A

Major Project

On

EARLY FIRE DETECTION SYSTEM

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

S. Uma Maheshwari (177R1A05A7)

B. Jhansi (177R1A0565)

K. Mahathi (177R1A0583)

Under the Guidance of

K. Karunakar

(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)

Recognized Under Section 2(f) & 12(B) of the UGCAct.1956,

Kandlakoya (V), Medchal Road, Hyderabad-501401.

2017-21

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**EARLY FIRE DETECTION SYSTEM**" being submitted by **S. Uma Maheshwari (177R1A05A7), B. Jhansi (177R1A0565) & K. Mahathi (177R1A0583)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by him/her under our guidance and supervision during the year 2020-21.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Mr. K. Karunakar
Associate Professor
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HoD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on ________________

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. We take this opportunity to express my profound gratitude and deep regard to my guide

Mr. K. Karunakar, Assistant Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) Coordinators: **Mr. J. Narasimha Rao, Mr. B. P. Deepak Kumar, Mr. K. Murali, Dr. Suwarna Gothane and Mr. B. Ramji** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We would like to express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

S. Uma Maheshwari (177R1A05A4)

B. Jhansi (177R1A0565)

K. Mahathi (177R1A0583)

ABSTRACT

Fire disasters are man-made disasters, which cause ecological, social, and economic damage. To minimize these losses, early detection of fire and an autonomous response are important and helpful to disaster management systems. Therefore, in this article, we propose an early fire detection framework using fine-tuned convolutional neural networks for cameras, which can detect fire in varying indoor and outdoor environments. To ensure the autonomous response, we propose an adaptive prioritization mechanism for cameras in the surveillance system. Finally, we propose a dynamic channel selection algorithm for cameras based on cognitive radio networks, ensuring reliable data dissemination. Experimental results verify the higher accuracy of our fire detection scheme compared to state-of-the-art methods and validate the applicability of our framework for effective fire disaster management.

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1.INTRODUCTION

1 INTRODUCTION

1.1 PROJECT SCOPE

Emotion plays a significant role in protecting the safety of emergency response personnel. Property loss can be reduced and downtime for the operation minimized through early detection because control efforts are started while the fire is still small. Most alarm systems provide information to emergency responders on the location of the fire, speeding the process oof fire control.

1.2 PROJECT PURPOSE

Purpose of our Fire Detection Systems are designed to discover fires early
their development when time will still be available for safe evacuation of occupants.
To be useful, detectors must be couples with alarms. Alarm Sytems provide notice to at least the building occupants and usually transmit a signal on or off site. In some cases, alarms may go directly to the fire department, although in most locations this is no longer the typical approach.

1.3 PROJECT FEATURES

In our project we are using CNN models, and modules such as data preparation and fire detection. Data preparation Module analyzes the data, finds possible issues and suggests and applies corrections in a systematic way to the dataset. The fire detection module, the input image or video is processed through the convolution layer and pooling layer for detecting appropriately without producing any false results.

This early fire detection system has been made by using a dataset i.e.: Fire Detection System. Fire detection dataset consists of two classes. They are Non-Fire Images and Fire Images. Using smart cameras we can identify various suspicious incidents such as collisions, medical emergencies, and fires. And, using the great potentials of CNNs, we can detect fire from images or videos at an early stage.

2. SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

SYSTEM ANALYSIS

A fire detection system plays a pivotal role in our daily life. A fire detector is a device that detects fire and issues an alarm to alert nearby people of threat of a potential fire. It is imperative that smoke detectors regularly maintained and checked that they operate properly. This will ensure early warning to allow emergency responses to occur well before a fire causes serious damage. Early detection can save lives and help limit damage and downtime.

2.1 PROBLEM DEFINITION

Early fire detection system plays a crucial role in daily life. A fire detection system can limit the emission of toxic products created by the combustion, as well as global warming gases produced by the fire itself. These effects often overlooked, but undoubtedly occur in all fire fire scenarios. Therefore, reducing the likelihood of a fire is an important part.

2.2 EXISTING SYSTEM

Traditional fire detection technologies, like smoke and heat detectors, are not suitable for large spaces, complex buildings, or spaces with many disturbances.
Due to the limitations of above detection technologies, missed detections, false alarms, detection delays and other problems often occur, making it even more difficult to achieve early fire warnings.
However, many of the existing research has only been assessed on balanced datasets, which can lead to the unsatisfied results and mislead real-world performance as fire is a rare and abnormal real-life event.

2.2.1 LIMITATIONS OF EXISTING SYSTEM

- Missed Detections Due To Manual selection of fire feature
- False alarms, detection delays

2.3 PROPOSED SYSTEM

We have proposed a deep learning based fire detection method, which uses vision-based systems to detect fire using Convolutional Neural Networks during surveillance.

Our CNN model have been implemented for a cost-effective fire detection.

We are using fire detection dataset for training our model.

To balance the efficiency and accuracy, the model is fine-tunes considering the nature of the target problem and fire data.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- Accuracy is more
- Balanced Efficiency and Accuracy
- No More delay in detection
- Cost-effective architecture

2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement.

2.4.3 BEHAVIORAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

2.5 SOFTWARE REQUIREMENTS

Operating System	:	Windows/Linux.
Language used	:	Python 3
IDE	:	Anaconda, Jupyter notebook.

Libraries

1. **Numpy:**

Numpy is a general-purpose array-processing package. It provides a high performance Multi dimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.

Provides information about their positions within the image.

2. **Matplotlib:**

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

3.Python 3:

Python libraries play a vital role in developing machine learning, data science, data visualization, image and data manipulation applications and more.

4.Keras:

Keras is a high-level neural networks API, capable of running on top of Tensorflow, Theano, and CNTK. It enables fast experimentation through a high level, user-friendly, modular and extensible API. Keras can also be run on both CPU and GPU

5. Pytorch:

PyTorch is an open source machine learning library based on the Torch library, **used** for applications such as computer vision and natural language processing, primarily developed by Facebook's AI Research lab (FAIR). It is free and open-source software released under the Modified BSD license.

6. TorchFussion :

Based on PyTorch and fully compatible with pure PyTorch and other pytorch packages, **TorchFusion** provides a comprehensive extensible training inference with your PyTorch models, A GAN framework that greatly simplifies the process of experimenting with Generative Adversarial Networks Goodfellow et al. 2014, with concrete implementations of a number of GAN algorithms, and a number of high level network layers and utilities to help you be more productive in your work.

2. ARCHITECTURE

3.ARCHITECTURE

3.1 PROJECT ARCITECTURE

This project architecture shows the procedure followed for early fire detection using machine learning, starting from input to final prediction.

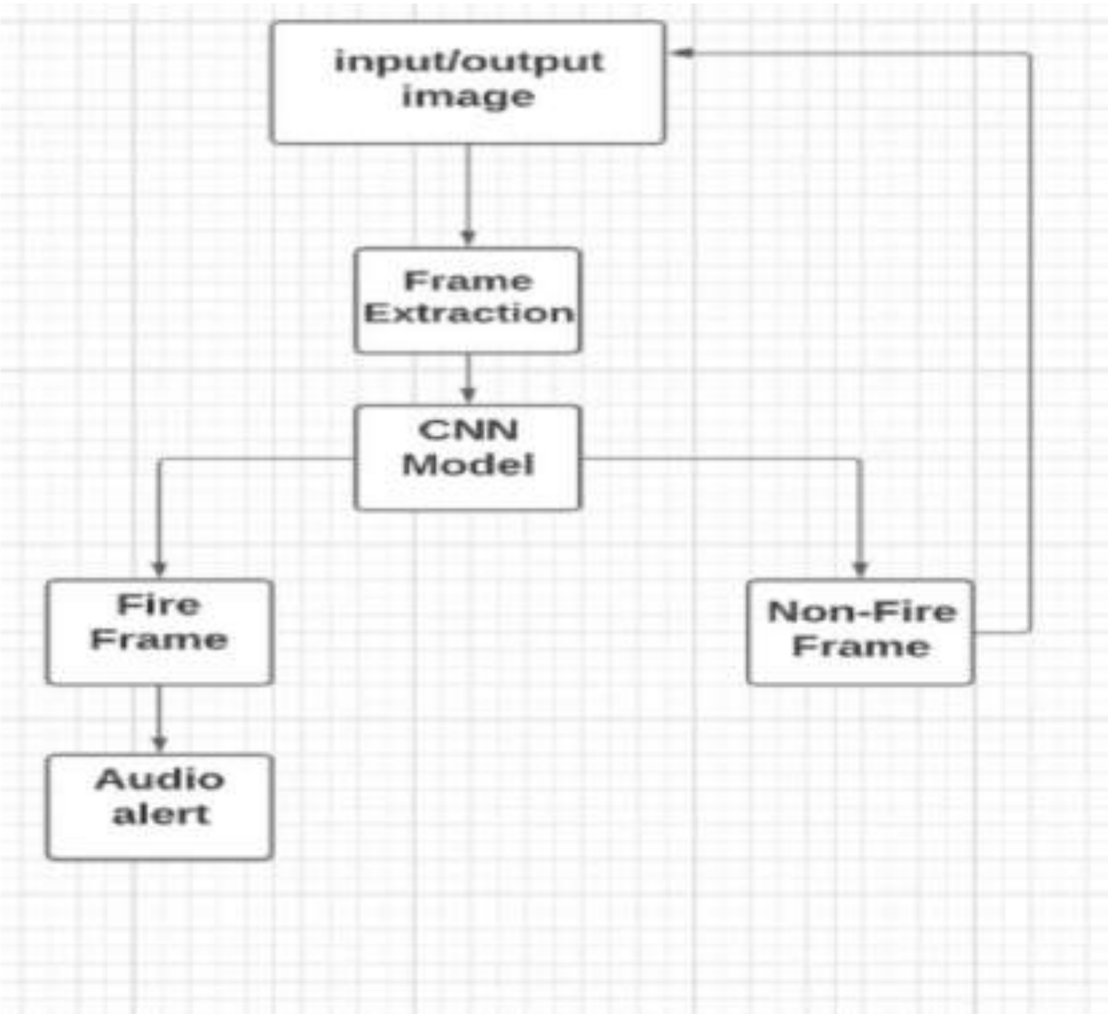

Figure 3.1: Project Architecture of Early Fire Detection

3.2 DESCRIPTION

By using CNN architecture, we are training our model and dataset. At the time of training a dataset, it takes the surveillance videos and gives the output by detecting the images and videos. Firstly, we will be given the input and output images. After, that we process the images and then extract the images. The sigmoid layer will classify the images as Fire frame and Non-Fire Frame.

3.3.CNN ARCHITECTURE

We have three stages in our CNN architecture. They are as follows 1. Image pre-processing, 2. Feature extraction, and 3. Fire detection.

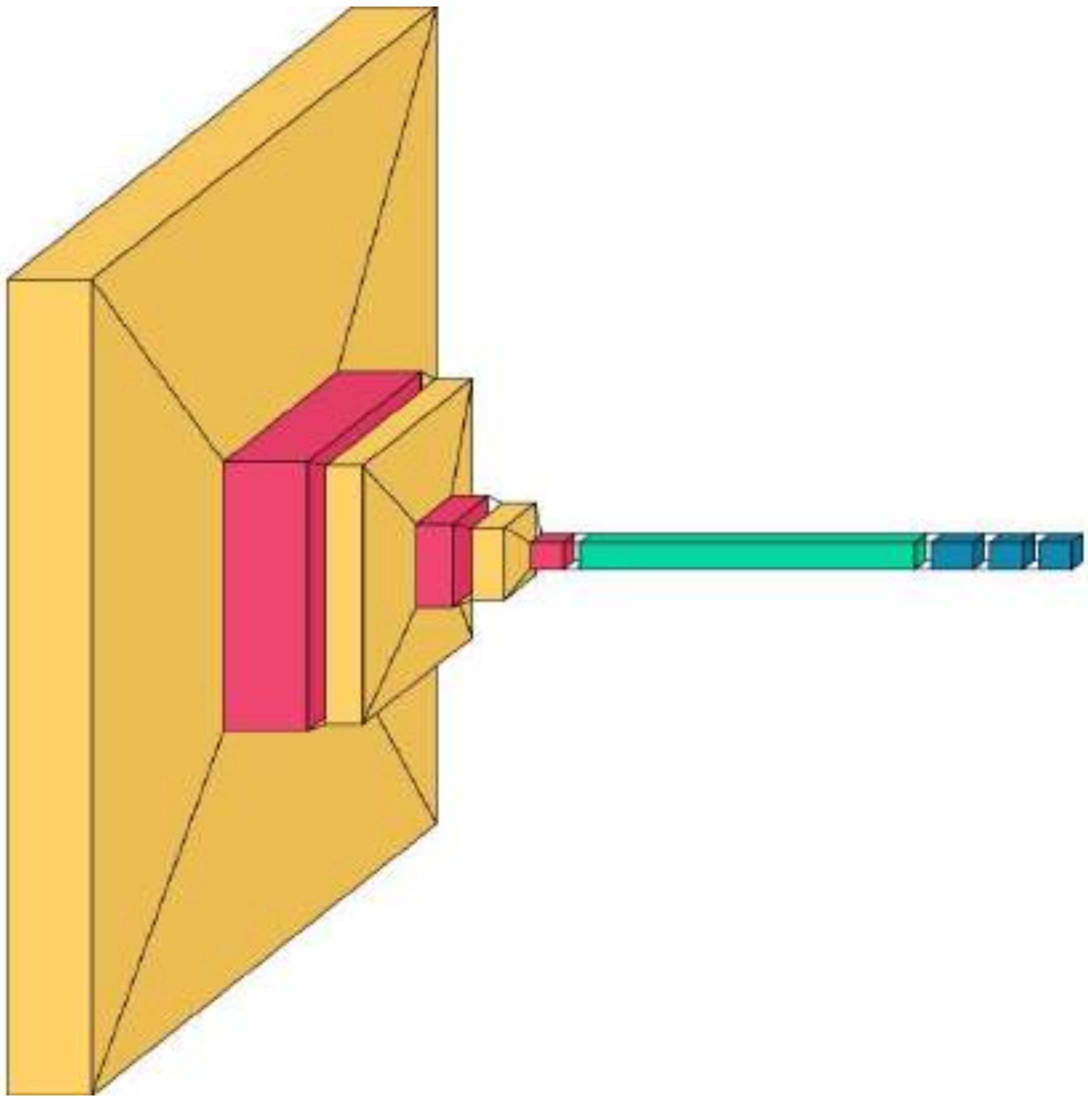

Fig:3.3 CNN Architecture

3.4 USE CASE DIAGRAM

Here we have two actors in this use case diagram, one is a user and the other is the model. The audio file to which the emotion has to be recognised is being made and saved by the user. This audio file is inputted to the model. The emotion of the speaker in the audio file is predicted and the output emotion is displayed. This emotion output can be viewed by the user and the new audio file given by the user is updated to the existing dataset by the admin

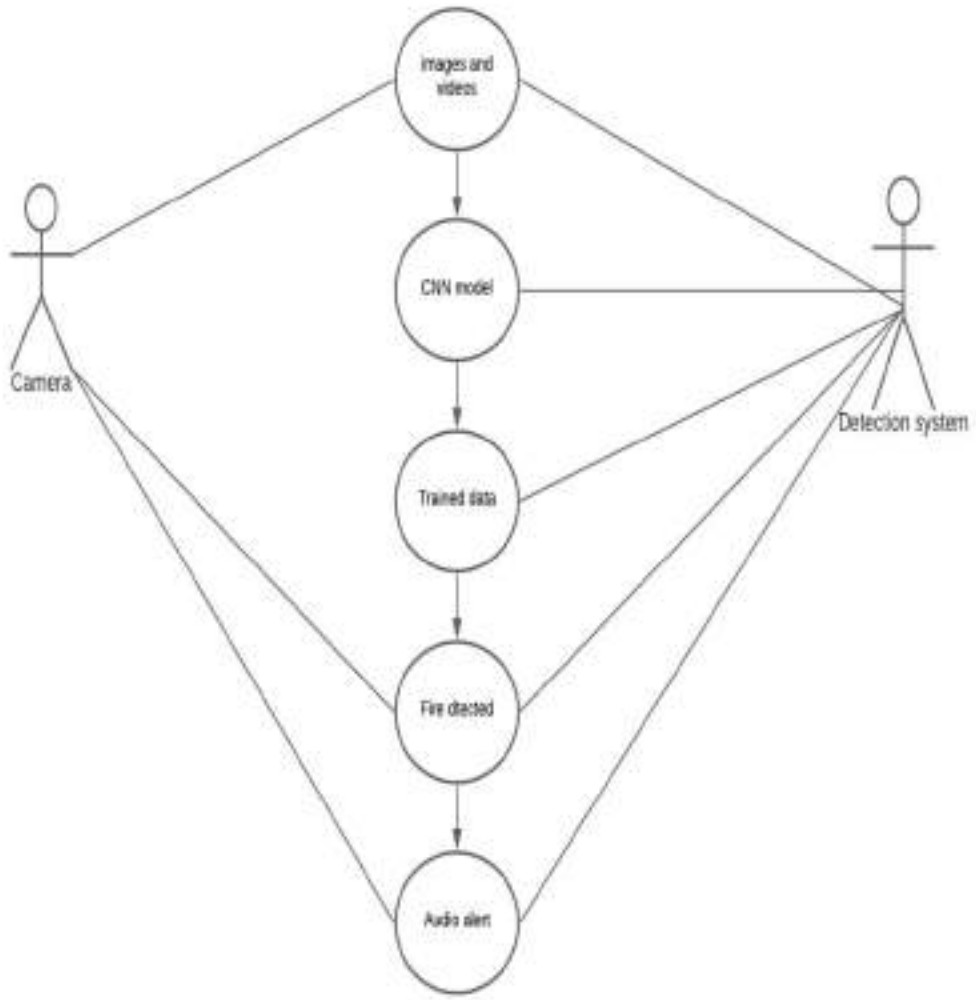

Fig:3.4 Usecase diagram for Early Fire detection System

3.5 CLASS DIAGRAM

A Class Diagram describes the structure of a system by showing it's classes, their attributes, methods, and the relations between them. A class represents an entity, a noun, of the system.

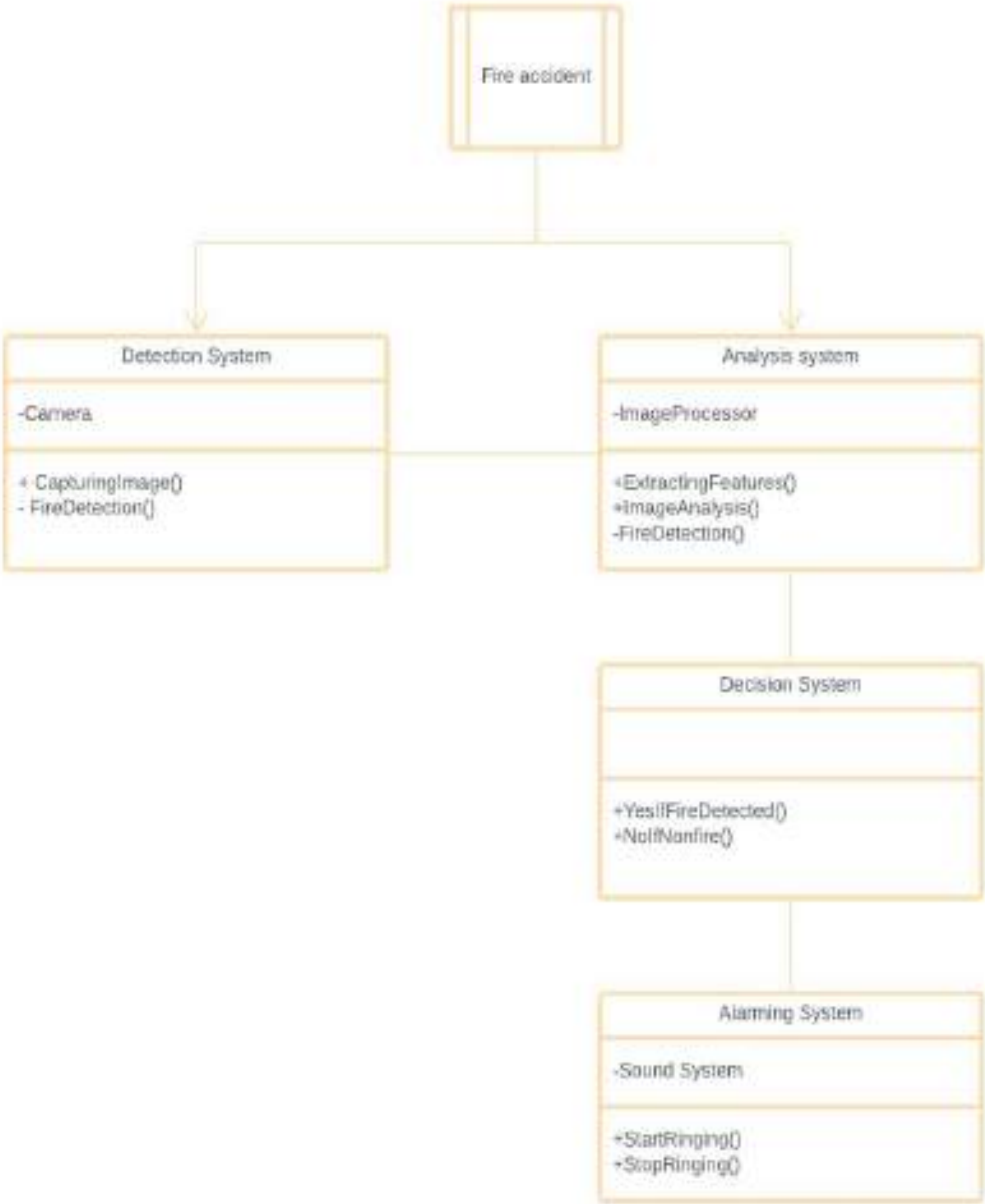

Fig: 3.5 Class Diagram For Early Fire Detection System

3.6 SEQUENCE DIAGRAM

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

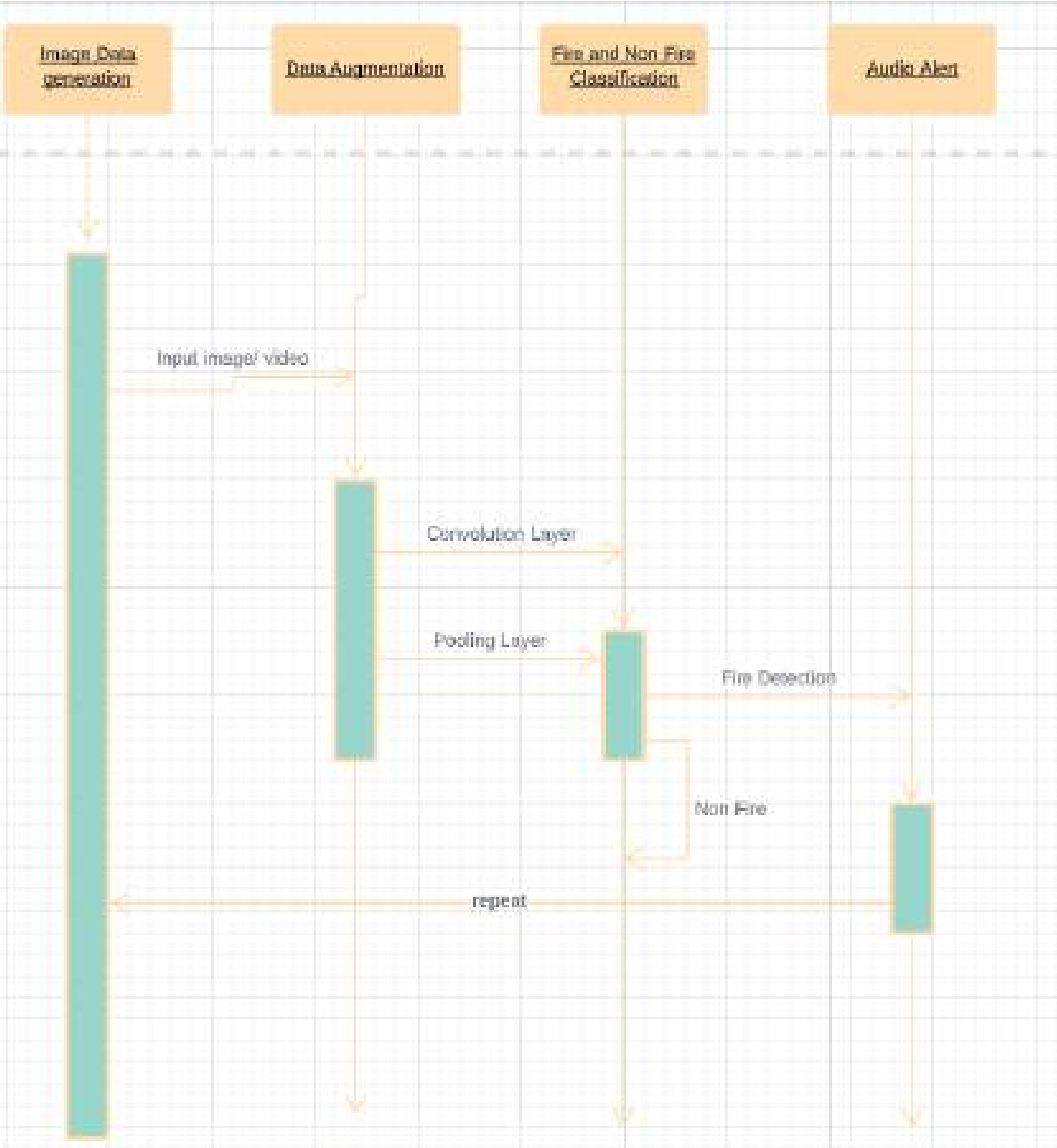

Fig:3.6 Sequence diagram for Early Fire Detection System

3.7 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

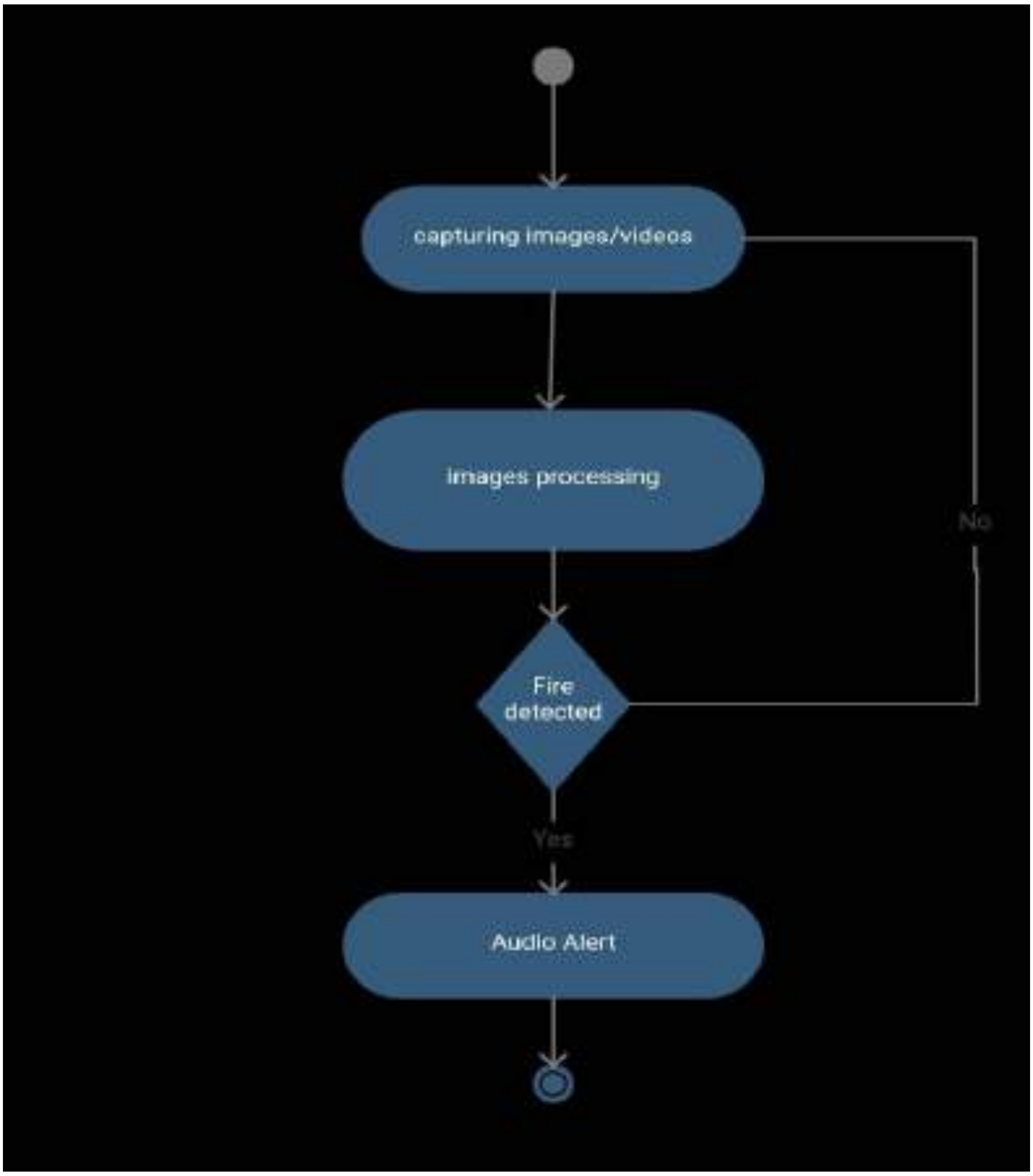

Fig:3.7 Activity Diagram for Early Fire Detection System

4.IMPLEMENTATION

4. IMPLEMENTATION

4.1 SAMPLE CODE

```
import numpy as np
import pandas as pd
import os
import h5py
import cv2
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
import cv2 as cv
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score
from sklearn.metrics import accuracy_score
from PIL import Image
import pyttsx3

from tensorflow.keras.callbacks import EarlyStopping
from sklearn.metrics import confusion_matrix
import seaborn as sns

from keras.models import load_model
from keras.preprocessing import image
model = load_model('Trained_Model2.h5')
def alarm(msg):
    engine = pyttsx3.init()
    engine.setProperty('rate', 150)
    engine.say(msg)
    engine.runAndWait()
frameWidth = 640  # CAMERA RESOLUTION
frameHeight = 480
```

```
threshold = 80
cap = cv2.VideoCapture(0)
cap.set(3, frameWidth)
cap.set(4, frameHeight)
cap.set(10, brightness)
font = cv2.FONT_HERSHEY_SIMPLEX


def getName(Diff_Pred):
    if(Diff_Pred == 1):
        return "Fire"


while True:
    success, imgOrignal = cap.read()
    # PROCESS IMAGE
    img = np.asarray(imgOrignal)
    img = cv2.resize(img, (160, 160))
    cv2.imshow("Processed Image", img)
    img = img / 255
    img = img.reshape(1, 160, 160, 3)
    cv2.putText(imgOrignal, "CLASS: ", (20, 35), font, 0.75, (0, 0, 255), 2,
cv2.LINE_AA)
    cv2.putText(imgOrignal, "PROBABILITY: ", (20, 75), font, 0.75, (0, 0, 255), 2,
cv2.LINE_AA)
    # PREDICT IMAGE
    predictions = model.predict(img)
    predictions = int(predictions)
    probabilityValue = np.amax(predictions)

    if(probabilityValue*100 > threshold):
```

```
cv2.putText(imgOrignal, str(predictions) + " "+getName(predictions), (120, 35), font,
0.75, (0, 0, 255), 2, cv2.LINE_AA)
        cv2.putText(imgOrignal, str(round(probabilityValue * 100, 2)) + "%", (180, 75),
font, 0.75, (0, 0, 255), 2,cv2.LINE_AA)
        alarm("Fire Alert")
    else:
        cv2.putText(imgOrignal, " No Fire" , (120, 35), font, 0.75, (0, 0, 255),
2,cv2.LINE_AA)
        cv2.putText(imgOrignal, "100%", (180, 75), font, 0.75, (0, 0, 255),
2,cv2.LINE_AA)
    cv2.imshow("Result", imgOrignal)

    if cv2.waitKey(1) and 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

4.2 #TESTING CODE

```
import numpy as np
import pandas as pd
import os
import h5py
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
import cv2 as cv
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score
from sklearn.metrics import accuracy_score
from tensorflow.keras.callbacks import EarlyStopping
from sklearn.metrics import confusion_matrix
import seaborn as sns

# Preprocessing
X, Y = [], []
img_size= (160, 160)
for root,_,files in
os.walk("C:/Users/91918/Downloads/Uma/Uma/fire_dataset/fire_images"):
    for file in files:
        x = cv.imread(os.path.join(root, file))
        if(x is None):
            print(file)
            continue
        x = cv.resize(x, img_size, interpolation=cv.INTER_AREA)
        X.append(x/255.0)
        Y.append(1)
for root,_,files in
os.walk("C:/Users/91918/Downloads/Uma/Uma/fire_dataset/non_fire_images"):
```

```
for file in files:
    x = cv.imread(os.path.join(root, file))
    if(x is None):
        print(file)
        continue
    x = cv.resize(x, img_size, interpolation=cv.INTER_AREA)
    X.append(x/255.0)
    Y.append(0)

X = np.stack(X)
Y = np.asarray(Y)
num_fire_images = np.sum(Y == 1)
num_non_fire_images = np.sum(Y == 0)
print("Number of Fire Images:", num_fire_images)
print("Number of non-Fire Images:", num_non_fire_images)
X_tv, X_test, Y_tv, Y_test = train_test_split(X, Y, test_size = 0.1, shuffle=True,
stratify=Y, random_state=1)
X_train, X_val, Y_train, Y_val = train_test_split(X_tv, Y_tv, test_size=0.2,
shuffle=True, stratify=Y_tv, random_state=2)
class_weights = {0:num_fire_images/X.shape[0],
1:num_non_fire_images/X.shape[0]}
print("Class Weights:",class_weights)

#Model

from keras.models import Sequential
from keras import layers
model = Sequential()

model.add(layers.Conv2D(512,(3,3),input_shape = (160,160,3),activation='relu'))
model.add(layers.MaxPooling2D(3,3))
model.add(layers.Conv2D(224,(3,3),activation='relu'))
model.add(layers.MaxPooling2D(3,3))
```

```
model.add(layers.Conv2D(128,(3,3),activation='relu'))
model.add(layers.MaxPooling2D(3,3))

model.add(layers.Flatten())

model.add(layers.Dense(256, activation = 'relu'))
model.add(layers.Dense(64,activation = 'relu'))
model.add(layers.Dense(1,activation= "sigmoid"))
callbacks = [EarlyStopping(monitor = 'val_loss',patience =
18,restore_best_weights=True)]
model.compile(optimizer='adam',loss = 'binary_crossentropy',metrics=['accuracy'])
model.fit(X_train,Y_train,validation_data=(X_val,Y_val),epochs = 30,batch_size =
32,callbacks = callbacks)

# Model testing
Y_pred = (model.predict(X_test)>0.5).astype(np.int64)
acc = accuracy_score(Y_test, Y_pred)
rec = recall_score(Y_test, Y_pred)
pre = precision_score(Y_test, Y_pred)

print("Test Accuracy:",acc*100)
print("Test Precision:",pre*100)
print("Test Recall:",rec)

#CONFUSION MATRIX
plt.figure(figsize = (20,20))

sns.heatmap(confusion_matrix(Y_test,Y_pred),annot = True)
plt.title("CONFUSION MATRIX")
plt.xlabel("Predicted")
plt.ylabel("True")

model.save("Trained_Model2.h5")
```

```
y_pred = model.predict(X_test)
y_pred = y_pred.reshape(-1)
y_pred[y_pred<0.5] = 0
y_pred[y_pred>=0.5] = 1
y_pred = y_pred.astype('int')
y_pred

from keras.preprocessing import image
image_path = "C:/Users/91918/Downloads/nonfire.png"
img = image.load_img(image_path,target_size=(160,160))
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
Diff_Pred = model.predict(x)
classIndex = model.predict_classes(x)
print(classIndex)

from keras.preprocessing import image
image_path = "C:/Users/91918/Downloads/fire.png"
img = image.load_img(image_path,target_size=(160,160))
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
Diff_Pred = model.predict(x)
classIndex = model.predict_classes(x)
print(classIndex)
```

5 SCREENSHOTS

5.1 OUTPUT SCREENSHOT WITH FIRE

Fig5.1:Output Screenshot with fire

5.2 OUTPUT SCREENSHOT WITHOUT FIRE

Fig5.2: Output screenshot with no fire

5.3 TRAINING ACCURACY AND TRAINING LOSS

5.3.1 TRAINING ACCURACY

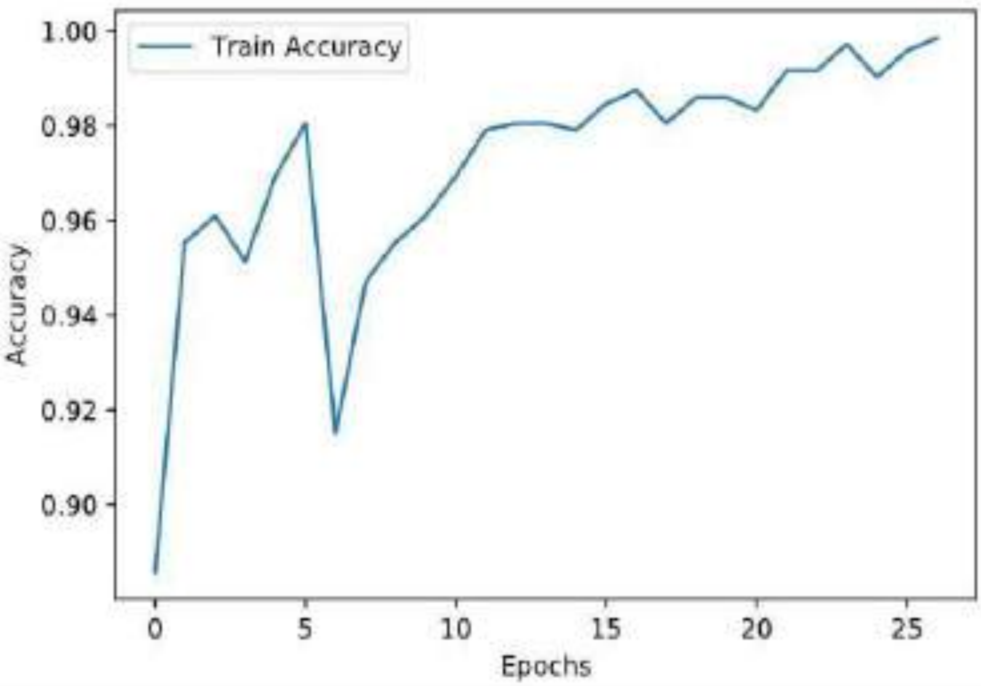

Fig:5.3.1 Train accuracy

The above figure shows the Train accuracy of our proposed model. Here, we can observe the acccuracy to epoch. After every epoch, the accuracy is increasing respectively. Where epoch is a particular period of time. When we use less epoch the accuracy is increasing.

5.4 TRAINING LOSS

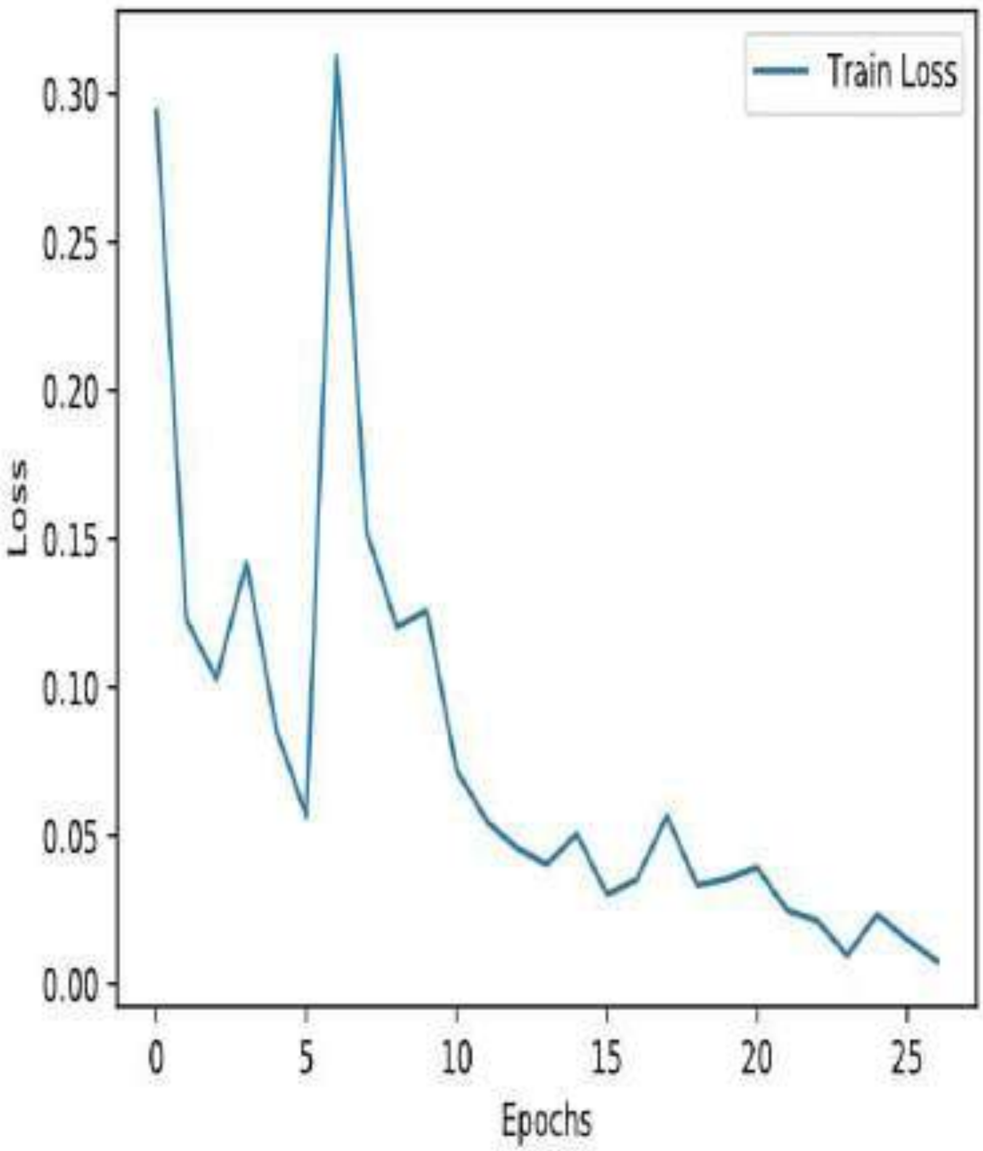

Fig5.4: Training Loss

The above figure shows the loss to epochs. The above figure depicting that when the number of epochs are increasing the rate of accuracy is decreasing accordingly.

Therefore, we can say that number of epochs and the accuracy are vice-versa. When number of epochs are increasing the accuracy will be decreased, and when then number of epochs are decreasing then the accuracy is increasing.

6 . CONCLUSION

6. CONCLUSION & FUTURE SCOPE

6.1 PROJECT CONCLUSION

.collisions, medical emergencies, and fires.
Failure to control fire at an early stage can lead to huge disasters.
Our proposed system shows two custom models for fire detection, which can be of assistance to disaster management teams in managing fire disasters on time, thus preventing huge losses.
Using the great potential of CNNs, we can detect fire from images or videos at an early stage.

6.2 FUTURE SCOPE

We can implement it in cctv cameras so that we can prevent loss of of fire in cities and as well as in forest areas. By implementing an early fire detection system we can avoid fire accidents in earlier and take precautions for not letting the fire to spread.

7.BIBILOGRAPHY

8.BIBILOGRAPHY

9.1 BOOK REFERENCE

None

9.3 TECHNICAL PUBLICATION REFERENCE

[1] V.Hristidis, S.-C. Chen, T. Li, S. Luis, and Y. Deng, "Survey of data management and analysis in disaster
situations," Journal of Systems and Software, vol. 83, pp. 1701-1714, 2010.

[2] N. R. Adam, B. Shafiq, and R. Staffin, "Spatial computing and social media in the context of disaster
management," IEEE Intelligent Systems, vol. 27, pp. 90-96, 2012.

[3] X. Song, L. Sun, J. Lei, D. Tao, G. Yuan, and M. Song, "Event-based large scale surveillance video
summarization," Neurocomputing, vol. 187, pp. 66-74, 2016.

[4] R. Chi, Z.-M. Lu, and Q.-G. Ji, "Real-time multi-feature based fire flame detection in video," IET Image
Processing, vol. 11, pp. 31-37, 2016.

[5] S.-W. Lo, J.-H. Wu, F.-P. Lin, and C.-H. Hsu, "Cyber surveillance for flood disasters," Sensors, vol. 15,
pp. 2369-2387, 2015.

[6] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in
Image Processing, 2004. ICIP'04. 2004 International Conference on, 2004, pp. 1707-1710.

[7] D. Guha-Sapir, F. Vos, R. Below, and S. Penserre, "Annual disaster statistical review 2015: the numbers
and trends," http://www.cred.be/sites/default/files/ADSR_2015.pdf, 2015.

[8] T. Toulouse, L. Rossi, M. Akhloufi, T. Celik, and X. Maldague, "Benchmarking of wildland fire colour

segmentation algorithms," IET Image Processing, vol. 9, pp. 1064-1072, 2015.

[9] T.Qiu, Y. Yan, and G. Lu, "An autoadaptive edge-detection algorithm for flame and fire image processing,"
IEEE Transactions on instrumentation and measurement, vol. 61, pp. 1486-1493, 2012.

[10] C.-B. Liu and N. Ahuja, "Vision based fire detection," in Pattern Recognition, 2004. ICPR 2004.
Proceedings of the 17th International Conference on, 2004, pp. 134-137.

[11] T.Celik, H. Demirel, H. Ozkaramanli, and M. Uyguroglu, "Fire detection using statistical color model in
video sequences," Journal of Visual Communication and Image Representation, vol. 18, pp. 176-185, 2007.

[12] B. C. Ko, S. J. Ham, and J. Y. Nam, "Modeling and formalization of fuzzy finite automata for detection of
irregular fire flames," IEEE Transactions on Circuits and Systems for Video Technology, vol. 21, pp. 1903-
1912, 2011.

[13] 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 TRANSACTIONS on circuits
and systems for video technology, vol. 25, pp. 1545-1556, 2015.

[14] M. Mueller, P. Karasev, I. Kolesov, and A. Tannenbaum, "Optical flow estimation for flame detection in
videos," IEEE Transactions on Image Processing, vol. 22, pp. 2786-2797, 2013.

[15] B. U. Töreyin, Y. Dedeoğlu, U. Güdükbay, and A. E. Cetin, "Computer vision based method for real-time.

Early Fire Detection System

K. Karunakar[1],S.Uma Maheshwari[2], b. Jhansi[3], K. Mahathi[4]
[1]*Associate Professor, CSE, CMR Technical Campus, Hyderabad, India*
[2]*Student, CSE, CMR Technical Campus, Hyderabad, India*
[3]*Student, CSE, CMR Technical Campus, Hyderabad, India*
[4]*Student, CSE, CMR Technical Campus, Hyderabad, India*

***Abstract*: Fire disasters can be both manufactured and naturally occurred, but most of them are man-made disasters. Fire disasters result in huge losses both economically and ecologically. We need an early fire detection system, that produces an autonomous response and helps in the early detection of disaster occurrence. Therefore, we propose an early fire detection framework using convolutional neural networks (CNN) for cameras, which detect fire in varying indoor and outdoor environments. In such cases, if the surveillance area is too large such as large buildings, complex spaces, or it would be tough to feature recognition. Applying the convolutional neural network (CNN) technology to image recognition can avoid randomness to a large extent in the feature extraction process. To ensure the autonomous response, we propose an efficient mechanism for cameras in the surveillance system.**

KeyWords:Images, deep learning, disaster management, Fire detection.

Introduction

The project Early fire detection system is important as it helps to achieve the fire as soon as possible that can be applicable in real life. Failure to control fire at an early stage can lead to large disasters. Our proposed model shows a CNN model for fire detection is assistance to disaster management teams for managing disasters that can prevent losses. For detecting fire images and videos at an early stage, the potential CNNs can be used. Our proposed model will start with the input image layer with the size of size160*160*3 for the proposed model. Then we used layers that contain batch normalization, ReLu(rectifier linear unit), Conv2d, Max-pooling layer, and dense layer, Where Conv2d layer used for mapping features for the input images with a filter size of [3*3]. Usually, this size is for the CNN architecture. Max-pooling used for selecting the maximum elements from the region map covered by the filter, Width, and Height of the input could be defined by the Filter size. We have used sigmoid for the binary classification, as our model gives theoutput in the form of binary values such as '0' and '1' represent fire, and non-fire respectively. Adamoptimizer had used. In thisproject, there is no separate code for the execution as internally tests the validation. The batch size is 32. It gives the accuracy with 96% as we have reduced the batch size to 32%lost, and the error rate is 0.01. When we get the error rate as 0.01 then it gives 96% accuracy.

1. Literature Survey

In this project, we have critically discuss the fire detection methods of the current literature along with its strengths and weakness. Later, we briefly highlight our approach of solving the problems of some of the early fire detection methods. Finally, we have discussed that how early fire detection can be used in effective disaster management systems. The advancements in technology have resulted in a wide variety of sensors for different applications like wireless capsule sensors for visualization of interior of a human body, vehicle sensors for obstacle detection, and fire alarming sensors. The current fire alarming sensors such as infrared, ion, and optical sensors need close proximity of the heat, fire, radiation or smoke for activation. As an alternate to these sensors, the vision-based sensors are widely used, which provide many advantages compared to the traditional sensors such as lower cost, fast response time, larger coverage of surveillance area, and less human intervention, avoiding the need of visiting the location from where the fire alert has been triggered.

Liu et al. [10] investigated three differentmodels including spectral,spatial and temporal for fire regions in images. However, their method is based onassumption considering irregularshape of fire, whichis not always the case as moving objects canalso change theirshape. Another firedetection approach ispresented in[22] for forests using contours based on wavelet analysis and FFT. Authors in [23] investigated YCbCrcolor model and devised new rules for effective separation of luminance and chrominance components, which leaded them to rule based pixels classification of flame. Another color model YUV along with motionwas explored by authors in [24] for classification into candidate pixels for fire or non-fire. Besides the investigation of color models, specific low level features of fire regions such as skewness, color, roughness, area size etc., have also been used for determining the frame-to-frame changes, which in combination with Bayes classifier can recognize fire [17]. Anothermethod is presented in [25] considering lookup table for detection of fireregions and their confirmation using temporal variation. This method is based on heuristic features, decreasing the surety ofgetting the same results while changing the input data.

In our model the images are captured using a PC Cam and were able to get an accuracy of 96%. The proposed model was evaluated based on fire detectiondataset.

By considering the aforementioned fire detection methods, it can be observed that some of the methods are too naïve, whose execution time is fast but such methods compromise on accuracy, producing a large number of false alarms. Some methods have achieved good fire detection accuracies but their execution time is too much, hence they cannot be applied in the real-world environments especially in critical areas where minor delay can lead to huge disasters. Hence, for more accurate and early detection of fire, we need a robust mechanism, which can detect fire during varying conditions andcansend the important keyframes and alertimmediately to disaster management systems.

2. Methodology

A. *System Architecture*

System Architecture describes how we have approached or achieved our project i.e: early fire detection system. The way we have approached to our aim can be explained clearly with the system architecture

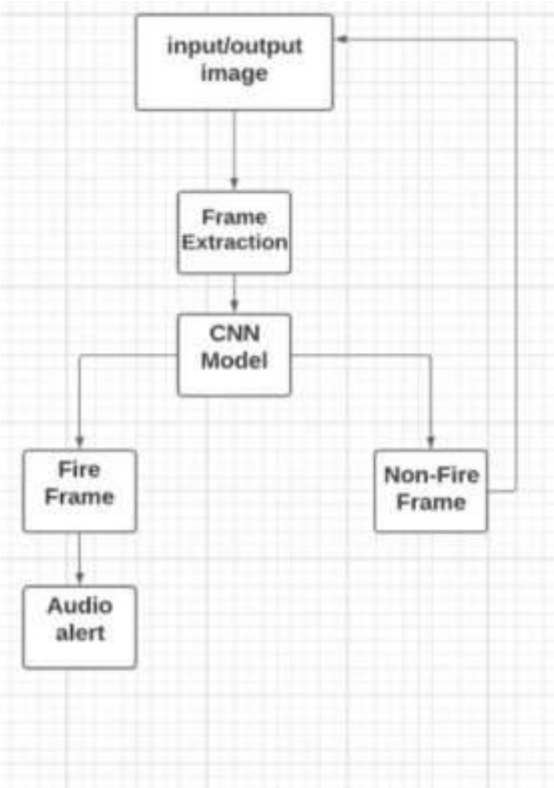

Fig. 1.: System Architecture

In this project, an early fire detection system was built with the help of deep learning, which uses vision-based systems to detect fire using Convolutional Neural Network architecture. By using CNN architecture, we are training our model and dataset. At the time of training a dataset, it takes the surveillance videos and gives the output by detecting the images and videos. Firstly, we will be given the input and output images. After, that we process the images and then extract the images. The sigmoid layer will classify the images as Fire frame and Non-Fire Frame. Sigmoid is used for binary classification. At the time of training a model, it shows as '0' for non-fire and '1' for fire. Our CNN model is implemented for cost-effective fire detection.

B. *CNN Architecture*

We have three stages in our CNN architecture. They are as follows 1. Image pre-processing, 2. Feature extraction, and 3. Fire detection.

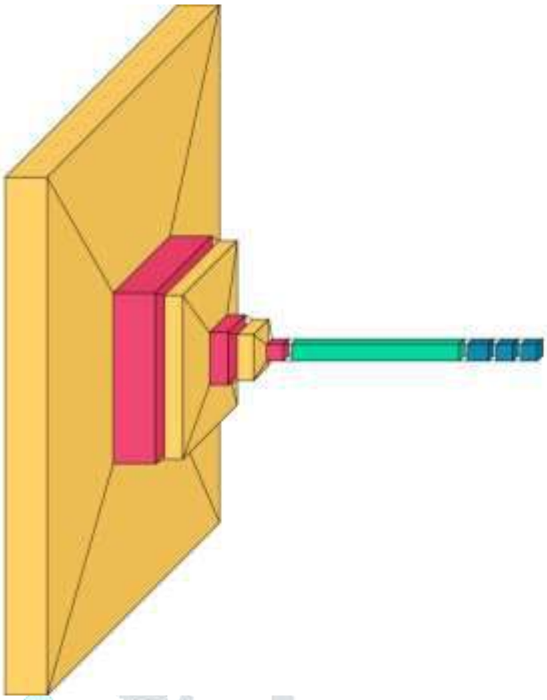

Fig.2: CNN Architecture

At stage1, it processes the image data from a camera, at stage2, feature extraction is the main part of algorithms, it extracts the features of smoke and flame. The input image or video is processed through the convolution layer and pooling layer for detecting appropriately without producing any false results. After flattening, the process trains as mentioned in the above figure. Through the activation of ReLu, we change to dense layer to 256.Again 64 with activation ReLu. Atlast, we get 1 with the activation of sigmoid. Here these layers help the machine in differentiating different environments like dangerous threats, normal environments. This is done by feature extraction and classification using the probability distribution of the softmax layer between fire and non-fire classes.

C. *Dataset Creation*

Dataset used is Fire Detection Dataset

Fire Detection Dataset consists of two classes. They are:

1. Non-Fire Images
2. Fire Images

1. Non-Fire Images

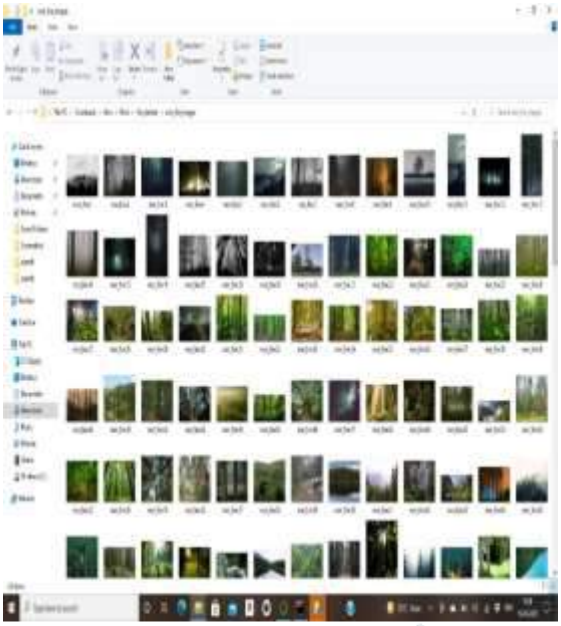

Fig.3: Non-Fire Images

2. Fire Images

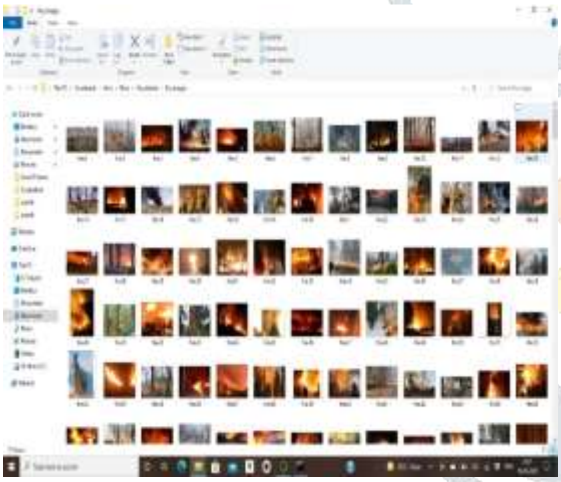

Fig. 4: Fire Images

D. Training and Testing

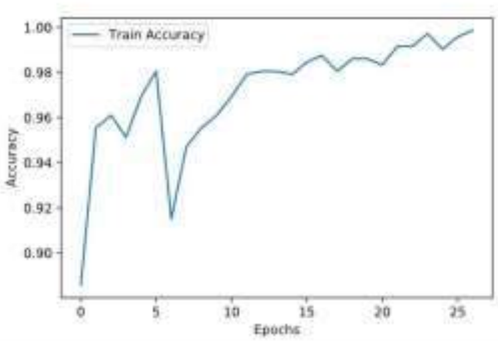

Fig.5: Train Accuracy

The above figure shows the Train accuracy of our proposed model. Here, we can observe the acccuracy to epoch. After every epoch, the accuracy is increasing respectively. Where epoch is a particular period of time. When we use less epoch the accuracy is increasing.

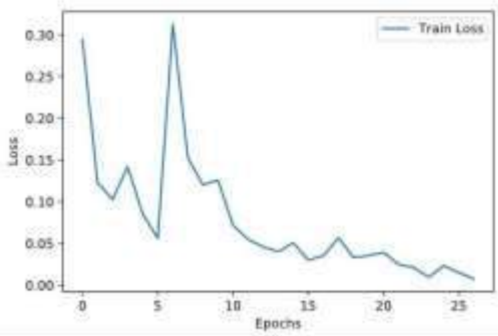

Fig.6: Train loss

The above figure shows the loss to epochs. The above figure depicting that when the number of epochs are increasing the rate of accuracy is decreasing accordingly. Therefore, we can say that number of epochs and the accuracy are vice-versa.

When number of epochs are increasing the accuracy will be decreased, and when then number of epochs are decreasing then the accuracy is increasing.

3.Results and Discussions

Fig.7: Detection of fire image

When it achieve the fire images or fire videos, it gives an alarm continuously to alert the people around.

Fig.8: Detection of Non-Fire image

If it doesn't find or detect any fire it doesn't give any alarm sound but it shows us like No fire with 100% probability.

Fig.9: Detecting in Live with No fire

The above figure depicts that, No fire has been detected. So, it doesn't give any alarm.

Fig.10: Detecting Fire with Fire

The above figure depicts that, the fire has been found or achieved with 100% probability.

4.Conclusion

Using smart cameras you can identify various suspicious incidents such as collisions, medical emergencies, and fires. Failure to control fire at an early stage can lead to huge disasters. Our proposed system shows two custom models for fire detection, which can be of assistance to disastermanagement teams in managing fire disasters on time, thuspreventing huge losses. Using the great potential of CNNs, we candetect fire from images or videos at an early stage. The system achieved a maximal accuracy of 96%. There is a lot of scope scope of this project.

5.References

1. K. Muhammad, J. Ahmad, I. Mehmood, *et al.*
Convolutional neural networks based fire detection in surveillance videos
IEEE Access, 6 (2018), pp. 18174-18183
View Record in ScopusGoogle Scholar
[2]

2. C. Tao, J. Zhang, P. Wang

Smoke detection based on deep convolutional neural networks2016 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII) (2016), pp. 150-153
View Record in ScopusGoogle Scholar
[3]

3 Filonenko, L. Kurnianggoro, K. Jo
Comparative study of modern convolutional neural networks for smoke detection on image data
2017 10th International Conference on Human System Interactions (HSI) (2017), pp. 64-68

4 IEEE Access, 6 (2018), pp. 18174-18183
View Record in ScopusGoogle Scholar
5 [2]
6 C. Tao, J. Zhang, P. Wang
7 **Smoke detection based on deep convolutional neural networks**
8 2016 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII) (2016), pp. 150-153
9 View Record in ScopusGoogle Scholar
10 [3]
11 A. Filonenko, L. Kurnianggoro, K. Jo
12 Comparative study of modern convolutional neural networks for smoke detection on image data
13 2017 10th International Conference on Human System Interactions (HSI) (2017), pp. 64-68
14 CrossRefView Record in ScopusGoogle Scholar
15 [4]
16 Z. Yin, B. Wan, F. Yuan, *et al.*
17 A deep normalization and convolutional neural network for image smoke detection
18 IEEE ACCESS, 5 (2017), pp. 18429-18438
19 CrossRefView Record in ScopusGoogle Scholar
20 [5]
21 A.J. Dunnings, T.P. Breckon
22 Experimentally defined convolutional neuralnetwork architecturevariantsfor non-temporal real-time fire detection
23 2018 25th IEEE International Conference on Image Processing (ICIP) (2018), pp. 1558-1562
24 CrossRefGoogle Scholar
25 [6]
26 A. Namozov, Y. Cho
27 An efficient deep learning algorithm for fire and smoke detection with limited data
28 Adv. Electr. Comput. Eng., 18 (2018), pp. 121-128
29 CrossRefView Record in ScopusGoogle Scholar
30 [7]
31 W. Mao, W. Wang, Z. Dou, Y. Li
32 Fire recognition based on multi-channel convolutional neural network
33 Fire Technol., 54 (2018), pp. 531-554
34 CrossRefView Record in ScopusGoogle Scholar
35 [8]
36 K. Muhammad, J. Ahmad, S.W. Baik
37 Early fire detection using convolutional neural networks during surveillance for effectivedisaster management
38 Neurocomputing, 288 (2018), pp. 30-42
39 ArticleDownloadPDFView Record in ScopusGoogle Scholar
40 [9]
41 Y. Hu, X. Lu
42 Real-time video fire smoke detection by utilizing spatial-temporal ConvNet features
43 Multimed. Tool. Appl., 77 (2018), pp. 29283-29301

44 CrossRefView Record in ScopusGoogle Scholar
45 [10]
46 A. Namozov, Y. Cho
47 An efficient deep learning algorithm for fire and smoke detection with limited data
48 Adv. Electr. Comput. Eng., 18 (2018), pp. 121-128
49 CrossRefView Record in ScopusGoogle Scholar
50 [11]
51 L. Wonjae, K. Seonghyun, L. Yong-Tae, L. Hyun-Woo, C. Min
52 Deep neural networks for wild fire detection with unmanned aerial vehicle
53 2017 IEEE International Conference on Consumer Electronics (ICCE) (2017), pp. 252-253
54 Google Scholar
55 [12]
56 Y. Luo, L. Zhao, P. Liu, D. Huang
57 Fire smoke detection algorithm based on motion characteristic and convolutional neural networks
58 Multimed. Tool. Appl., 77 (2018), pp. 15075-15092
59 CrossRefView Record in ScopusGoogle Scholar

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Sirigandam Umamaheshwari

In recognition of the publication of the paper entitled

Early Fire Detection System

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 8 Issue 6 , June-2021 | Date of Publication: 2021-06-08

Parin P
EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2106105 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2106105 Registration ID : 310326

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Bandi Jhansi

In recognition of the publication of the paper entitled

Early Fire Detection System

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 8 Issue 6 , June-2021 | Date of Publication: 2021-06-08

Parisa P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2106105 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2106105** **Registration ID : 310326**

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Konda Mahathi

In recognition of the publication of the paper entitled

Early Fire Detection System

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 8 Issue 6 , June-2021 | Date of Publication: 2021-06-08

Parin P
EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2106105 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2106105** **Registration ID : 310326**

Journal of Emerging Technologies and Innovative Research

An International Open Access Journal Peer-reviewed, Refereed Journal

www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

K Karunakar

In recognition of the publication of the paper entitled

Early Fire Detection System

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 8 Issue 6 , June-2021 | Date of Publication: 2021-06-08

Parin P

EDITOR

EDITOR IN CHIEF

ISSN 2349-5162

JETIR2106105 **Research Paper Weblink http://www.jetir.org/view?paper=JETIR2106105** **Registration ID : 310326**