iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VI **Month of publication:** June 2023

DOI: https://doi.org/10.22214/ijraset.2023.53888

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

Real Time Fire Detection using Deep Convolutional Neural Networks and Long-Short Term Memory in Video Surveillance

Abhilash S[1], Amogha Natekar J[2], Charan S[3], Deekshith G D[4]
Department of Computer Science and Engineering, Sapthagiri College of Engineering, #14/5, Chikkasandra, Hesaraghatta Main Road, Bangalore-57, India

Abstract: ***Fire detection remains a significant area of research with relevant, practical safety solutions and is an essential component of a building safety monitoring system. The most typical catastrophe is fire. Negligence on the part of individuals is unquestionably the primary cause. Sensor-based systems that have independently detected temperature and smoke perform the majority of fire detections. In order to construct an automated fire detection system that can handle complex real-world fire occurrences, an effective monitoring system is an extremely important development. An early discovery framework is important to keep fires from fanning crazy. In image classification and other computer vision tasks, convolutional neural networks (CNNs) have demonstrated cutting-edge performance. The main issue with CNN-based fire recognition frameworks is their execution progressively observation organizations. In order to eliminate false fire alarms, we propose a computationally efficient CNN architecture for fire detection by comprehending the scene and employing long short-term memory. The input received from the CNN model is used to represent the genuine fire. This framework is built utilizing highlight map determination calculation and fire limitation calculation for fire recognition, confinement and semantic comprehension of the scene and discovery of the fire is gathered which is essentially because of its expanded profundity with effectiveness and precision by thinking about the particular qualities of the issue of interest and the assortment of fire information.***
Keywords: fire detection, deep learning, convolutional neural networks and long short-term memory.

I. INTRODUCTION

Fires are frequently mentioned in the news. Fires kill thousands of people and damage billions of dollars' worth of property each year. When managing apartment buildings, warehouses, forests, substations, railways, and tunnels, fire monitoring and protection are always major concerns. The consequences are frequently disastrous if fires are not discovered early enough and grow out of control. Fostering a framework that can consequently distinguish fire at a beginning phase is vital for safeguarding both human existences as well as property.

Sensors like carbon dioxide detectors, ionization detectors, and photoelectric detectors are used in the first generation of fire detection systems. Sensor-based systems still have a lot of limitations, especially in large, open areas, despite some success in detecting fire. Since each sensor only monitors a small area, a fire far from a sensor cannot be immediately detected. Additionally, the method's accuracy is heavily reliant on sensor reliability and sensor density, which results in some installation and cost issues. A lot of people use sensor-based fire detection. However, it is common to miss real fires and have a high false detection rate.

In the field of security surveillance, digital cameras, specifically closed-circuit television (CCTV), have been rapidly evolving. Security cameras are less complicated to set up than sensor-based systems and can be used to monitor large, open areas. Currently, CCTV is prevalent everywhere, and employing a CCTV system for fire monitoring may be a cost-effective option. As of late, sensor-based frameworks have begun to be supplanted by reconnaissance cameras and video investigation frameworks. A lot of image processing algorithms have been proposed for using video analysis to detect smoke and fire, and some of them have been very successful. Although it has been demonstrated that computer vision-based systems outperform sensor-based systems, their performance and outcomes are still far from ideal.

Algorithms that use sample data, also known as training data, to automatically construct a mathematical model and make decisions without being explicitly programmed to do so are known as machine learning (ML) algorithms. Since the 1950s, machine learning has been evolving; in any case, at first, the documented outcomes were not extremely great. The central troubles looked during this period in the advancement of AI were information assortment and the restrictions of processing assets.

©IJRASET: All Rights are Reserved

Data collection has become easier than ever in the last ten years, however, as computers have become faster and the internet has grown in popularity, and the development of machine learning (ML) is rapidly progressing. A cutting-edge machine learning algorithm has recently emerged as a new subfield of ML known as deep learning.

The performance of these algorithms in computer vision applications, such as image classification and object detection, has been extremely impressive. This huge change gives an amazing chance to tackle numerous issues that actually exist in PC vision, including fire recognition.

This paper proposes a strategy to handle fire recordings utilizing a multiple-stages CNN-LSTM model. A CNN model and an LSTM model are used in the proposed approach to effectively detect fire in videos by extracting spatial and temporal features. The following are the method's main contributions:

For detecting fire in videos, we have proposed a method that combines fire candidate extraction with CNN-LSTM classification.

1) This will accurately determine fire
2) The proposed technique can recognize fire at various scales furthermore, ecological circumstances.
3) The proposed technique is quick; as a result, it can be incorporated into actual applications.
4) We have gathered a fire images dataset for preparing and testing the calculation.

The following is the order of Sections in the paper:

Area 2 audits the connected works. The proposed system is explained in Section 3. The outcomes of the experiment are described in Section 4. Future examination bearings and a conversation are given in Section 5.

II. RELATED WORK

Many methods for using video surveillance to detect fires early have been suggested in the literature. The workflows of these methods are primarily divided into the following three substeps: feature extraction, classification, and fire-colored pixel detection.

We can easily infer from observation that the flame's colour varies depending on the material and temperature from red-yellow to white. Many proposed methods rely on modelling fire-colored pixels, and flame colour is a crucial element of fire detection in many algorithms. These methods may be loosely divided into the following three groups: Gaussian distribution-based models, polynomial-based colour models, and fire colour rule-based methods. Numerous heuristic rules that are based on the RGB (Red, Green, and Blue) [1]–[6], Ycbcr [7]–[9], YUV [10], or Lab [11] colour spaces are necessary for the colour rules. A rule is estimated to categorise the fire-colored pixels in a picture after the fire region is manually divided and the connection between the pixel values of the three channels is examined.

A two-dimensional depiction of the colour of the flame pixel is the RGB distribution of the fire flame pixels in photographs.

Observing various types of fires will yield various colour distributions and flame pixel processes depending on the burning substance. To categorise the fire-colored pixels in a picture based on this distribution, many heuristic methods along with decision thresholds are computed.

Despite the fact that several techniques are utilised to simulate the colour of flames, the majority of these methods have had success identifying fire-colored pixels in photographs. However, many objects in real-world scenarios might have similar colours, so if we rely only on the colour information, false detections might happen. Therefore, additional measures are required to get rid of things that aren't fire-colored.

In addition to colour, mobility is a crucial characteristic for identifying fire. In surveillance applications, moving objects are frequently segmented out of a scene using background subtraction. Several methods [2]-[4], [6], [11]-[13] treat fire as a moving object under the premise that the appearance of fire will alter the background. As a result, these methods start the segmentation of flames with background subtraction. The majority of the articles in the state-of-the-art literature employ background-based or frame difference approaches to identify nonstationary pixels. Combining the previously identified colour and motion results, candidate fire pixels are discovered. These techniques, however, cannot differentiate moving things that are coloured like fire from actual fire.

As a result, more processes are necessary to precisely extract flames from a video sequence.

Flickers are another common characteristic for spotting fire in literature. It is simple to see how fire's brightness fluctuates erratically over time. The author of [12] presented a 1-D wavelet transform for the investigation of temporal colour fluctuation based on this characteristic. Wavelet signals make it simple to see a signal's random nature, which is a fundamental quality of flame pixels. Similar to this, [1] described an algorithm that identifies fire using the flicker characteristic. Due to the tendency of fire to frequently flicker over a region, this method analyses the cumulative time derivative of luminance, providing the strongest values to the areas of the fire that flicker.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The author of [4] calculated the flame flickering frequency and found that it was around 10 Hz; hence, they used a hidden Markov model (HMM) to confirm the potential flame pixels' frequencies to see if they were roughly 10 Hz.

It was possible to extract the normalised red skewness, LH wavelet coefficient skewness, HL wavelet coefficient skewness, and HH wavelet coefficient skewness in Ref. [13] by using candidate flame pixels. The probability that there is fire in the present frame was then determined using Bayesian networks.

The position and area, wavelet data, border and flickering frequency, as well as the temporal and spatial fluctuation of the intensity, have all been extracted from candidate pixels using different techniques. Based on these characteristics, probabilistic models are estimated, and heuristic criteria are used to discriminate between actual flames and other moving objects. Although these algorithms may have high true detection rates, they are still too high to satisfy the needs of security applications. A significant drawback of probabilistic models is that it might be challenging to select the best categorization criteria. While a strict threshold might result in many missed fires, a lenient threshold might cause many false-positives.

It is difficult to reduce erroneous fire detection, but more and more research is aiming to do so. Recently, various machine learning-based picture classification techniques have been created, and these techniques may be an effective way to tell genuine fires apart from non-fires.

Deep learning is a subset of machine learning that achieves great power and flexibility by learning to represent the world as a nested hierarchy of concepts, where each concept is defined in relation to simpler concepts and more abstract representations are computed in terms of less abstract representations. Using its hidden layer architecture, a deep learning technique defines low-level categories and learns them incrementally. Compared to conventional machine learning algorithms, which use little to no feature engineering, deep learning promises to provide more accurate machine learning algorithms. CNN-based deep learning algorithms have become the most advanced method for classifying images and identifying objects. These methods have demonstrated exceptionally strong performance in picture classification and object recognition applications within computer vision. Numerous studies demonstrate that CNNs are excellent at classifying images, and that using deep learning algorithms to classify fire may be able to minimise false alarms. Recently, there has been a movement towards applying CNNs to increase the accuracy of fire detection, and some of these systems have had significant success. In Ref. [19], an imbalanced dataset was created to study the issue of fire detection in the actual world. The failure of several cutting-edge CNN models is attributed, according to the author, to fire, an uncommon occurrence that occurs under real-world circumstances.

To solve this issue, they suggested a more complex CNN model, which showed promising results in spotting fire in videos. The author of [20] used a CNN image classifier [14] at the final layer of a cascade classification model to identify moving objects that resemble fire from actual flames. Compared to earlier research techniques, the accuracy of CNN's picture categorization is outstanding. However, CNNs ignore the temporal properties of fire flames and are only effective at modelling an object's 2D texture. In many instances, human eyes are unable to discern between many things from a single image; instead, we must examine an object's dynamic texture over time in order to draw a firm judgement. Following the CNN model, Ref. [21] used the extreme learning machine classifier to detect fire in videos. Their approach outperforms cutting-edge deep CNNs in fire detection accuracy while still processing data quickly. In [22], the author attempted to use cutting-edge CNN object detection models including YOLO, Faster-RCNN, R-FCN, and SSD to directly detect fire. These methods, however, are likewise based on 2D textures and do not take temporal information into account. Additionally, the amorphous nature of fire flames makes it challenging to create training datasets for deep learning fire flame detection models. Referencing [12], [23] also proposed a randomness test model to confirm the dynamic textures of fire flames, and [12] proposed a 1-D wavelet transform for temporal colour variation analysis of fire flames. The main drawback of these probabilistic approaches is that it might be challenging to choose the best decision threshold, and customised analytical features demand a high level of knowledge.

By looping the LSTM units, which each consist of a memory cell and three multiplicative components (an input gate, an output gate, and a forget gate), an LSTM network [24, 25] is created. Instead of using only one neural network layer like the RNN unit, each memory cell (or LSTM unit) employs four. In addition, the cell state from the previous cell is controlled using the output of one neural network layer, where a sigmoid function is applied. The past cell state is "forgotten" if the sigmoid function's output is zero since it has a range of 0 to 1 values. If not, it builds up in the cell after the regulating factor multiplies it. The LSTM may retain information for a long time when this manipulation technique is used. Recurrent neural networks (RNNs) and long short-term memory (LSTM) networks have achieved great success when processing sequential multimedia data and have produced state-of-the-art results in speech recognition, digital signal processing, video processing, and text data analysis. CNNs and handcrafted feature probabilistic models, however, have limitations when modelling dynamic textures.

In [26], CNN-LSTM was employed to analyse X-ray pictures in order to identify the new coronavirus (COVID-19). In order to get the final results, the LSTM model is fed with the possible features that were extracted using the CNN. They claim that their approach can identify the appearance of COVID-19 with a sensitivity of 99.3% and accuracy of 99.4%, respectively. Similar to [27], the author there suggested using a CNN and an RNN to address the issue of COVID-19 detection from X-ray pictures. They employed an RNN to extract temporal information and a pretrained VGG-19 model as the framework for extracting spatial features. Additionally, they employed Grad-CAM (gradient-weighted class activation mapping) to identify the area that causes COVID-19 in X-ray images. The author of [28] used a 3D CNN with an LSTM in order to tackle the foreground segmentation problem. An encoder-decoder was used to represent the foreground-background segmentation issue. The findings demonstrate that their suggested algorithm performed competitively when measured against state-of-the-art techniques in terms of the figure of merit. Amin Ullah suggested a model that combines a CNN and LSTM to analyse the dynamic textures of picture sequences [24], drawing inspiration from LSTM's effectiveness for sequential data processing. For real-time action identification in video, this model significantly improves upon both CNN and LSTM network advancements While LSTM performs sequence analysis to forecast human activities, Each image in a sequence can have its own CNN acting as a feature extractor.

The CNN-LSTM combo is effective at identifying human activity, making it a good choice for examining the dynamic characteristics of fire flames. Numerous studies have successfully used this model to improve the precision of fire flame detection in the literature. Before adding LSTM layers with a thick layer on the output, the author of [29] added CNN layers on the front end. In order to distinguish between fire and non-fire in videos, the CNN layer serves as a feature extractor, while LSTM serves as a video classifier. Similar to [30], the author suggested a CNN-LSTM model for classifying fire picture sequences. The author employs optical flow to convert raw photos into motion images before inputting the images to CNNs for feature extraction, in contrast to [29] who enters raw images straight into the CNN input layer. Both of the aforementioned algorithms have produced positive results, but because there is no pre-processing step to localise the position of fire flames, they are only appropriate for large fires where the fire flames make up a significant portion of the image. When the flame is little and just covers a small portion of the picture, the prediction's accuracy may suffer.

III.FIRE DETECTION

The Figure 6 provides an illustration of the proposed fire flame detection architecture. This strategy mostly consists of three parts. The colour and motion properties will be used in the first stage to detect and localise candidate fire flames, which are almost certainly real fires. Our algorithm is capable of identifying numerous fire possibilities in an image and following them in subsequent frames. This method can guarantee that the movie has no missing footage of actual fire. The following step involves feeding a cropped image of each candidate from the sequence of images into the CNN model, which will then use CNN layers to convert the input image into visual features. Finally, the LSTM model receives the image sequence's extracted fire flame features and outputs a prediction.

A. Candidate Detection of Fire Regions

We aim to localise the candidate fire regions, or the areas with a high likelihood of apparent fire flames, in the first step. At this stage, the only factors employed to identify fire flames are colour and mobility. A distinctive characteristic of fire flames is their colouring, which typically ranges from yellow to red to white, depending on the temperature of the fire. RGB colour information is typically translated into a mathematical space that separates brightness (or luminance) information from colour information in order to imitate the colour sensing characteristics of the human visual system. The HSV (Hue, Saturation, Value of Intensity) colour model among these is appropriate for offering a more human-oriented way to describe colours. HSV is therefore particularly useful for colour analysis.

The classification of pixels with a fire colour is shown in the image. We can see that numerous pixels in the background image have been misclassified as being fire-colored. Therefore, colour alone is insufficient to detect fire, and additional post-processing steps are required to get rid of stationary fire-colored background image pixels. In addition to colour, flame flickering is a helpful characteristic that can be utilised to enhance the classification of fire pixels. Flickering causes both the intensity of the flame pixels and the surrounding pixels to fluctuate erratically over time. The fire pixel flicker frequency has been targeted by numerous prior attempts, but it is particularly challenging to measure.

$$HT1 \le H \le HT2 \quad (1)$$

$$R > RT \quad (2)$$

$$S > (255 - R) \times ST\,RT \quad (3)$$

When an item is moving, its time derivative is nonzero and zero for stationary regions. Thus, a moving object can be followed using the time derivative for the video images. If an object rotates a region periodically, the sum of the absolute values of the derivatives rises. The flickering of the fire in a fire scene causes a permanent rise in the pixel values close to the fire region.

$$E(t) = I(t) - I(t-1), \quad (4)$$
$$A(t) = \alpha E(t-1) + (1-\alpha)E(t), \quad (5)$$

B. Fire Candidate Classification Using CNN-LSTM

According to analysis, earlier fire prediction techniques that relied solely on a single image are insufficient, and in order to make decisions, we must see items one after the other. We employ a CNN-LSTM model for classifying genuine fire from non-fire situations. The CNN-LSTM architecture employs LSTM to help with sequence prediction and CNN layers to extract features from input data, as seen in the illustration.

For the purpose of extracting visual features, CNNs will be used to process the sequence photos from each possible fire location. Following that, a many-to-one multilayer LSTM network that temporarily fuses this extracted information is fed these independent features. The CNN-LSTM model's architecture for classifying fires and non-fires is shown in the image. To improve network performance, we employ a multilayer LSTM model. Two LSTM layers are included in the architecture, one of which moves in the forward direction and the other in the reverse direction. Then, using the hidden states of both layers, their combined output is calculated. To reach a judgement, the softmax classifier is used in the LSTM's final state.

For the purpose of extracting visual features, CNNs will be used to process the sequence photos from each possible fire location. Following that, a many-to-one multilayer LSTM network that temporarily fuses this extracted information is fed these independent features. The CNN-LSTM model's architecture for classifying fires and non-fires is shown in Figure 6. To improve network performance, we employ a multilayer LSTM model. Two LSTM layers are included in the architecture, one of which moves in the forward direction and the other in the reverse direction. Then, using the hidden states of both layers, their combined output is calculated. To reach a judgement, the softmax classifier is used in the LSTM's final state.

Equations (10, 11, and 12) show how to do this.

$$f_t = \sigma(W_f [h_{t-1}, x_t] + b_f), \quad (6)$$
$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \quad (7)$$
$$\hat{c}_t = \tanh(W_C[h_{t-1}, x_t] + b_C), \quad (8)$$
$$C_t = f_t \times C_{t-1} + i_t \times \hat{c}_t, \quad (9)$$
$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \quad (10)$$
$$h_t = o_t \times \tanh(C_t), \quad (11)$$
$$\text{Fire} = \text{softmax}(h_t). \quad (12)$$

The number of layers in the neural network models has been increased, which has improved the performance of the deep neural network. In a similar manner, we stacked two LSTM layers in our network to increase system accuracy. The design of the two-layer LSTM network used in our techniques is depicted in image. The input to layer 2 comes from its time and the output of the current time step of layer one, while layer 1 derives input through CNN feature extraction. Additionally, we employ a bidirectional LSTM model, in which the output at time t depends both on the frames that will come after it and on the frames that came before it. Two LSTMs are piled on top of one another in the bidirectional LSTM that is employed in this application.

C. CNN and LSTM Model Architecture

Our base model was built by combining the CNN and LSTM layers. We experimented with Conv3d and (CNN and LSTM), and since we have a small dataset, we need to use transfer learning in order to achieve high accuracy. In order to feed the data into CNN-LSTM model, we used (time Distribution methods) on each item in our model, which had the shape (1x256x256x3). For each group of tensors, the Time Distribution operation does the same operation. In the base model, a collection of tensors with the shape [frames, h, w, colours] is made up of 40 consecutive frames, each of which is represented by a single tensor. Each video (a group of tensors) enters the LSTM frame by frame, each with the shape of [h, w, colours]. The model applies the same weight and the same computation for that group of tensors; however, the calculations are updated anytime a new group is received. Our LSTM layer is made up of 32 Cells, meaning that we are attempting to learn a time relation between 32-time steps. Each Cell represents a time step, each time step is a frame, and the 32-time step is made up of 32 consecutive frames.

The output of the time distributed model is a 2d tensor, which is fed into the layer. We utilise the global max pooling instead of the FNN (feedforward neural network) since a prior study [14] demonstrates that the global max pooling is more resistant to spatial translations. We take the whole sequence prediction from the LSTM [8] units rather than the last prediction.

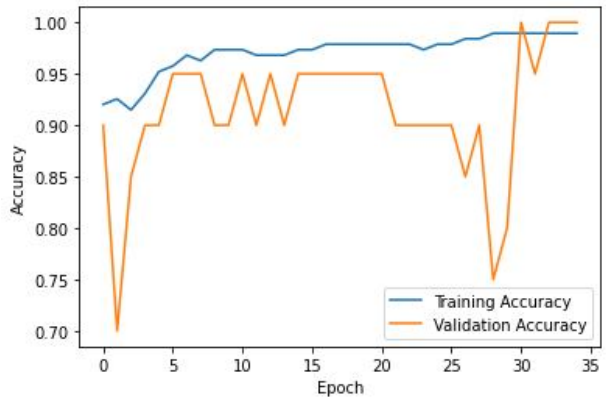


Figure 4. Validation vs training accuracy graph for the proposed model

Also, Fig (5) shows the loss of the model both train (blue line) and test (red line) where x-axis represent the loss and y-axis represent the epoch number.

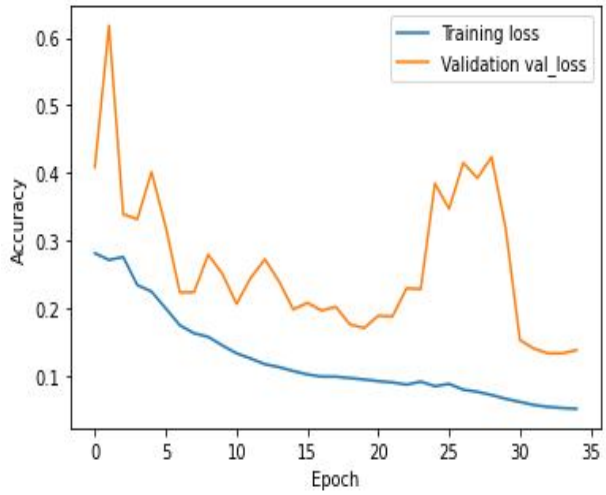


Figure 5. Validation vs training loss graph for the proposed model

Additionally, we used the Adam optimizer with a learning rate of 0.001, monitored the test loss to retain only the best model, and decreased learning rate by a factor of 0.5 when the test loss was not decreasing.

The proposed method's general steps are as follows:

The following steps are used to read a sequence of frames in a 4d tensor (frame, H, W, RGB); apply pre-train CNN for each frame; group the results from the previous step; and flatten the tensor into a 2d form (frames, SP), where SP is (H*W*RGB) and represents a spatial feature vector for one frame.

1) Use the feature vector generated from the previous phase as the input for the LSTM, where SP stands for the input and Frame for the time step. For instance, for a 32-frame input, we have (SP1, SP2,.. SP40), each of which goes into a time step of the LSTM. To obtain the outcome as a 1d tensor, take the global average of the output from the preceding step.

2) Feed the results of the previous phase into the output layer, which is a dense layer with sigmoid activation and reflects the likelihood that a fire would occur in the provided video.

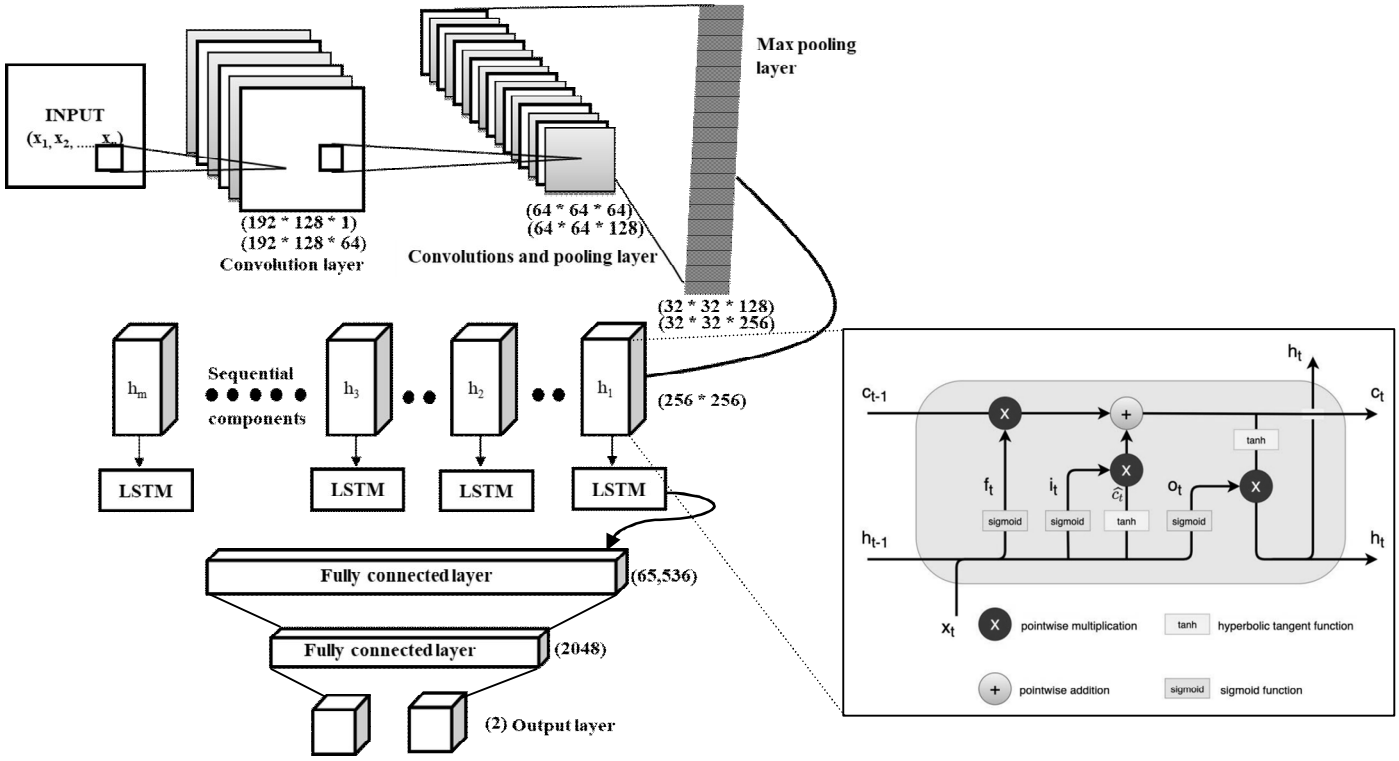


Figure 6. Architecture of the CNN-LSTM model

IV. RESULTS AND DISCUSSION

For the gathered data set, the proposed system has a base test set accuracy of 97.47%. The accuracy of the model is shown in Fig. 4 as both the train (blue line) and test (orange line) lines, with the accuracy represented on the x-axis and the epoch number on the y-axis. The results are as shown in the figure.7. which shows various outputs for minor and major fires.

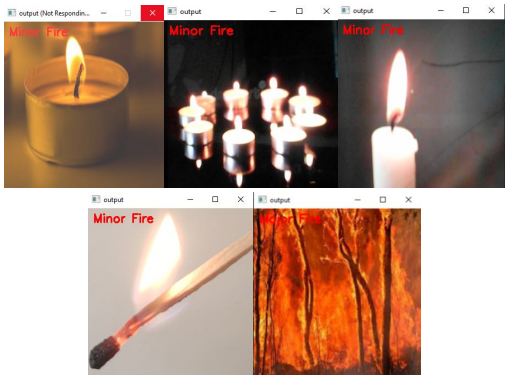


Figure. 7. Results for the fire input images given

V. CONCLUSION AND FUTURE WORK

Automated fire detection is crucial because early fire warnings increase escape chances and lessen fire damage. In order to be used in an early fire alarm system, this proposed work offers an image datasets-based fire detection algorithm.

To analyze fire flames in both the spatial and temporal domains, this technique combines an LSTM network with a CNN. Our approach significantly outperformed the available techniques by utilizing the benefits of both CNNs and LSTM in computer vision applications.

Our algorithm is quick, dependable, and suitable for application in real-time surveillance systems, according to the trial results.

In addition, the following benefits of our suggested approach can be emphasized:

1) Because the suggested model has no thresholds, it can be used in a variety of meteorological scenarios.
2) The suggested model has a low rate of false alarms and can reach high accuracy.
3) Our suggested method adds a fire candidate extraction stage so that our system can detect varied sized fires, in contrast to previous systems that apply fire classifiers to whole images, leading to missed detections of small flames. The CNN-LSTM model's processing of small, clipped fire pictures as input allows our solution to be quick as well.
4) We gathered the fire video dataset from two main sources: fire films that were crawled from the internet and fire videos that were captured during the real-world application of our suggested method in a variety of weather circumstances.

Despite the promising outcomes, our suggested method still has a number of drawbacks, including the instability of the fundamental features used to extract candidates for non-fire objects, and the absence of fire image data. Future research will see the following improvements to the suggested approach:

a) During the deployment of the suggested system in the various real-world contexts, we will gather more data, particularly for non-fire objects.
b) To remove the instability of the current fundamental features, we will integrate a fire candidate segmentation model
c) By altering the annotation of fire flames, we will test the end-to-end model training approach.
d) To increase the capacity to identify fires early, we will look at integrating smoke detection.

VI. ACKNOWLEDGMENT

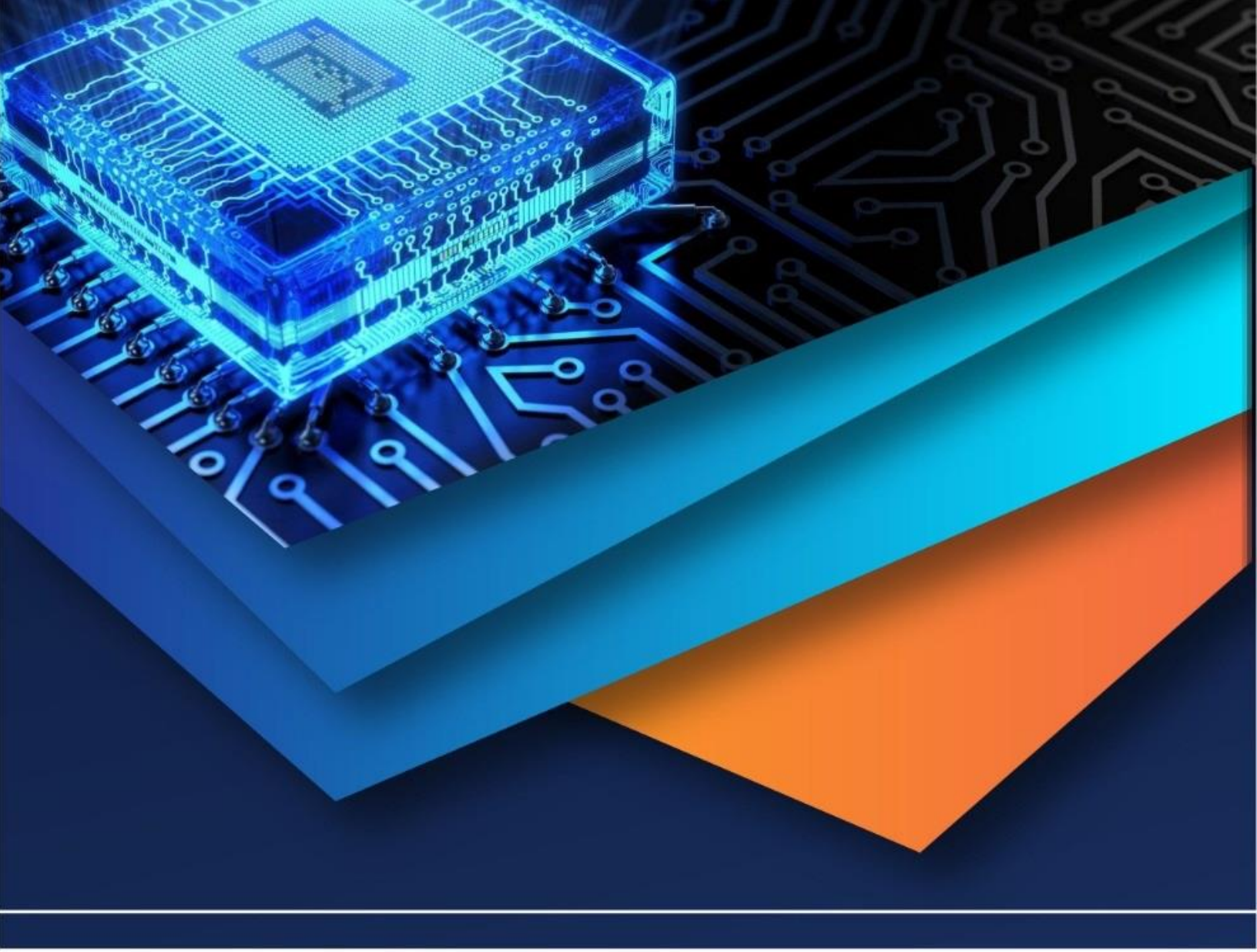
Any achievement does not depend solely on the individual efforts but on the guidance, encouragement and co-operation of intellectuals, elders and friends. We extend our sincere thanks to Dr. Kamalakshi Naganna, Professor and Head, Department of Computer Science and Engineering, Sapthagiri College of Engineering, and Prof. Shwetha A B, Assistant Professor, Department of Computer Science and Engineering, Sapthagiri College of Engineering, for constant support, advice and regular assistance throughout the work. Finally, we thank our parents and friends for their moral support.

REFERENCES

[1] X. Qi and J. Ebert, ‘‘A computer vision-based method for fire detection in color videos,’’ Int. J. Imag., vol. 2, no. S09, pp. 22–34, 2009.
[2] P.-H. Huang, J.-Y. Su, Z.-M. Lu, and J.-S. Pan, ‘‘A fire-alarming method based on video processing,’’ in Proc. Int. Conf. Intell. Inf. Hiding Multimedia, Dec. 2006, pp. 359–364.
[3] J. Chen, Y. He, and J. Wang, ‘‘Multi-feature fusion based fast video flame detection,’’ Building Environ., vol. 45, no. 5, pp. 1113–1122, May 2010.
[4] Z. Teng, J.-H. Kim, and D.-J. Kang, ‘‘Fire detection based on hidden Markov models,’’ Int. J. Control, Autom. Syst., vol. 8, no. 4, pp. 822–830, Aug. 2010.
[5] B. C. Ko, K.-H. Cheong, and J.-Y. Nam, ‘‘Fire detection based on vision sensor and support vector machines,’’ Fire Saf. J., vol. 44, no. 3, pp. 322–329, Apr. 2009.
[6] B. U. Töreyin, ‘‘Fire detection algorithms using multimodal signal and image analysis,’’ Ph.D. dissertation, Dept. Elect. Electron. Eng., Bilkent Univ., Ankara, Turkey, 2009.
[7] A. E. Gunawaardena, R. M. M. Ruwanthika, and A. G. B. P. Jayasekara, ‘‘Computer vision based fire alarming system,’’ in Proc. Moratuwa Eng. Res. Conf. (MERCon), Apr. 2016, pp. 325–330.
[8] I. Chakraborty and T. K. Paul, ‘‘A hybrid clustering algorithm for fire detection in video and analysis with color based thresholding method,’’ in Proc. Int. Conf. Adv. Comput. Eng., Jun. 2010, pp. 277–280.
[9] T. Çelik and H. Demirel, ‘‘Fire detection in video sequences using a generic color model,’’ Fire Saf. J., vol. 44, no. 2, pp. 147–158, Feb. 2009.
[10] G. Marbach, M. Loepfe, and T. Brupbacher, ‘‘An image processing technique for fire detection in video images,’’ Fire Saf. J., vol. 41, no. 4, pp. 285–289, Jun. 2006.
[11] T. Celik, ‘‘Fast and efficient method for fire detection using image processing,’’ ETRI J., vol. 32, no. 6, pp. 881–890, Dec. 2010.
[12] 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, no. 1, pp. 49–58, Jan. 2006.
[13] B. Ko, K.-H. Cheong, and J.-Y. Nam, ‘‘Early fire detection algorithm based on irregular patterns of flames and hierarchical Bayesian networks,’’ Fire Saf. J., vol. 45, no. 4, pp. 262–270, Jun. 2010.
[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ‘‘ImageNet classification with deep convolutional neural networks,’’ in Proc. Adv. Neural Inf. Process. Syst. (NIPS), vol. 25. Stateline, NV, USA, Dec. 2012, pp. 1097–1105.
[15] K. He, X. Zhang, S. Ren, and J. Sun, ‘‘Deep residual learning for image recognition,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.
[16] A. K. Reyes, J. C. Caicedo, and J. E. Camargo, ‘‘Fine-tuning deep convolutional networks for plant recognition,’’ CLEF Working Notes, vol. 1391, pp. 467–475, Sep. 2015.
[17] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, ‘‘Learning and transferring mid-level image representations using convolutional neural networks,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 1717–1724.

[18] Y. Bengio, ‘‘Deep learning of representations for unsupervised and transfer learning,’’ in Proc. ICML Workshop Unsupervised Transf. Learn., 2012, pp. 17–36.
[19] J. Sharma, O.-C. Granmo, M. Goodwin, and J. T. Fidje, ‘‘Deep convolutional neural networks for fire detection in images,’’ in Proc. Int. Conf. Eng. Appl. Neural Netw. Cham, Switzerland: Springer, 2017, pp. 183–193.
[20] N. M. Dung and S. Ro, ‘‘Algorithm for fire detection using a camera surveillance system,’’ in Proc. Int. Conf. Image Graph. Process. (ICIGP), 2018, pp. 38–42.
[21] J. Sharma, O.-C. Granmo, and M. Goodwin, ‘‘Deep CNN-ELM hybrid models for fire detection in images,’’ in Proc. Int. Conf. Artif. Neural Netw. Cham, Switzerland: Springer, 2018, pp. 245–259.
[22] P. Li and W. Zhao, ‘‘Image fire detection algorithms based on convolutional neural networks,’’ Case Stud. Thermal Eng., vol. 19, Jun. 2020, Art. no. 100625.
[23] D. Jin, S. Li, and H. Kim, ‘‘Robust fire detection using logistic regression and randomness testing for real-time video surveillance,’’ in Proc. IEEE 10th Conf. Ind. Electron. Appl. (ICIEA), Jun. 2015, pp. 608–613.
[24] 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.
[25] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, T. Darrell, and K. Saenko, ‘‘Long-term recurrent convolutional networks for visual recognition and description,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 2625–2634.
[26] M. Z. Islam, M. M. Islam, and A. Asraf, ‘‘A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images,’’ Informat. Med. Unlocked, vol. 20, Jan. 2020, Art. no. 100412.
[27] M. S. Al-Rakhami, M. M. Islam, M. Z. Islam, A. Asraf, A. H. Sodhro, and W. Ding, ‘‘Diagnosis of COVID-19 from X-rays using combined CNN-RNN architecture with transfer learning,’’ MedRxiv, Aug. 2020.
[28] T. Akilan, Q. J. Wu, A. Safaei, J. Huo, and Y. Yang, ‘‘A 3D CNN-LSTM-based image-to-image foreground segmentation,’’ IEEE Trans. Intell. Transp. Syst., vol. 21, no. 3, pp. 959–971, Mar. 2020.
[29] X. T. Pham, H. V. Nguyen, and C. N. Le, ‘‘Real-time long short-term glance-based fire detection using CNN-LSTM neural network,’’ Int. J. Intell. Inf. Database Syst., vol. 1, no. 1, p. 1, 2021.
[30] C. Hu, P. Tang, W. Jin, Z. He, and W. Li, ‘‘Real-time fire detec tion based on deep convolutional long-recurrent networks and optical flow method,’’ in Proc. 37th Chin. Control Conf. (CCC), Jul. 2018, pp. 9061–9066.
[31] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, ‘‘ImageNet: A large-scale hierarchical image database,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)