



e-ISSN: 2319-8753 | p-ISSN: 2320-6710

# IJIRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 10, Issue 4, April 2021

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

**Impact Factor: 7.512**



9940 572 462



6381 907 438



ijirset@gmail.com



www.ijirset.com

# Video based Fire Detection Using Deep Learning

Arti Sawant<sup>1</sup>, Vaidehi Mane<sup>2</sup>, Akshaya Kadam<sup>3</sup>, Tanvi Shelar<sup>4</sup>, Mamta Pathare<sup>5</sup>

Assistant Professor, Dept of Electronics & Telecommunication Engineering, Vivekanand Education Society's Institute of Technology, Chembur, Mumbai, India<sup>1</sup>

Student, Dept of Electronics & Telecommunication Engineering, Vivekanand Education Society's Institute of Technology, Chembur, Mumbai, India<sup>2345</sup>

**ABSTRACT:** As a new fire detection technology, image fire detection has recently played a crucial role in reducing fire losses by alarming users early through early fire detection. Image fire detection is based on an algorithmic analysis of images. However, there is a lower accuracy, delayed detection, and a large amount of computation in common detection algorithms, including manually and machine automatically extracting image features. Therefore, novel image fire detection algorithms based on the advanced object detection CNN models of Faster-RCNN and YOLO v5 are proposed in this paper.

**KEYWORDS:** deep learning; fire detection; *Faster Region-based Convolutional Neural Network (R-CNN)*; YOLO v5.

## I. INTRODUCTION

Many places are becoming in firetrouble/hazards where they can loose their belongings/properties and lives. To reduce that problem the system for fire detection and alert in case fire outbreak should be designed and implemented. Fires are one of the biggest challenges in the world right now, due to the global warming that the planet is currently suffering from. We all know what fires are and what they are capable of causing great damage, whether to humans, animals, or other forms of life. In general, two broad categories of technologies can be identified: traditional fire alarm and fire detection by computer vision. Traditional fire alarm technology is based on smoke or heatsensors that require proximity for activation. These sensors need human involvement to confirm a fire in case of alarm. Furthermore, such systems require various equipment to provide information on the size, location, and burning degree of the fire. To overcome these limitations, researchers have been investigating computer vision-based methods combined with various types of supplementarySensors. There are three mainstages in the process of image fire detection algorithms, including image preprocessing, feature extraction, and fire detection. Among, feature extraction is the core part in algorithms. In the deep learning approach however, these exploration and exploitation processes can be replaced by the training of an appropriate neural network with a sufficient amount of data in order to avoid over fitting. In the proposed method, the suspected region of fire (SRoF) is detected with its spatial features against non-fire objects by the Faster Region-based Convolutional Neural Network (R-CNN). The features summarized from the object detection model in successive frames are accumulated by YOLO v5 to classify whether there is a fire or not, which can be treated as a person's glance for fire detection. It also reducing the false detections and misdetections, and to successfully interpret the temporal behavior of flame and smoke, which possibly reduces the false dispatch of firemen.

## II. PROPOSED SYSTEM

We assumed that the DTA process can greatly reduce erroneous fire detection. The proposed method uses the Faster R-CNN fire detection model to detect the SRoF based on its spatial features.

### Steps involved:

The fire system detection makes used of the convolution neural network (CNN) that detect fire automatically. CNN method is one of the neural networks that mostly applied for image recognition and classification. The following points represent the basic steps of this system: -

- The first step is to train the classifier to determine accuracy. And use only a small number of images, which is approximately 1213.
- In the second step, convert the frame captured by the webcam directly to grayscale.
- The third step is to convert the image to grayscale, and there is only one channel, either black or white, to facilitate its processing.
- The fourth step in which we use the gradient factor is to create a scale pyramid, directly the image is trained, and this will allow the scale operator to shrink the frame to the fire detection this result.

The algorithm we used was how much training and testing accuracy we got is inaccurate. But after modifying the codes and rewriting them and deepening the search for references, my result of the algorithm was the amount of accuracy of training and testing completely different and the number of images used were very accurate results.

## 2.1 BLOCK DIAGRAM:

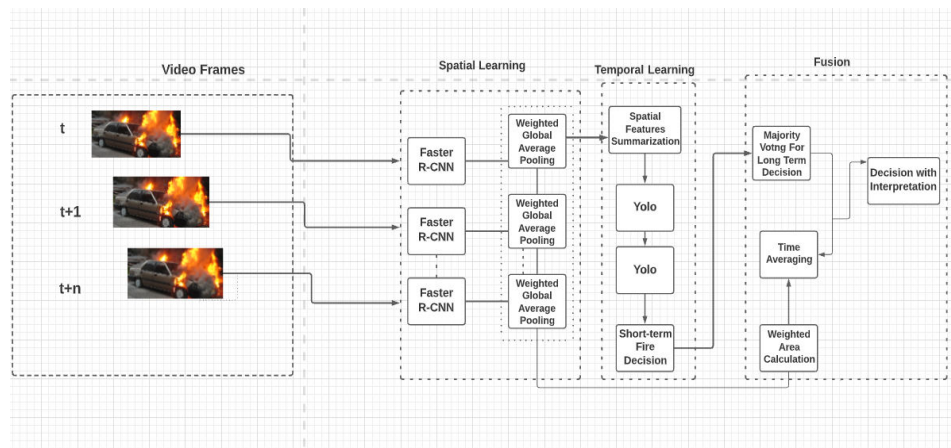


Fig 1: Fire detection algorithm based on CNN

- Initially we select a image or video.
- The selected image or video is converted into t numbers of frame
- This frames are passed further to faster R-CNN model. Then averaging pooling is done to the processed image.
- By the pooling process to replace fully connected layers into one feature map.
- The output of pooling is then processed by yolo in which the resultant output of R-CNN model and output of yolo is compared to get the exact location of fire.
- The last block also int greatest he information to interpret the dynamic fire behaviors in order to determine whether the area of SRoFs, including flame and smoke detected by bounding boxes at Faster R-CNN stages, is increasing or not
- The whole fire region is treated as one bounding box with their values.
- The output also get stored in our directory for future analysis.

## 2.2 FLOW CHART:

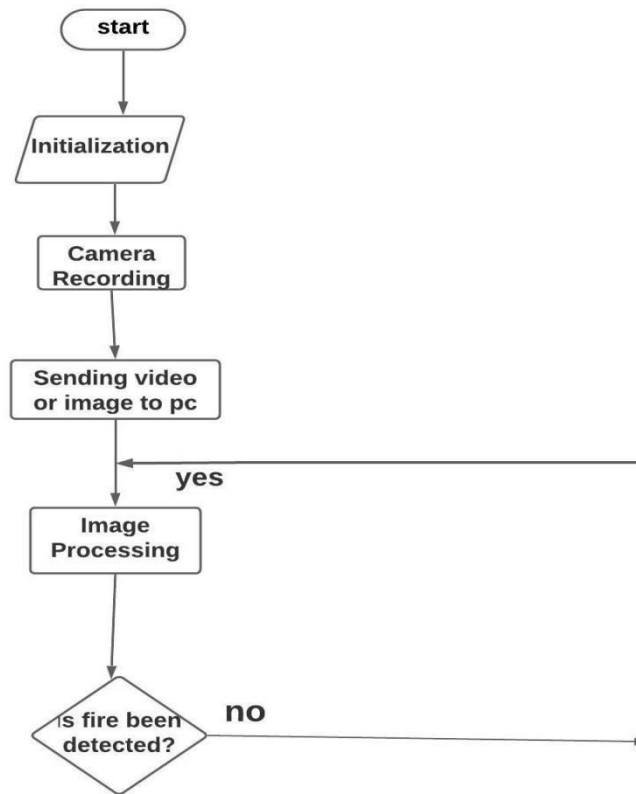


Fig 2: Flow chart of algorithm

In this we are working on downloaded dataset from kaggle and github.

- i. Initialing the process with selecting the random image or video file stored in our directory.
- ii. The selected picture is pre-processed for better detection and to avoid false detection.
- iii. The processed image is then passed to Faster R-CNN model for detection.
- iv. The output of faster R-CNN model is then processed by yolo in which the resultant output of R-CNN model and output of yolo is compared to get the exact location of fire.
- v. If the image contains a fire, the fire part is in bounding box with its value.
- vi. If the image does not contain fire, it will continue monitor the video or image.

## III. METHODOLOGY

We proposed a Deep Learning model for fire detection, to reduce false alarms. A GUI is created for real time video analysis and store all events monitored by system.

### A. Methods:

a. *Motion Detection:* Motion analysis acts an important role due to increase in demand vision based image interpretation. Movement in a video can be detected by motion detection. Analyzation of difference in images of video frames is done. An adaptive background subtraction method is used for detecting the sudden lightning conditions and to generate foreground mask. Adaptive background subtraction maintains background model and parameters of background model evolve over time.

b. *Data Augumentation:* Data augmentation is the technique of increasing the size of data used for training a model. For reliable predictions, the deep learning models often require a lot of training data, which is not always available.



Therefore, the existing data is augmented in order to make a better generalized model. Image augmentation can be applied as a pre-processing step before we train the model or can be applied in real time. Although data augmentation can be applied in various domains, it's commonly used in computer vision. Some of the most common data augmentation techniques used for images are:

- L **Flipping**: flipping the image vertically or horizontally
- L **Rotation**: rotates the image by a specified degree.
- L **Shearing**: shifts one part of the image like a parallelogram
- L **Cropping**: object appear in different positions in different proportions in the image
- L **Zoom in, Zoom out**
- L **Changing brightness or contrast**

#### B. Software used:

Python is becoming popular day by day, and is starting to replace many languages that are popular in this industry. The main reason behind the popularity of Python is because: - Python is popular for beginners because of its simplicity. - Python helps developers be more productive from development to deployment and maintenance.

a. *Tensorflow*: Tensorflow is just a computational framework for expressing algorithms that include a large number of tensor operations, where neural networks can be expressed as mathematical graphs that can be executed.

b. *Keras*: Keras provides an easier way to express neural networks. It also provides some utilities for manipulating data sets, aggregation models, evaluating results, visualizing graphs, and many more. Keras internally uses either Tensorflow or Theano as background. It provides several pre-prepared datasets and pre-processing models like Inception, SqueezeNet, VGG, ResNet, and others.

c. *ReLU*: ReLU stands for rectified linear activation unit and is considered one of the few milestones in the deep learning revolution. ReLU activation function formula:  $f(x) = \max(0, x)$ . ReLU function is its derivative both are monotonic. The function returns 0 if it receives any negative input, but for any positive value  $x$ , it returns that value back. Thus it gives an output that has a range from 0 to infinity. The ReLU function is simple and it consists of no heavy computation as there is no complicated math. The model can, therefore, take less time to train or run.

## IV. EXPERIMENTAL RESULTS

#### Fire image dataset:

Training the algorithms based on CNNs needs a large number of data. We have divided our dataset into two parts i.e Training and validation. However, current small-scale image/video fire databases cannot meet the needs, so we have collected the data from Kaggle, Github and google. Therefore 2,180 images are collected from small public fire image/video databases, large public images/video data sets, previous experimental data from research institutions. Images in the data set obtained from different sources vary in size.



Fig 3: Dataset

**Training of dataset:**

A training dataset is comprised of many rows of data. Epoch defines the number times that the learning algorithm will work through the entire training dataset. One epoch means that each sample in the training dataset has had an opportunity to update the internal model parameters. An epoch is comprised of one or more batches. It is common to create line plots that show epoch along x-axis as time and the error or skill of model on the y-axis. The number of epoch is traditionally large, often hundreds or thousands, allowing the learning algorithm to run until the error from the model has been sufficiently minimized. The number epochs is the number of complete passes through the training dataset. So, we completed 50 epochs in 15 batch size during training process.

```
Found 980 images belonging to 2 classes.
Found 239 images belonging to 2 classes.
Epoch 1/15
5/5 [=====] - 114s 22s/step - loss: 16.2777 - acc: 0.6007 - val_loss: 4.8254 - val_acc: 0.3071
Epoch 2/15
5/5 [=====] - 105s 22s/step - loss: 1.9982 - acc: 0.7000 - val_loss: 0.1471 - val_acc: 0.9286
Epoch 3/15
5/5 [=====] - 110s 22s/step - loss: 0.2673 - acc: 0.9257 - val_loss: 0.1386 - val_acc: 0.9357
Epoch 4/15
5/5 [=====] - 106s 21s/step - loss: 0.1906 - acc: 0.9433 - val_loss: 0.3362 - val_acc: 0.8786
Epoch 5/15
5/5 [=====] - 105s 20s/step - loss: 0.3261 - acc: 0.8883 - val_loss: 0.1322 - val_acc: 0.9500
Epoch 6/15
5/5 [=====] - 106s 21s/step - loss: 0.2480 - acc: 0.9165 - val_loss: 0.6262 - val_acc: 0.6929
Epoch 7/15
5/5 [=====] - 104s 22s/step - loss: 0.4262 - acc: 0.8180 - val_loss: 0.1191 - val_acc: 0.9714
Epoch 8/15
5/5 [=====] - 109s 22s/step - loss: 0.1431 - acc: 0.9580 - val_loss: 0.2385 - val_acc: 0.8714
Epoch 9/15
5/5 [=====] - 105s 21s/step - loss: 0.2734 - acc: 0.8796 - val_loss: 0.1817 - val_acc: 0.9000
Epoch 10/15
5/5 [=====] - 104s 20s/step - loss: 0.1657 - acc: 0.9265 - val_loss: 0.2151 - val_acc: 0.8571
Epoch 11/15
5/5 [=====] - 105s 20s/step - loss: 0.2593 - acc: 0.8639 - val_loss: 0.1751 - val_acc: 0.9143
Epoch 12/15
5/5 [=====] - 105s 21s/step - loss: 0.1361 - acc: 0.9535 - val_loss: 0.3636 - val_acc: 0.8071
```


Fig 4: Epoch

The working of proposed algorithms for fire detection was verified by Indoor & Outdoor images be taken from our dataset. Figure (6) shows the 4 image frames taken out from our dataset. If fire is detected it forms a bounding box around the fire and smoke as well with its probability, to indicate if it's a fire or any object with same colour combination as fire.

## Real Time Fire Detection

- ☐ Detect fire in a video
- ☒ Detect fire in an image

Upload an image



Drag and drop file here  
Limit 200MB per file • PNG, JPG

Browse files

Fig 5: GUI Setup



Fig 6: Fire Detected in images with Bounding box

The proposed algorithm is also tested on videoframes, it shows the bounding box around the fire shown in Figure (7) and continuously monitor the video.







Fig 7: Fire detected in videos with bounding box

## V. CONCLUSION

We have been able to test the code in our nearby surroundings and have successfully inferred the results which are quite accurate. To improve the performance of image fire detection technology and reduce false fire detection issue, the advanced object detection CNNs of Faster-RCNN and YOLO v3 are used to develop algorithms of image fire detection. The proposed algorithms can automatically extract complex image fire features and successfully detect fire in different scenes. The highest accurate algorithm based on YOLO v3, with 93.7% accuracy, detects fire the most quickly and is the strongest robust.

## REFERENCES

- [1] M. V. D. Prasad, G. Jaya Sree, K. Gnanendra, P. V. V. Kishore and D. Anil Kumar "FIRE DETECTION USING COMPUTER VISION MODELS IN SURVEILLANCE VIDEOS" VOL. 12, NO. 19, OCTOBER 2017.
- [2] Daniel Y. T. Chino, Letricia P. S. Avalhais, Jose F. Rodrigues Jr., Agma J. M. Traina "Detection of Fire in Still Images by Integrating Pixel Color and Texture Analysis" 10 june 2015
- [3] Salim Said AL-Ghadani, C. Jayakumari "Innovating Fire Detection System Fire using Artificial Intelligence by Image Processing", Volume-9 Issue-11, September 2020
- [4] Faming Gong, 1 Chuantao Li, 1 Wenjuan Gong "A Real-Time Fire Detection Method from Video with Multifeature Fusion" 14 July 2019
- [5] Byoungjun kin "A video based fire detection using deep learning" 18 july 2019.
- [6] Sidhant Goyal, MD Shagill, Arwinder Kaur, Harpreet Vohra, Ashima Singh "A YOLO based Technique for Early Forest Fire Detection",
- [7] VOL 9, April 2020.
- [8] Wangda zhoa "Image fire detection algorithm based on convolutional neural network", VOL 19, June 2020.
- [9] K. Muhammad, J. Ahmad, I. Mehmood, et al. "Convolutional neural networks based fire detection in surveillance videos IEEE Access", 6 (2018)





**Impact Factor:  
7.512**



# **INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH**

**IN SCIENCE | ENGINEERING | TECHNOLOGY**

**9940 572 462 6381 907 438 ijirset@gmail.com**



**www.ijirset.com**

Scan to save the contact details