International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 01 | January - 2023 Impact Factor: 7.185 ISSN: 2582-3930

FOREST FIRE ANALYSIS USING MACHINE LEARNING

CHIRAG VARSHNEY
Department of IT, Maharaja Agrasen Institute of Technology, GGSIP University
Delhi, India

TUSHAR
Department of IT, Maharaja Agrasen Institute of Technology, GGSIP University
Delhi, India

Dr.Amita Goel
Maharaja Agrasen Institute of Technology

Er.Nidhi Sengar
Maharaja Agrasen Institute of Technology

Dr. Vasudha Bahl
Maharaja Agrasen Institute of Technology

1.) ABSTRACT

Forest fires have become a major threat worldwide, causing many negative impacts on human habitats and forest ecosystems. Climate change and the greenhouse effect are some of the consequences of this destruction. Interestingly, the rate of forest fires occurring due to human activities is higher. Therefore, in order to minimize the devastation caused by forest fires, it is necessary to detect forest fires at an early stage. This paper proposes a system and method that can be used to detect forest fires at an early stage using a CNN model. Detecting smoke and fire is a difficult task, because variations in color and texture are so whimsical. Many smoke and fire image classification methods have been proposed to overcome this problem, but most of them are rule-based methods where accuracy is lower or homemade methods are expensive to produce. In this paper, we propose a new fire detection system using convolutional neural network. Fire detection using machine learning. It can be extremely difficult to detect fire and smoke with sensors already installed in buildings. They are slow and unprofitable due to their rudimentary design and technology. To solve this problem caused by training the network on a limited dataset, we improve the number of training images available using traditional techniques of data usage and enhancement.

2.) INTRODUCTION

Fire is cause of many hazards in this world and yearly many people die of fire and toxins from smoke.
Although many buildings and vehicles already have fire prevention and fire protection systems due to the high number in the fire incidents but due to inaccuracy and old technology these systems fail sometimes. Forests are important in nature as it helps in balancing the ecosystem. Thus, when a fire occurs in forest it causes huge damage to not only wildlife but whole ecology. As, most of the time when system fails to predict a forest fire it causes huge damage. The emission of large amount of carbon dioxide. from the forest fire is harmful for the environment. Also causes the imbalance in nature and increase global warming and kills a lot of flora and fauna. Thus, even can make certain plant and animals go extinct.
Thus, detecting fire at early stage is necessary as it helps in reducing the damage. Several methods and tools have been using to recognize fire or smoke in visual scenes. Most of the traditional fire and smoke detection methods use sensor-based tools or handcrafted tools. The main problem of these methods is that not only they are used on old technology but also, they are installed on a certain area. If a fire started on polar opposite of system installed then it cannot detect the fire thus causing huge loses.

3.) LITERATURE SURVERY

Torien [1] was the first one which introduced video-based detection algorithm. First, he used a hybrid background to estimate moving pixels and regions. Thus a 3 framed difference operation is performed to determine region of motion, after that using adaptive background subtraction to extract the entire moving region. Then to detect fire-colored pixels, he used gaussian mixture model in the RGB color space and to analyze the flame flicker, temporal wavelet transform is performed.at last, fire mask pixels of color variation in spatial wavelet analysis if moving region is performed. Significant spatial variations presuppose fire region.

Celik [2] developed two models: one for fire detection and the other for smoke detection. Instead of older models, a rule based fuzzy model was used. This choice made the classification more robust in effectively discriminating fire and fire like colored objects. For smoke, a statistical analysis was carried out on the basic idea of color of smoke is gray.

Borges [3] for bayes classifier used a multidimensional feature as input which were generally the boundary roughness od the potential fire regions, skewness which is the third order statistical moment of the potential fire regions, variance, and finally the amount of fire from frame to frame (which varies due to flame flickering).

Verstockt[4] for both visual and non-visual flame of moving object he introduced a multisensory fire detector. He uses ordinary video and long wave infrared thermal images. First to extract moving objects, he operates dynamic background subtraction, also these LWIR moving objects are usually filtered by histogram based hot objects segmentation. This the distinctive geometric, temporal and spatial disorder characteristics of the flame regions are analyzing with focus. Then LWIR flame probability is calculated combining bounding box disorder, the principal orientation disorder and the histogram roughness of the hot moving objects in LWIR. Similar operation done on the ordinary videos to get flame probability.

Yuanbin Wang [5] for fire detection the color of the image from camera is highly important. And as the size of forest are very large it is not possible to watch all the forest at the same time thus resulting in the difficulties in detection the fire. Thus, with help of Convolutional Neural Network (CNN) technology it would be easier to avoids the blindness and accurate level of fire identification.to access the attributes of the fire occurs or not it uses a support vector mechanism for image classification. Thus, a fire can be detected be analyzing the color of the flame in picture.

Using the number, locate the fire
plotted pixels in a picture in accordance with The color and intensity of fire can be measured. the flames. Therefore, it ought to be simpler to Find the source of the fire and put it out. This approach needs to be more efficient and trustworthy in locating the fire. The precision be much superior to the other
methods.

Xingkui Zhu[6] The dataset is preprocessed and fed into not just one but two individual object detectors, to obtain more accuracy than a single object detector, YOLOv5 and Efficient, two separate object detectors, integrated in parallel mode. This does not consider the entire image even if it uses integrated object detectors. To address this issue, a new classifier is introduced. In order to fully utilize the information, Efficient Net examines the image as a whole. A decision strategy algorithm that weighs the opinions of the three different object detectors will determine the outcomes, improving model performance and lowering the incidence of false positives. According to this research, they have successfully balanced average accuracy, average recall, false positives, and latency.

Goodfellow, Ian [7] Our research includes a fresh dataset with 2440 photos of fire and smoke. We construct a dataset of fire and smoke photographs that may be used to train our network and test our method because, as far as we are aware, there is no publicly available dataset for smoke or fire image recognition. Given that CNN models need a large number of images to train properly, we employ a variety of data augmentation techniques to extract additional training instances from the few original images that make up our data benchmark. We create fresh training images by combining conventional image alteration techniques with generative adversarial networks (GANs) [10], which improves the performance of our network and addresses the overfitting issue.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 01 | January - 2023 Impact Factor: 7.185 ISSN: 2582-3930

4.) TECHNOLOGY USED

Our project uses the preexisting technology of CNN (convolutional layers) and data augmentation to give high accuracy of fire and smoke detection. It also uses different functions to improve the accuracy as compared to other systems.

4.1) CNN (CONVOLUTIONAL NEURAL NETWORK)

Machine learning includes convolutional neural networks, also known as convnets or CNNs. It is a subset of the several artificial neural network models that are employed for diverse purposes and data sets. A CNN is a particular type of network design for deep learning algorithms that is utilized for tasks like image recognition and pixel data processing. Although there are different kinds of neural networks in deep learning, CNNs are the preferred network architecture for identifying and recognizing objects. The structure of a CNN is comparable to the connection structure of the human brain. Similar to how the brain has billions of neurons, CNNs also have neurons, but they are structured differently. A convolutional layer, a pooling layer, and a fully connected (FC) layer make up a deep learning CNN. The first layer is the convolutional layer, while the final layer is the FC layer.

Convolutional layer. The convolutional layer, the central component of a CNN, is where most computations take place. The first convolutional layer may be followed by a subsequent convolutional layer. A kernel or filter inside this layer moves over the image's receptive fields during the convolution process to determine whether a feature is present.

Pooling layer. The pooling layer similarly to the convolutional layer sweeps a kernel or filter across the input image. Contrary to the convolutional layer, the pooling layer has fewer input parameters but also causes some information to be lost. Positively, this layer simplifies the CNN and increases its effectiveness.

Fully connected layer. Based on the features extracted in the preceding layers, picture categorization in the CNN takes place in the FC layer. Fully connected in this context means that every activation unit or node of the subsequent layer is connected to input or last layer.

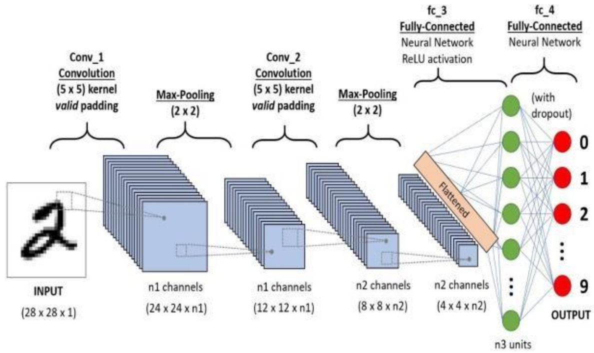

FIG 1. CNN

4.2) DATA AUGMENTATION

The most crucial component of the project is data augmentation, which will raise the overall accuracy of the image that will be used in the project. In data augmentation, we enhance the total number of images that can be used in neural networks by altering, flipping, or rotating existing photos. As a result, a single photograph provides us with numerous images to which we may apply our algorithm to obtain accurate fire or smoke detection. As a result, data augmentation is the process of creating new data points from current data in order to artificially increase the amount of data. In order to amplify the dataset, this may involve making small adjustments to the data or utilizing machine learning models to produce new data points in the latent space of the original data.

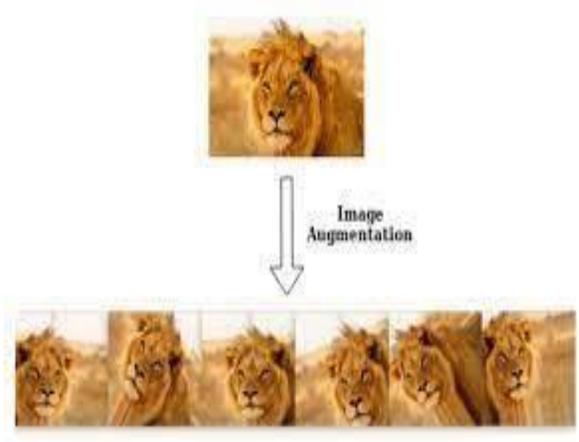

FIG 2.

DATA AUGMENTED IMAGE

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 01 | January - 2023 Impact Factor: 7.185 ISSN: 2582-3930

5.) PROPOSED RESEARCH

5.1) DATASET ACQUISITON

Getting the dataset is the most crucial and vital step in constructing this project. Currently, there are a number of datasets available online for both fire and smoke, and we used one from Kaggle.
More than 1000 photographs of fire and smoke are included in this dataset, which will assist to improve and increase the accuracy of this research. We have gathered this dataset, which will be enhanced, split into training and test sets, and fed to neural networks to construct the system.

FIG 3. TEST IMAGES

5.1.1) Data Preprocessing

This is the subsequent stage of system preparation. In this, data is cleaned, processed, or otherwise made to be as useful as possible. In order to prevent the system from producing undesirable outcomes, it involves deleting unwanted noises, images, or objects from the frame.

5.1.2) Data Augmentation

Even with a dataset of 1000 photographs of fire and smoke, the system may still be slightly inaccurate because there may occasionally be identical images taken from different angles that the algorithm doesn't deem to be meaningful. Thus, data augmentation is carried out to improve the system. In order to improve the system's ability to detect fire and smoke, more photos from different perspectives and with distinct color variances are added to the collection.

FIG 4.DATA AUGMENTED IMAGES

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 01 | January - 2023 Impact Factor: 7.185 ISSN: 2582-3930

FIG 5. DATA AUGMENTED IMAGES

5.1.3) Traditional Image Transformations

Generating new images using traditional image transformations includes making color and geometric changes such as changing the color palette, or cropping and translating the image by some degrees. As the color plays a major role in deciding if image is accurately correct or not.
Thus, in this project we have taken similar image from 4 different angles and 2 different shades to make our output better.

5.2) ARCHITECTURE OF CNN

Architecture of a project is like skeleton of the project to get the final product and conclusion.

Making a correct architecture for project is necessary as better the architecture more accurate the project will be,

Thus, my own CNN sequential model's architecture has: -

1 convolution layer with a filter size of 32, the same padding, and the Rectified Linear Unit as the activation function. The input size is 224x224x3.

1 x Maxpool layer with a pool size of 2x2 and a stride of 2x2.
A 64-filter convolutional layer with the same padding and activation will be used (Rectified Linear Unit)
1 x Maxpool layer with a pool size of 2x2 and a stride of 2x2.
1 x 256-filter convolution layer with identical padding and activation functions will be used (Rectified Linear Unit)

1 x Maxpool layer with a pool size of 2x2 and a stride of 2x2
In essence, the relu activation function is applied to each layer to prevent the transmission of any negative values to the layer below.
I flatten the data into vectors that result from the convolutions after implementing all the convolution layers, and then I transfer the data to the dense layer.
Flatten one layer.
1 × Dense layer with 128 units with relu activation (Rectified Linear Unit)
A sigmoid activation function will be used for the 1-unit Dense Softmax layer. When there is a binary classification issue, the sigmoid activation function is applied.

```
cnn.summary()
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 224, 224, 32)      896

 max_pooling2d (MaxPooling2D  (None, 112, 112, 32)     0
 )

 conv2d_1 (Conv2D)           (None, 112, 112, 64)      18496

 max_pooling2d_1 (MaxPooling  (None, 56, 56, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 56, 56, 256)       147712

 max_pooling2d_2 (MaxPooling  (None, 28, 28, 256)      0
 2D)

 flatten (Flatten)           (None, 200704)            0

 dense (Dense)               (None, 128)               25690240

 dense_1 (Dense)             (None, 1)                 129

=================================================================
Total params: 25,857,473
Trainable params: 25,857,473
Non-trainable params: 0
```

FIG 6. CNN MODEL CODE

5.2.1) FEATURE ENGINEERING: -

For system to identify the fire it needs to know the feature of fire and smoke, how it looks to computer. As it is easily seen by human eye it is not as easy for the computer. Thus, addition of these features to help the computer in identifying the images is known as feature engineering. Fire emits reddish color; it has a shape under different circumstances and motion depending on the fuel it uses to burn. Thus, in this project, we extract the different features of fire and smoke and give them to the training set. The neural network extracts these features using feature extraction network in the CNN which is powered by a custom algorithm. After this image are categorized as either fire or non-fire scenarios. These are extracted using found boxes using image descriptors.

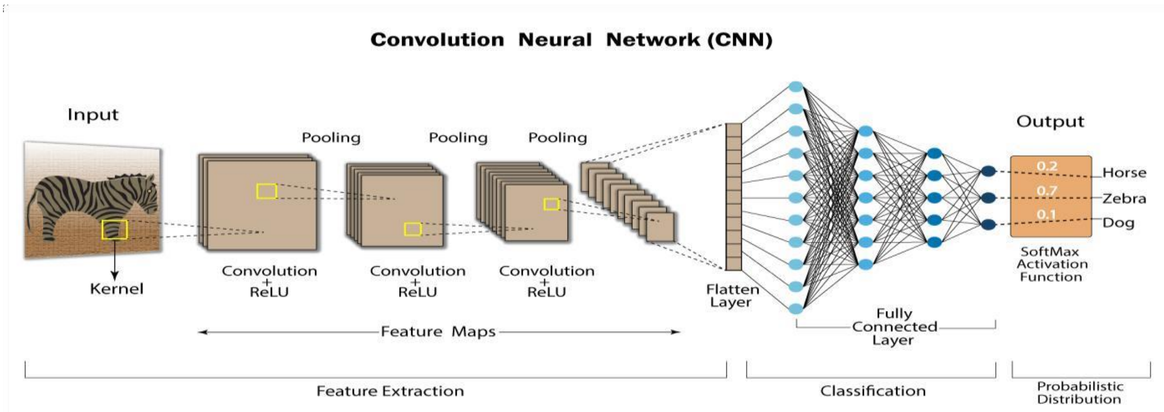

FIG 7. CNN IMAGE

5.3) MODEL BUILDING

All experiments are performed using the Keras software package on the Windows 11 operating system, running on a PC with Intel(R) Core (TM) i7-7700HQ CPU 2.80 GHz with an Nvidia GTX 1050 Ti GPU. For comparative purposes, we also implement some of the other leading deep CNNs using TensorFlow software package and done using Jupyter Notebook ide.

1.) In the first step we installed all the necessary functions like tensorflow and keras on jupyter notebook and downloaded fire and smoke dataset on the computer.

2.) Then we did data preprocessing and data augmentation on the ide by the algorithm present on the dataset. After the algo used, our data was divided into 2 categories for test and training sets.

3.) Then we built CNN layers and fed data to it to get the result and finally plotted graphs and system help us find out if given image was fire/smoke or not

6.) ANALYSIS AND VALIDATION

After all system formation, CNN building and feeding the data system gave us accurate images of fire and smoke out of dataset.

```
test_image=image.load_img(image_for_testing,target_size=(224,224))
test_image=image.img_to_array(test_image)
test_image=test_image/255
test_image=np.expand_dims(test_image,axis=0)
result=cnn.predict(test_image)
classes_x=np.argmax(result,axis=1)


Catagories=['Fire','Smoke']

image_show=PIL.Image.open(image_for_testing)
plt.imshow(image_show)

plt.title(Catagories[int(result[0][0])])
plt.show()
1/1 [==============================] - 0s 28ms/step
```

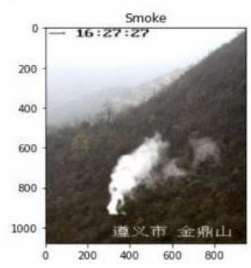

FIG 8. SMOKE

```
result=cnn.predict(test_image)
classes_x=np.argmax(result,axis=1)


Catagories=['Fire','Smoke']

image_show=PIL.Image.open(image_for_testing)
plt.imshow(image_show)

plt.title(Catagories[int(result[0][0])])
plt.show()
```

```
1/1 [==============================] - 0s 27ms/step
```

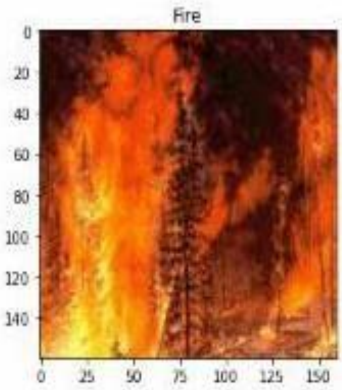

FIG 9. FIRE

```
test_image=test_image/255
test_image=np.expand_dims(test_image,axis=0)
result=cnn.predict(test_image)
classes_x=np.argmax(result,axis=1)


Catagories=['Fire','Smoke']

image_show=PIL.Image.open(image_for_testing)
plt.imshow(image_show)

plt.title(Catagories[int(result[0][0])])
plt.show()
```

```
1/1 [==============================] - 0s 125ms/ste
```

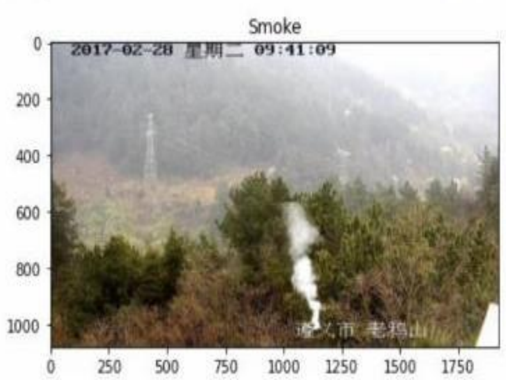

FIG 10. SMOKE RESULT

6.1) Accuracy

We made an epochs table to check the training and validation accuracy of our project.

Epochs	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
1	0.9800	0.1079	1.00	0.0086
2	0.9901	0.0409	0.9978	0.0095
3	0.9907	0.0345	0.9911	0.0192
4	0.9886	0.0452	0.9985	0.0109
5	0.9899	0.0350	0.9963	0.0103

Model accuracy and loss

```
plt.plot(history.history['accuracy'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train'],loc='upper left')
plt.show()
```

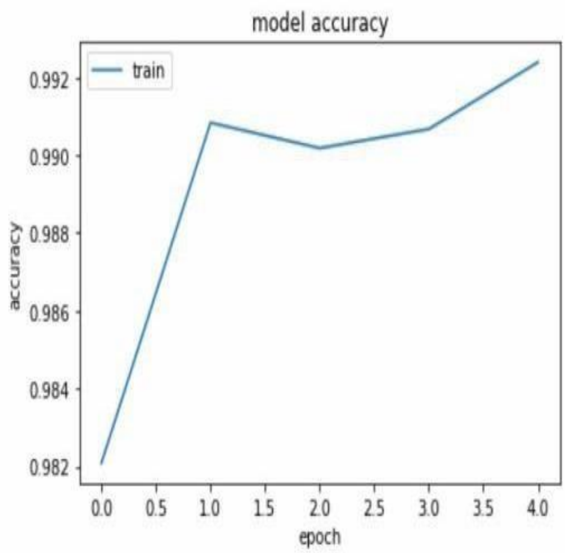

FIG 11. MODEL ACCURACY GRAPH

From this image we can see that our model accuracy is very high.

```
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('Training accuracy Vs validation accuracy')
plt.ylabel('Accuracy')
plt.xlabel('epoch')

plt.show()
```

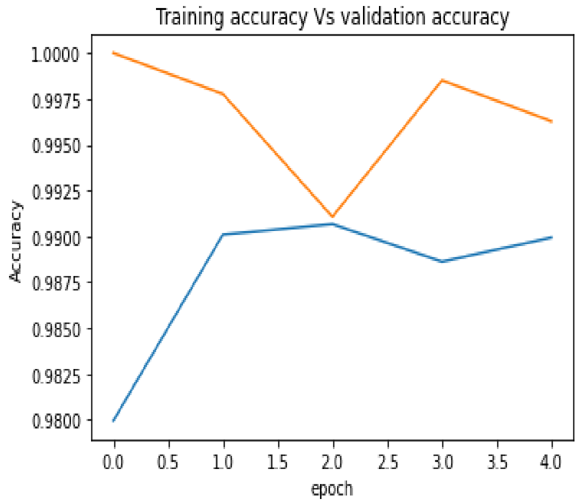

FIG 12. ACCURACY VS VALIDATION GRAPH

From this image we can see that our training and validation accuracy are very high and is accurate.

6.2) LOSS

As nothing is perfect and like every model our model also has some inaccuracies and loss but they are very minute and low as compared to other preexisting systems.

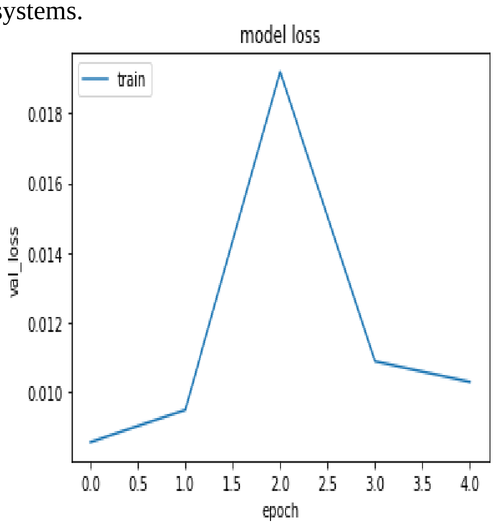

FIG 13. LOSS

```
plt.plot(history.history['val_loss'])
plt.plot(history.history['loss'])
plt.title('Training loss Vs Validation loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.show()
```

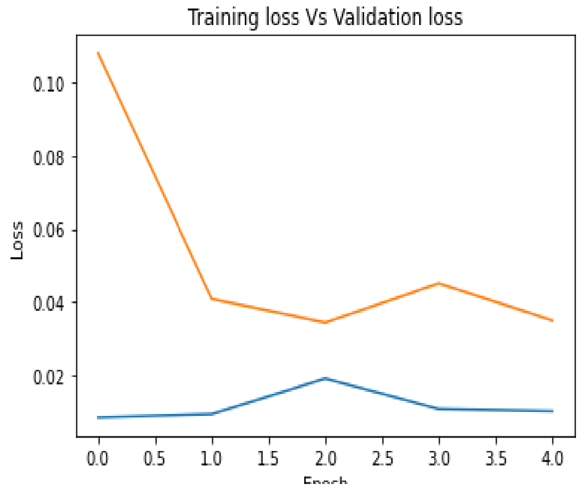

FIG 14. LOSS VS VALIDATION GRAPH

As we can see from the images that loss and inaccuracy are very low in our project.

7.) CONCLUSION

Since deep learning-based detection systems identify the combustible itself rather than its byproducts like smoke, heat, etc., they have overcome the limits of conventional smoke and heat detectors. Using deep CNNs, we shown in this research, it is possible to get very high classification performance, outperforming conventional systems since images from cameras may be used to detect fire and smoke directly. High resolution cameras extend the range, while high processing power CPUs with real-time image processing software shorten the detection time.

As Accuracy of our project is 0. 9967.It is very high accuracy as compared to other systems and done with use of high dataset and CNN technology.

Due to all these drawbacks in the existing algorithms, we came to the conclusion that CNN's give more accurate results.

8.) REFERENCES

[1] Töreyin, B. Uğur, Yiğithan Dedeoğlu, Uğur Güdükbay, and A. Enis Cetin. "Computer vision-based method for real-time fire and flame detection." Pattern recognition letters 27, no. 1, pp. 49-58, 2006. doi: 10.1016/j.patrec.2005.06.015

[2] T. Çelik, H. Özkaramanlı and H. Demirel, Fire and smoke detection without sensors: Image processing-based approach, Signal Processing Conference, 2007 15th European, Poznan, 2007, pp. 1794-1798.

[3] K. Borges, P. Vinicius, J. Mayer and E. Izquierdo, Efficient visual fire detection applied for video retrieval, Signal Processing Conference, 2008 16th European, IEEE, 2008.

[4] S. Verstockt, A. Vanoosthuyse, S. Van Hoecke, P. Lambert, and R. Van de Walle, Multi-sensor fire detection by fusing visual and non-visual flame features, In Proceedings of International Conference on Image and Signal Processing, June 2010, pp. 333 –341.

[5] Wang Y, Ren J. Image segmentation for forest fire in low illumination environment based on color feature. *Fire Sci Technol* 2017; 10: 75–78.

[6] Xingkui Zhu The dataset is preprocessed and fed into not just one but two individual object detectors, to obtain more accuracy than a single object detector.
Thu, 26 Aug 2021
cs arXiv:2108.11539

[7] Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." Advances in neural information processing systems, pp. 2672-2680. 2014