol. 20, no. 5, pp. 721–731, 2020.

[13] R. Dianat, M. Jamshidi, R. Tavakoli and S. Abbaspour, "Fire and smoke detection using wavelet analysis and disorder characteristics," in *Proc. 3rd Int. Conf. of Computational Development*, Cairo, Egypt, pp. 262–265, 2019.

[14] P. Govil, K. Welch, M. Ball and J. Pennypacker, "Preliminary results from a wildfire detection system using deep learning on remote camera images," *Remote Sensors*, vol. 8, no. 1, pp. 76–89, 2020.

[15] Y. Habiboğlu and O. Günay, "Fire-Dat: Fire and flame datset," *Machine Visual Applications*, vol. 23, no. 6, pp. 1103–1113, 2020.

[16] M. Mueller, P. Karasev, I. Kolesov and A. Tannenbaum, "Optical flow estimation for flame detection in videos," *IEEE Transaction on Image Processing*, vol. 22, no. 7, pp. 2786–2797, 2019.

[17] R. Lascio, A. Greco, A. Saggese and M. Vento, "Improving fire detection reliability by a combination of video analytics," in *Proc. Int. Conf. of Image Analytics Recognition*, Ostrava, CZ, pp. 477–484, 2019.

[18] C. Szegedy and M. Pethay, "Going deeper with convolutions," in *Proc. IEEE Conf. of Computing of Visual Pattern Recognition*, Napoli, Italy, pp. 1–9, 2019.

[19] Y. Cun, X. Yu and Z. Chou, "Handwritten digit recognition with a back-propagation network," in *Proc. Advanced Neural Information Processing Systems*, Alexandria, Go, pp. 396–404, 1990.

[20] J. Ullah, K. Ahmad, K. Muhammad and M. Sajjad, "Action recognition in video sequences using deep Bi-directional LSTM with CNN features," *IEEE Access*, vol. 6, no. 2, pp. 1155–1166, 2021.

[21] M. Ullah, A. Mohamed and M. Morsey, "Action recognition in movie scenes using deep features of keyframes," *Journal of Korean Institute of Next Generation Computation*, vol. 13, no. 1, pp. 7–14, 2020.

[22] B. Chen, C. Yuan and J. Song, "SmokeNet: Satellite smoke scene detection using convolutional neural network with spatial and channel-wise attention," *Remote Sensors*, vol. 2, no. 1, pp. 256–267, 2019.

[23] R. Zhang, J. Shen and K. Sangaiah, "Medical image classification based on multi-scale non-negative sparse coding," *Artificial Intellectual Medical Informatics*, vol. 83, no. 1, pp. 44–51, 2020.

[24] O. Samuel, H. Hanouk and M. Bayoumi, "Pattern recognition of electromyography signals based on novel time domain features for limb motion classification," *Computation Electrical Engineering*, vol. 3, no. 1, pp. 45–57, 2020.

[25] A. Krizhevsky, A. Sutskever and I. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems, Pereira*, vol. 2, no. 1, pp. 1097–1105, 2019.

[26] A. Z. Vo, L. Hoang Son, M. T. Vo and T. Le, "A novel framework for fire prediction using deep transfer learning," *IEEE Access*, vol. 7, no. 1, pp. 178631–178639, 2019.

[27] A. Sun and Z. Xiao, "ThanosNet: A novel fire prediction model using metadata," in *Proc. of the IEEE Int. Conf. on Big Data (Big Data 2020)*, Atlanta, GA, USA, pp. 1394–1401, 2020.

[28] A. Masand, S. Chauhan, M. Jangid, R. Kumar and S. Roy, “ScrapNet: An efficient approach to fire prediction,” *IEEE Access*, vol. 9, no. 1, pp. 130947–130958, 2021.

[29] W. Sun, X. Chen, X. R. Zhang, G. Z. Dai, P. S. Chang *et al.,* “A multi-feature learning model with enhanced local attention for vehicle re-identification,” *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3549–3560, 2021.

[30] W. Sun, G. C. Zhang, X. R. Zhang, X. Zhang and N. N. Ge, “Fine-grained vehicle type classification using lightweight convolutional neural network with feature optimization and joint learning strategy,” *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 30803–30816, 2021.