EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRE

PROJECT REPORT

IBM NALAIYA THIRAN

Submitted By

TEAM ID: PNT2022TMID17130

TEAM MEMBERS:

SUBASH S	**-**	**6113191041106**
VASAN A	**-**	**6113191041111**
VASIKARAN K	**-**	**6113191041113**
VIJAY S	**-**	**6113191041115**

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

In

DEPARTMENT OF ELECTRONICS AND COMMUNICATION

MAHENDRA ENGINEERING COLLEGE (AUTONOMOUS),

CHAPTER NO	TITLE	PAGE NO
1	**INTRODUCTION**	3
	1.1 PROJECT OVERVIEW	3
	1.2 PURPOSE	3
2	**LITERATURE SURVEY**	4
	2.1 EXISTING SYSTEM	4
	2.2 REFERENCES	4
	2.3 PROBLEM STATEMENT DEFINITION	4
3	**IDEATION AND PROPOSED SOLUTION**	5
	3.1 EMPATHY MAP CANVAS	5
	3.2 IDEATION AND BRAINSTORMING	5
	3.3 PROPOSED SOLUTION	6
	3.4 PROBLEM-SOLUTION FIT	9
4	**REQUIREMENT ANALYSIS**	9
	4.1 FUNCTIONAL REQUIREMENT	9
	4.2 NON – FUNCTIONAL REQUIREMENT	10
5	**PROJECT DESIGN**	11
	5.1 DATA FLOW DIAGRAM	11
	5.2 SOLUTION AND TECHNICAL ARCHITECTURE	12
6	**PROJECT PLANNING AND SCHEDULING**	15
	6.1 SPRINT PLANNING AND ESTIMATION	15
	6.2 SPRINT DELIVERY SCHEDULE	15
7	**CODING AND SOLUTIONS**	17
	7.1 **FEATURE**	22
8	**TESTING**	23
	8.1 TEST CASE	23
	8.2 USER ACCEPTANCE	24
9	**RESULT**	24
10	**ADVANTAGES AND DISADVANTAGES**	25
11	**CONCLUSION**	25
12	**FUTURE SCOPE**	26
13	**APPENDIX**	26
	13.1 SOURCE CODE	26

EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRE

PROJECT REPORT

1. INTRODUCTION

Fire detection is one of the essential modules in an early warning system, which is used to identify abnormal events in a monitoring area. Fire detectors are used to provide the earliest possible warning of a fire. Conventional fire detectors currently use smoke and temperature sensors. If the sensor is placed in an open and wide area such as a forest, densely populated settlements, and roads, it will be less effective and cost a significant amount of money. In addition, conventional fire detectors have problems regarding delays and alarm sound errors. In other words, the utilization of camera monitoring is currently increasing to ensure citizens' safety. Therefore, it is possible for Closed Circuit Television (CCTV) cameras to detect fires using digital image processing and computer vision technology, referred to as image-based fire detection. The advantages of image-based fire detection compared to conventional fire detectors can be placed in an open and wide area so that the costs incurred can be cheaper.

1.1Project Overview

Image-based fire detection is strongly influenced by the features used to distinguish fire from other objects. Two types of features are often used to detect fire: handcrafted features and non-handcrafted features. Handcrafted features are designed with predetermined rules. Examples of these features are motion, shape, color, and texture. Meanwhile, non-handcrafted features are obtained directly from the neural network layers.

1.2 Purpose

This is a Requirements Specification document for Emerging methods for early detection of forest fire. This project is used to detect forest fire early so that it could reduce vulnerability ofupcoming disaster. This document describes the scope, objectives and goal of the new system. In addition to describing non-functional requirements, this document models the functional requirements with use cases, interaction diagrams, and class models.

2. LITERATURE SURVEY

2.1 Existing problem

- It is difficult to predict and detect Forest Fire in a sparsely populated forest area and it is more difficult if the prediction is done using ground-based methods like Camera or Video-Based approach.
- Using only IR images may lead to false alarms. The combination of IR images with visible range image together to reduce the false alarm rates of fire detection arm result.
- Weather data must be included as it plays a major role in the occurrence, growth, spread and the extinction of wildfires. It can impact on the strength and movement of fire, and thus burn more land which makes its extinction even more difficult.

2.2 References

1. Georgi Hristov; Jordan Raychev; Diyana Kinaneva; Plamen Zahariev , Published in: 2018 28th EAEEIE Annual Conference (EAEEIE).

2. Chi Yuan, Youmin Zhang, and Zhixiang Liu , Published in: 2015, Published by NRC Research Press.

3. Mohamed Hefeeda and Majid Bagheri , Published in: June 26, 2008.

4. PRIYADARSHINI M HANAMARADDI , Published in: January 2016.

5. Dr. Panagiotis Barmpoutis, Periklis Papaioannou, Dr. Kosmas Dimitropoulos, Dr. Nikos GRAMMALIDIS , Published in: 11 November 2020.

6. Vinay Chowdary , Mukul Kumar Gupta , Rajesh Singh, Published in:2018

7. Majid Bahrepour, Nirvana Meratnia, Paul Havinga , Published in: January 2008.

8. Dr.L.Latha , Published in: January 2015

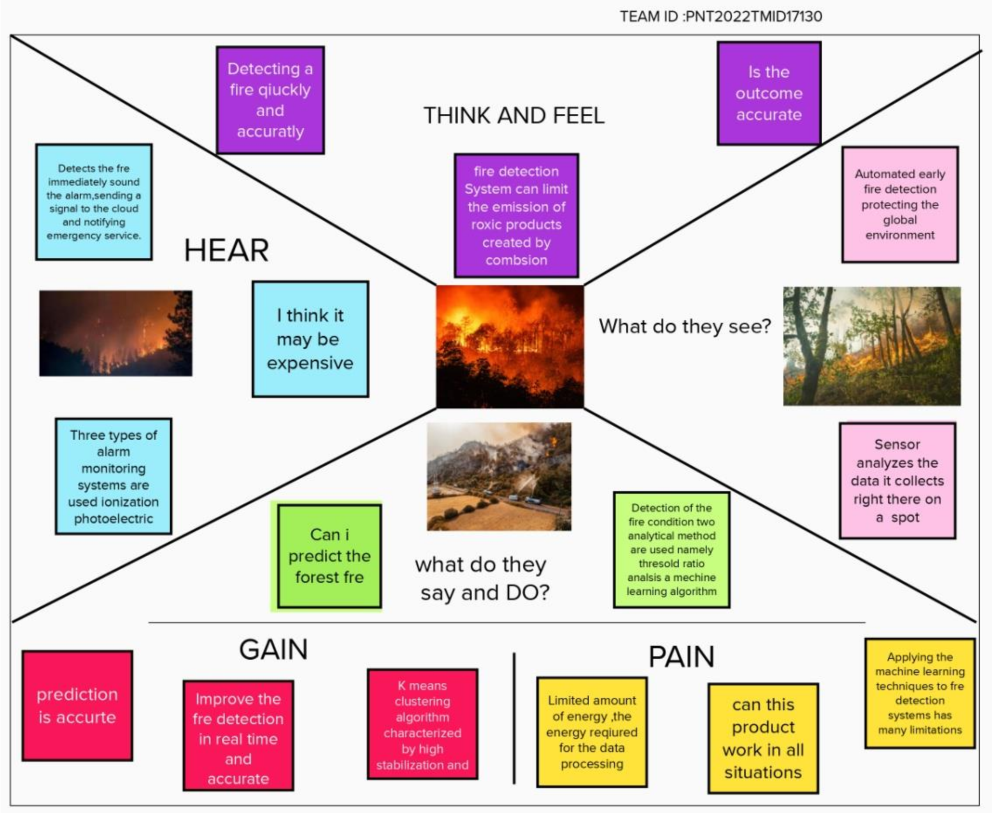
9. P. Piccinini, S. Calderara, and R. Cucchiara , Published in: September, 2006

2.3 Problem Statement Definition

Forest fires are a major environmental issue, Over 9 million acres of land have been destroyed due to treacherous wildfires. Satellites can be an important source of data prior to and also during the Fire due to its reliability and efficiency. The various real-time forest fire detection and prediction approaches, with the goal of informing the local fire authorities.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



TEAM ID :PNT2022TMID17130

THINK AND FEEL

Detecting a fire qiuckly and accuratly

Is the outcome accurate

fire detection System can limit the emission of roxic products created by combsion

Automated early fire detection protecting the global environment

HEAR

Detects the fre immediately sound the alarm,sending a signal to the cloud and notifying emergency service.

I think it may be expensive

Three types of alarm monitoring systems are used ionization photoelectric

What do they see?

Sensor analyzes the data it collects right there on a spot

Detection of the fire condition two analytical method are used namely thresold ratio analsis a mechine learning algorithm

Can i predict the forest fre

what do they say and DO?

GAIN

prediction is accurte

Improve the fre detection in real time and accurate

K means clustering algorithm charactenized by high stabilization and

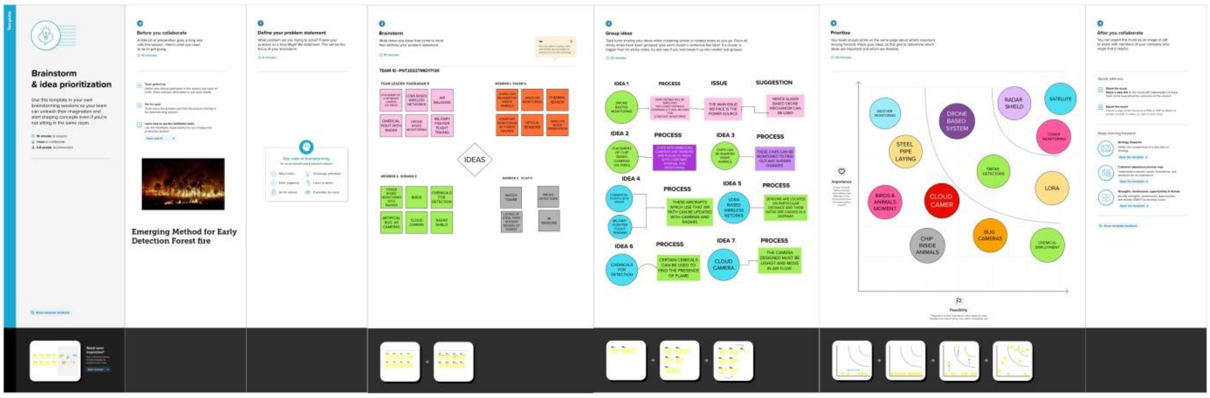
PAIN

Limited amount of energy ,the energy requried for the data processing

can this product work in all situations

Applying the machine learning techniques to fre detection systems has many limitations

3.2 Ideation & Brainstorming



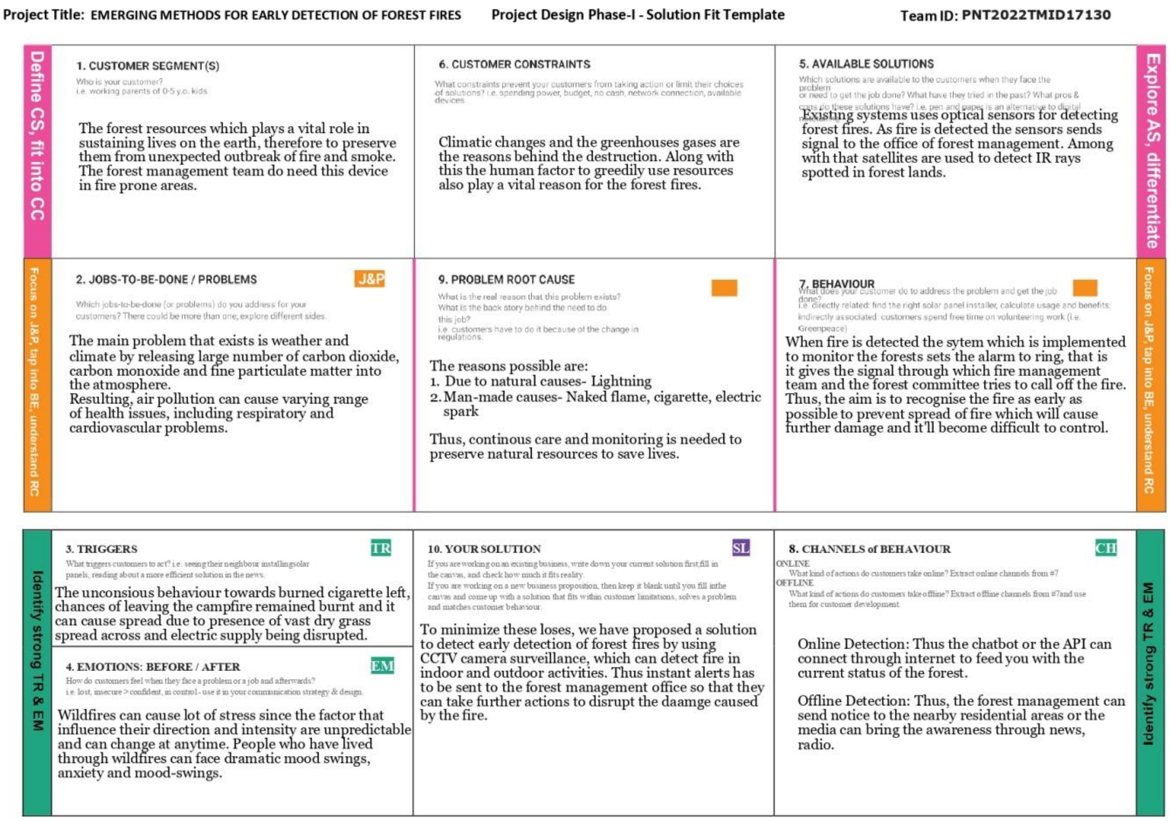
3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem tobe solved)	Forest fires are considered as one of the most widespread hazards in a forested landscape. They have a serious threat to forest and its flora and fauna. Unplanned and abrupt forest fires are a major cause of forest degradation, while a controlled fire to manage and check the spread of unwanted forest fires serves as the action to improve the forest. So therefore, we’ve to detect prevention measure which should be taken to identify the fire prone areas and the tools which needed to be developed to minimize the loss and as well as implement forest fire committee to work for the reduction of damage caused. A forest fire risk prediction algorithm, based on support vector machines, is presented. The algorithm depends on previous weather conditions in order to predict the fire hazard level of a day.
2.	Idea / Solution description	Our solution aims at collecting the vast range of dataset to test and train the model regularly by using **CNN** where the system can detect immediately if any ignition of fire is found, where the video can be surveyed by satellite. Then **Cloudant DB** is brought to use where the large amount of data is stored and fetched which acts as a server. **Open CV** acts as a tool for processing videos which are captured. To send alerts to forest committee **Twilio API** is used where alerts are passed on detection. **Watson Assistant** also a chatbot tool which can help you monitor if any guide is needed.

3.	Novelty / Uniqueness	Existing system uses electronic sensors to detect forest fire and smoke. The change in temperature indicates the presence of forest fire and smoke in a region which can be detected by the sensors using radiation heat. As forests are in remote area it's difficult for installation and maintenance of sensors. Our proposed system depends on using AI to make it cheaper and easier for the forest management. Accuracy and timely prediction using AI, CNN and API made it possible.
4.	Social Impact / Customer Satisfaction	Forest fires are dangerous for the existence of life as they carry wildlife and natural resources which gives life to various living bodies. Thus, fires are occurred expectedly or unexpectedly which has to be prevented as earlier as we can. Therefore, forest management should be active enough to be aware and keeping an eye to check the forest fields regularly. As, what we save, saves us should be the Social awareness to be brought to the people. The proposedsolution meets the customer satisfaction needs as it Provides immediate alerts as soon as any fire is spotted which helps the forest committee to take actions sooner.
5.	Business Model (Revenue Model)	A working model which gets the live captures from satellite needed to be implemented, where the camera can monitor continuously the forest area and a working trained model which can automatically show up if any spark, fire or smoke is detected. The model has to be trained widely using largedatasets which can be fed into databases and feedbacks canbe retrieved. Thus, video processing is the main motive for detection of forest fires, and then forest management team should be present to monitor the live video and to get readyto prevent fire from further extension if any alert is produced from the trained model. Thus, this proposed model can be implanted at fire-prone area to provide quick responses and practice prevention methods.

6.	Scalability of the Solution	The device should be compatible with a minimum of 4GB RAM to support usage of various software like **Anaconda Navigator** for python and data science. Testing and trainingundergo using latest technology like **Tensor Flow** and **Keras.** Importantly satellite needed to be accessed repeatedly via camera and the data generated have to processed by Open CV and further it should be connected with a alerting system and a messaging interface to send notifications.

3.4 Problem Solution fit



Project Title: EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRES Project Design Phase-I - Solution Fit Template Team ID: PNT2022TMID17130

Define CS, fit into CC

1. CUSTOMER SEGMENT(S)

Who is your customer?
i.e. working parents of 0-5 y.o. kids

The forest resources which plays a vital role in sustaining lives on the earth, therefore to preserve them from unexpected outbreak of fire and smoke. The forest management team do need this device in fire prone areas.

6. CUSTOMER CONSTRAINTS

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

Climatic changes and the greenhouses gases are the reasons behind the destruction. Along with this the human factor to greedily use resources also play a vital reason for the forest fires.

5. AVAILABLE SOLUTIONS

Which solutions are available to the customers when they face the problem
or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Existing systems uses optical sensors for detecting forest fires. As fire is detected the sensors sends signal to the office of forest management. Among with that satellites are used to detect IR rays spotted in forest lands.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

The main problem that exists is weather and climate by releasing large number of carbon dioxide, carbon monoxide and fine particulate matter into the atmosphere.
Resulting, air pollution can cause varying range of health issues, including respiratory and cardiovascular problems.

9. PROBLEM ROOT CAUSE

What is the real reason that this problem exists? What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

The reasons possible are:
1. Due to natural causes- Lightning
2. Man-made causes- Naked flame, cigarette, electric spark

Thus, continous care and monitoring is needed to preserve natural resources to save lives.

7. BEHAVIOUR

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

When fire is detected the sytem which is implemented to monitor the forests sets the alarm to ring, that is it gives the signal through which fire management team and the forest committee tries to call off the fire. Thus, the aim is to recognise the fire as early as possible to prevent spread of fire which will cause further damage and it'll become difficult to control.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installingsolar panels, reading about a more efficient solution in the news.

The unconsious behaviour towards burned cigarette left, chances of leaving the campfire remained burnt and it can cause spread due to presence of vast dry grass spread across and electric supply being disrupted.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Wildfires can cause lot of stress since the factor that influence their direction and intensity are unpredictable and can change at anytime. People who have lived through wildfires can face dramatic mood swings, anxiety and mood-swings.

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first,fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill inthe canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

To minimize these loses, we have proposed a solution to detect early detection of forest fires by using CCTV camera surveillance, which can detect fire in indoor and outdoor activities. Thus instant alerts has to be sent to the forest management office so that they can take further actions to disrupt the daamge caused by the fire.

8. CHANNELS of BEHAVIOUR CH

ONLINE
What kind of actions do customers take online? Extract online channels from #7
OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7and use them for customer development

Online Detection: Thus the chatbot or the API can connect through internet to feed you with the current status of the forest.

Offline Detection: Thus, the forest management can send notice to the nearby residential areas or the media can bring the awareness through news, radio.

Identify strong TR & EM

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

High Priority

1. The system shall take training sets of fire images and recognize whether there is a fire orthe beginning of a fire (smoke) or if there is no fire.
2. The system shall send a notification to the admin when it recognizes a fire in the imagegiven.
3. The system shall take real inputs of camera images and determine whether the image contains a fire or not.
4. The system shall be able to take images with a variety of sizes and convert it to one fixedimage to be used throughout the application.
5. The system shall run as a service on either a Windows or Linux operating system.
6. In the event that the computer on which the system is running shuts down, the system service should start automatically when the computer restarts.

Medium Priority

The system shall provide following facility that will allow web pages that the user ispermitted to access. The system must support the following facility:

- Send alert message
- Customer data management

4.2 Non-Functional requirement

1. **Reliability**
   - The system shall be completely operational at least x% of the time.
   - Down time after a failure shall not exceed x hours.

2. **Usability**

- Customer should be able to use the system in his job for x days .
- A user who already knows what camera he is using should be able to connect and viewthat page in x seconds.

3. **Performance**

- The system should be able to support x simultaneous users.
- The mean time to view a web page over a 56Kbps modem connection shall not exceed xseconds..

4. **Security**

- The system shall provide password protected access to web pages that are to be viewedonly by users.

5. **Supportability**

- The system should be able to accommodate many camera links.
- The system web site shall be viewable from chrome or any browser.

6. **Interfaces**

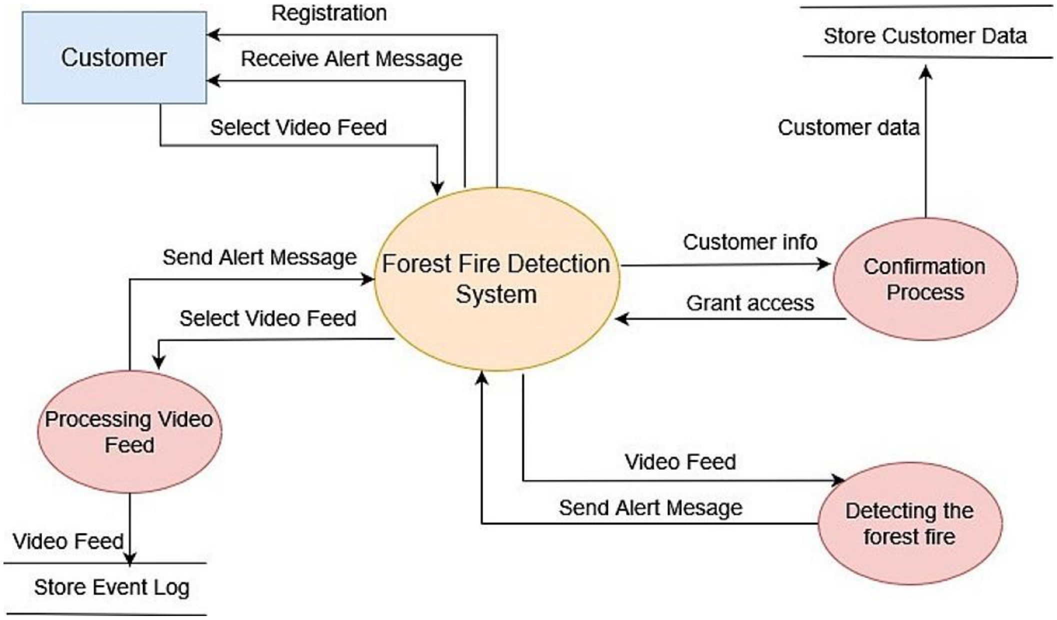
The system must interface with

- The cloudant db for customer and customer log information
- The acquired web site search engine

5. PROJECT DESIGN

5.1 Data Flow Diagram

Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored. Here, the DFD is used to represent the flow of the forest fire detection system.

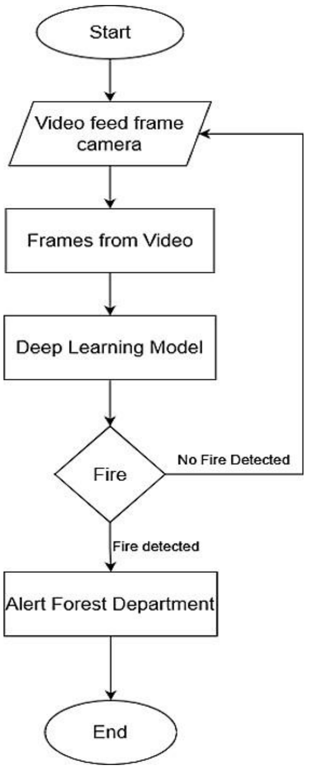


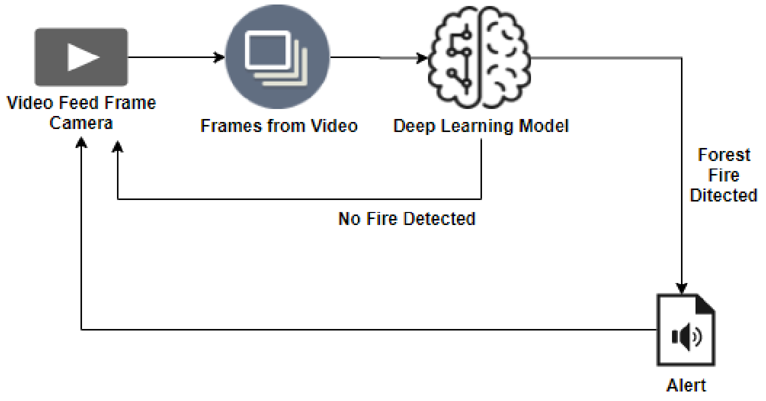
A flow diagram is a visualization of a sequence of actions, movements within a system and/or decision points. They're a detailed explanation of each step in a process, no matter the level of complexity of that process. Here, the flow diagram is used to represent the working flow of the forest fire detection system.

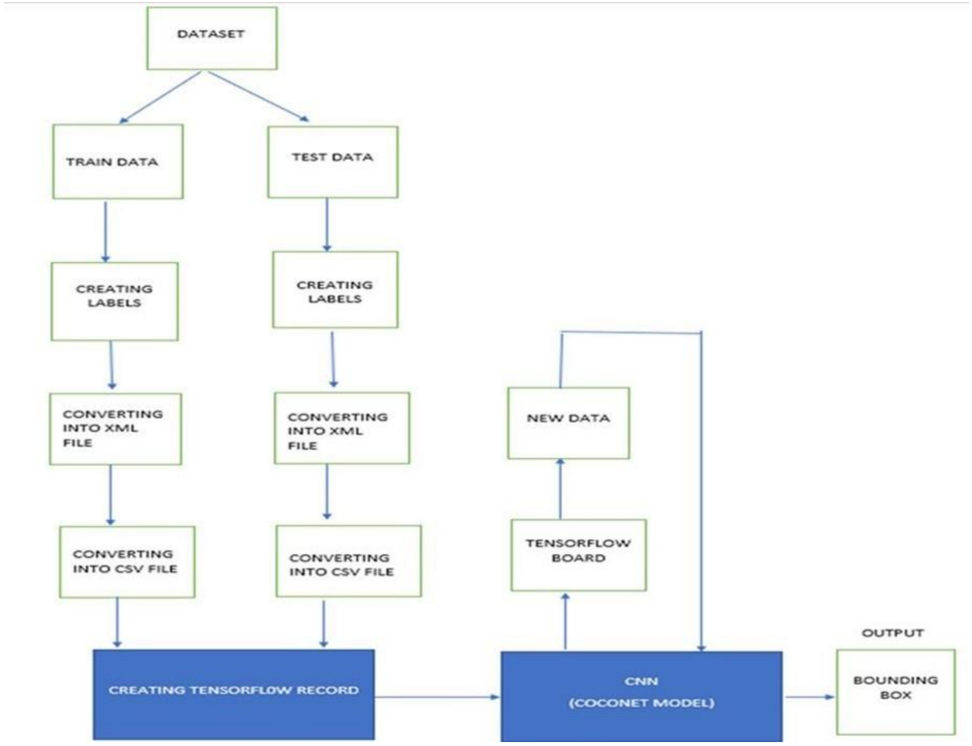
5.2 Solution & Technical Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behaviour, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed and delivered.






DATASET
TRAIN DATA
TEST DATA
CREATING LABELS
CREATING LABELS
CONVERTING INTO XML FILE
CONVERTING INTO XML FILE
CONVERTING INTO CSV FILE
CONVERTING INTO CSV FILE
CREATING TENSORFLOW RECORD
NEW DATA
TENSORFLOW BOARD
CNN (COCONET MODEL)
OUTPUT
BOUNDING BOX

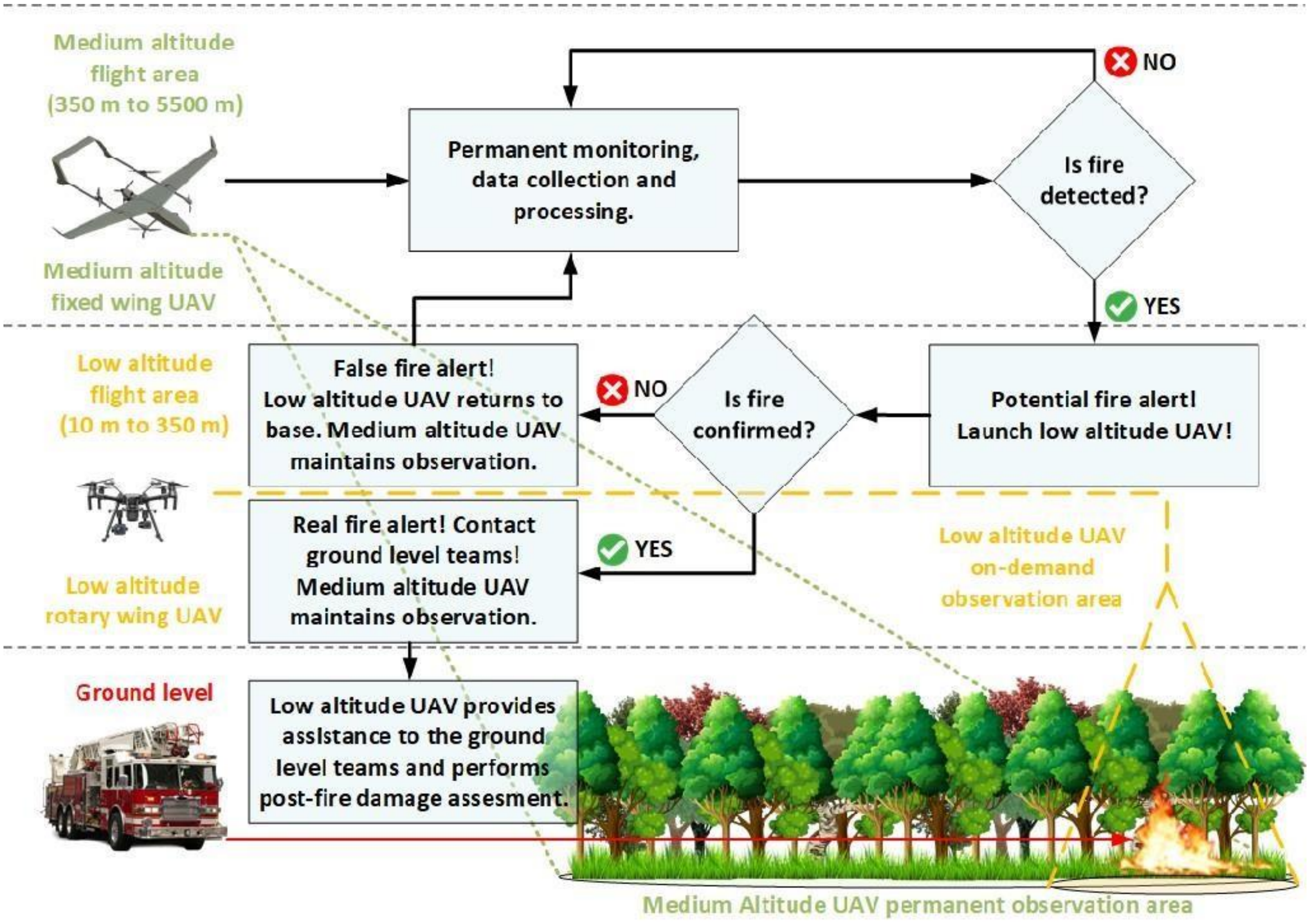

Medium altitude flight area (350 m to 5500 m)
Medium altitude fixed wing UAV
Permanent monitoring, data collection and processing.
NO
Is fire detected?
YES
Low altitude flight area (10 m to 350 m)
False fire alert! Low altitude UAV returns to base. Medium altitude UAV maintains observation.
NO
Is fire confirmed?
Potential fire alert! Launch low altitude UAV!
Real fire alert! Contact ground level teams! Medium altitude UAV maintains observation.
YES
Low altitude rotary wing UAV
Low altitude UAV on-demand observation area
Ground level
Low altitude UAV provides assIstance to the ground level teams and performs post-fire damage assesment.
Medium Altitude UAV permanent observation area

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Mobile App, database system	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Java / Python
3.	Camera	Logic for a process in the application	FPV Camera technology
4.	Smoke sensor	Logic for a process in the application	MQZ, etct
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	database system	File storage requirements	Other Storage Service or Local Filesystem
8.	Rotary--wing UAV	Purpose of firefighting used in the application	IBM Weather API, etc.
9.	EFixed--wing UAV	Purpose of weather monitoring.used in the application	Aadhar API, etc.
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Technology used
4.	Availability	Justify the availability of application (e.g. use of loadbalancers, distributed servers etc.)	Technology used
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Technology used

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Total story points	Duration	Sprint start Date	Sprint End Date (Planned)	Story points complete	
Sprint-1	20	6 Days	21 Oct 2022	27 Oct 2022	20	27 Oct 2022
Sprint-2	20	6 Days	28 Oct 2022	2 Nov 2022	20	02 Nov 2022
Sprint-3	20	6 Days	03 Nov 2022	9 Nov 2022	20	09 Nov 2022
Sprint-4	20	6 Days	10 Nov 2022	16 Nov 2022	20	16 Nov 2022

6.2 Sprint Delivery Schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password,and confirming my password.	20	High	VASIKARAN K VASAN A
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application usage.	20	High	SUBASH S VIJAY S

Sprint-1	Login	USN-3	As a user and a forest fire department staff, I will be providedwith unique login ID and password.	20	High	VASIKARAN K

Sprint-2	IBM Cloud Server	USN-4	The forest fire is detected using computer vision algorithm based cameras. These cameras continuously monitor the forest andthe data is sent to the server.	20	High	VASIKARAN K
Sprint-2		USN-5	I can fetch the details/data from thecloud server.	20	High	VASIKARAN K
Sprint-3	Data Collection	USN-6	I must gather information aboutforest fires.	20	High	VASAN A
Sprint-3		USN-7	I must draft and point out the algorithms to predict the forest fire.	20	Medium	SUBASH SVIJAY S
Sprint-	Algorithm	USN-	I must determine the precision of	20	High	VASAN A

4	Implementation	8	each algorithm.			
Sprint-4		USN-9	Extracting and assessing theDataset	20	High	VASIKARAN K
Sprint-4	Evaluating theAlgorithm	USN-10	I must determine the precision,recall and accuracy of the algorithm.	20	High	SUBASH S VIJAY S

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) periteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

7. CODING & SOLUTIONING

Here the arguments which are given inside the image data generator class are, rescale, shear_range, rotation range of image, and zoom range that we can consider for images, etc.

- Image shifts via the width_shift_range and height_shift_range arguments.
- Image flips via the horizontal_flip and vertical_flip arguments.
- Image rotations via the rotation_range argument
- Image brightness via the brightness_range argument.
- Image zoom via the zoom_range argument.

We can pass many other arguments inside the ImageDataGenerator Class.

The ImageDataGenerator class has three methods **flow ( ), flow_from_directory ( ), and flow_from_dataframe ( )** to read the images from a big numpy array and folders containing images.

flow_from_directory ( ) expects at least one directory under the

given directory path. Apply **flow_from_directory ( )**methodfor

Train folder

Now will apply the**flow_from_directory ( )**methodfortest folder.

- The directory must be set to the path where your training folders are present.
- The target_size is the size of your input images, every image will be resized to this size.
- batch_size: No. of images to be yielded from the generator per batch.
- "batch_size" in both train and test generators is to some number that divides your totalnumber of images in your train set and train set respectively.
- class_mode: Set "binary" if you have only two classes to predict, if not set to"categorical".

Keras has 2 ways to define a neural network:

- Sequential
- Function API

Now, will initialize our model.

We will be adding three layers for CNN

- Convolution layer
- Pooling layer
- Flattening layer

Adding Convolutional Layer

The convolutional layer is the first and core layer of CNN. It is one of the building blocks of a CNN and is used for extracting important features from the image.

In the Convolution operation, the input image will be convolved with the feature detector/filters to get a feature map. The important role of the feature detector is to extract the features from the image. The group of feature maps is called a feature layer.

In the convolution2D function, we gave arguments that include 32,(3,3), that refers to we are applying 32 filters of 3x3 matrix filter, and input_shape is the input image shape with RGB, here 64x64 is the size and 3 represent the channel, RGB colour images.

Activation Function:These are the functions that help us to decide if we need to activate the node or not. These functions introduce non-linearity in the networks.

Adding Pooling Layer

Max Poolingselects the maximum element from the region of the feature map covered by the filter. Thus, the output after max-pooling layer would be a feature map containing the most prominent features of the previous feature map. After the convolution layer, a pooling layer is added. Max pooling layer can be added using MaxPooling2D class. It takes the pool size as a parameter. Efficient size of the pooling matrix is (2,2). It returns the pooled feature maps.

Adding Flatten Layer

Now the pooled feature map from the pooling layer will be converted into one single dimension matrix or map, where each pixel in one single column, nothing but flattening. The flattening layer converts the multi-dimension matrix to one single dimension layer.

Adding Hidden layers

This step is to add a dense layer (hidden layer). We flatten the feature map and convert it into a vector or single dimensional array in the Flatten layer. This vector array is fed it as an input to the neural network and applies an activation function, such as sigmoid or other, and returns the output.

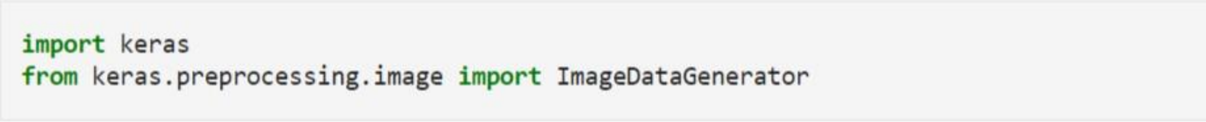
Adding output layer

This step is to add a dense layer (output layer) where you will be specifying the number of classes your dependent variable has, activation function and weight initializer as the arguments. We use add () method to add dense layers. In this layer, no need of mentioning input dimensions as we have mentions them in the above layer itself.

With both the training data defined and model defined, it's time to configure the learning process. This is accomplished with a call to the compile () method of the Sequential model class. Compilation requires 3 arguments: an optimizer, a loss function, and a list of metrics.

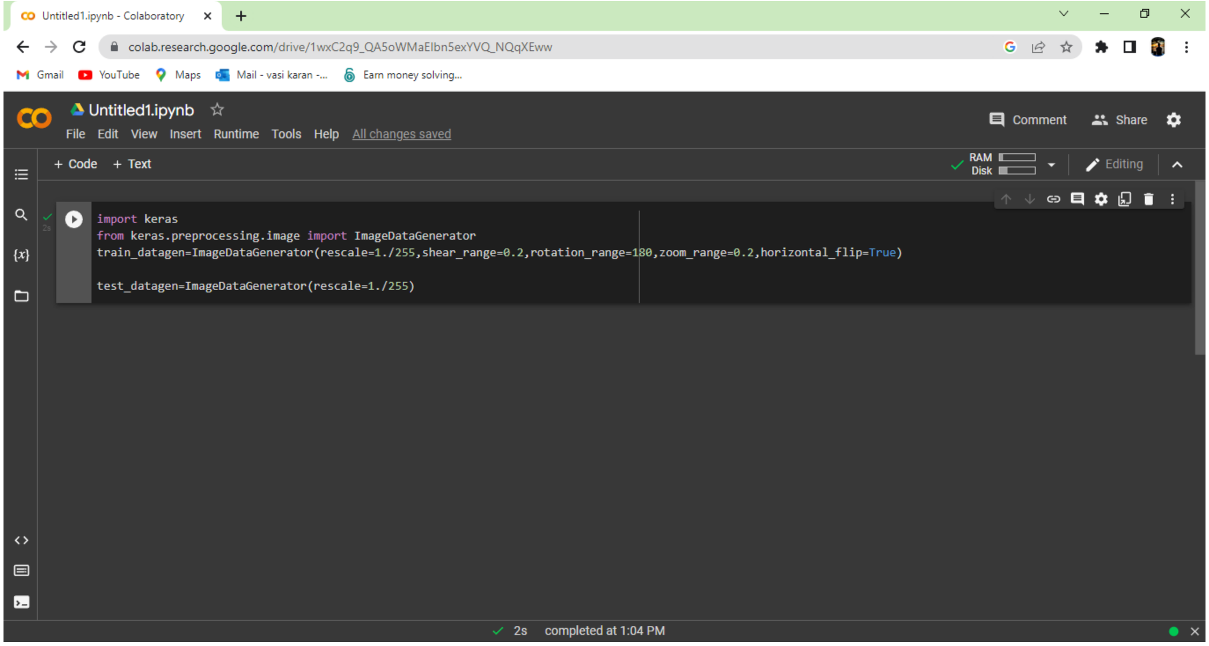
Your model is to be saved for future purposes. This saved model also is integrated with an android application or web application in order to predict something.

Importing the ImageDataGenerator



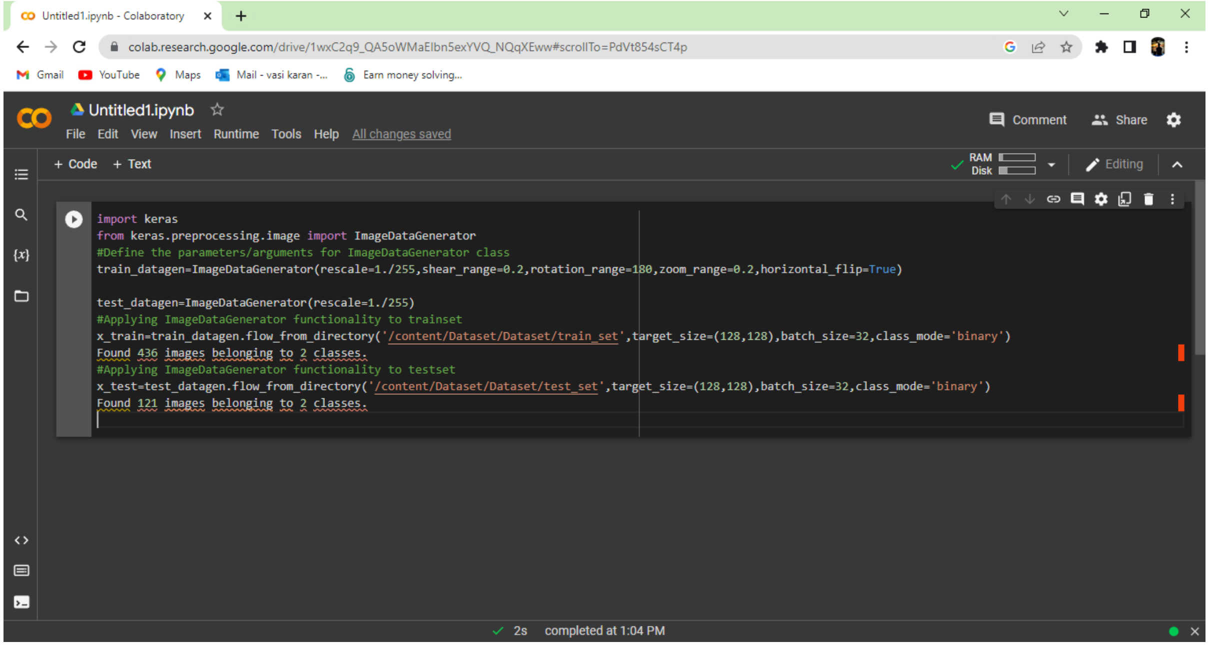
```
import keras
from keras.preprocessing.image import ImageDataGenerator
```

7.1 Feature 1


Untitled1.ipynb - Colaboratory
colab.research.google.com/drive/1wxC2q9_QA5oWMaElbn5exYVQ_NQqXEww
Gmail
YouTube
Maps
Mail - vasi karan -...
Earn money solving...
Untitled1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
2s completed at 1:04 PM

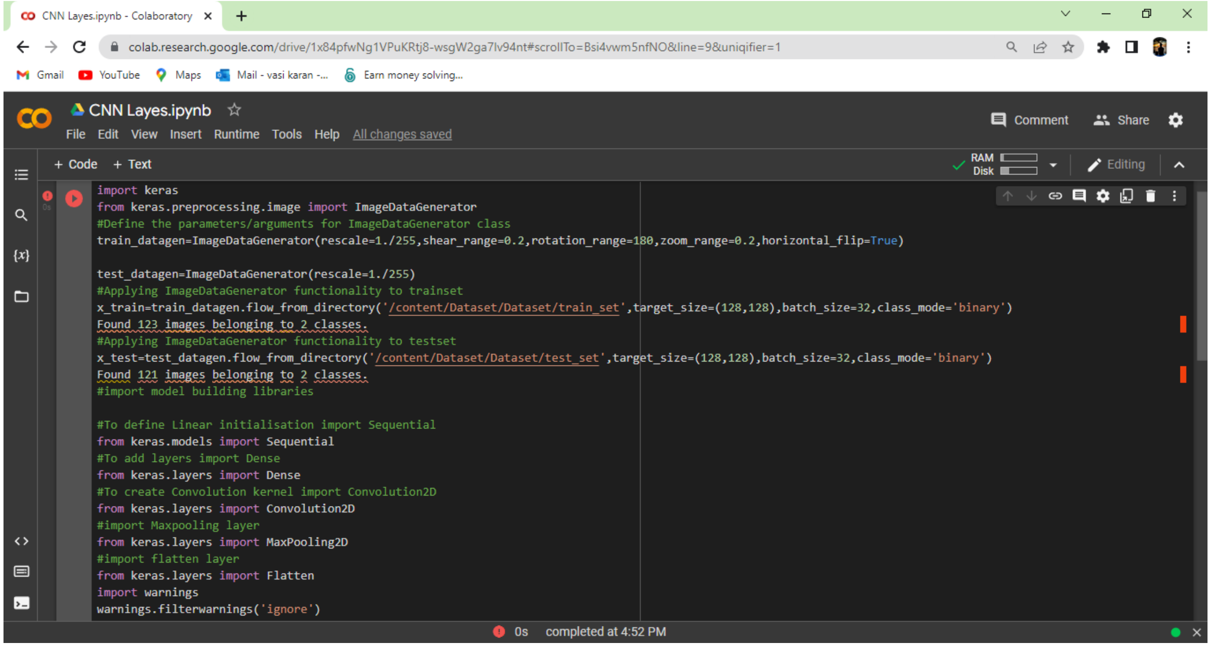
```
import keras
from keras.preprocessing.image import ImageDataGenerator
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,rotation_range=180,zoom_range=0.2,horizontal_flip=True)

test_datagen=ImageDataGenerator(rescale=1./255)
```


Untitled1.ipynb - Colaboratory
colab.research.google.com/drive/1wxC2q9_QA5oWMaElbn5exYVQ_NQqXEww#scrollTo=PdVt854sCT4p
Gmail
YouTube
Maps
Mail - vasi karan -...
Earn money solving...
Untitled1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
2s completed at 1:04 PM

```
import keras
from keras.preprocessing.image import ImageDataGenerator
#Define the parameters/arguments for ImageDataGenerator class
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,rotation_range=180,zoom_range=0.2,horizontal_flip=True)

test_datagen=ImageDataGenerator(rescale=1./255)
#Applying ImageDataGenerator functionality to trainset
x_train=train_datagen.flow_from_directory('/content/Dataset/Dataset/train_set',target_size=(128,128),batch_size=32,class_mode='binary')
Found 436 images belonging to 2 classes.
#Applying ImageDataGenerator functionality to testset
x_test=test_datagen.flow_from_directory('/content/Dataset/Dataset/test_set',target_size=(128,128),batch_size=32,class_mode='binary')
Found 121 images belonging to 2 classes.
```



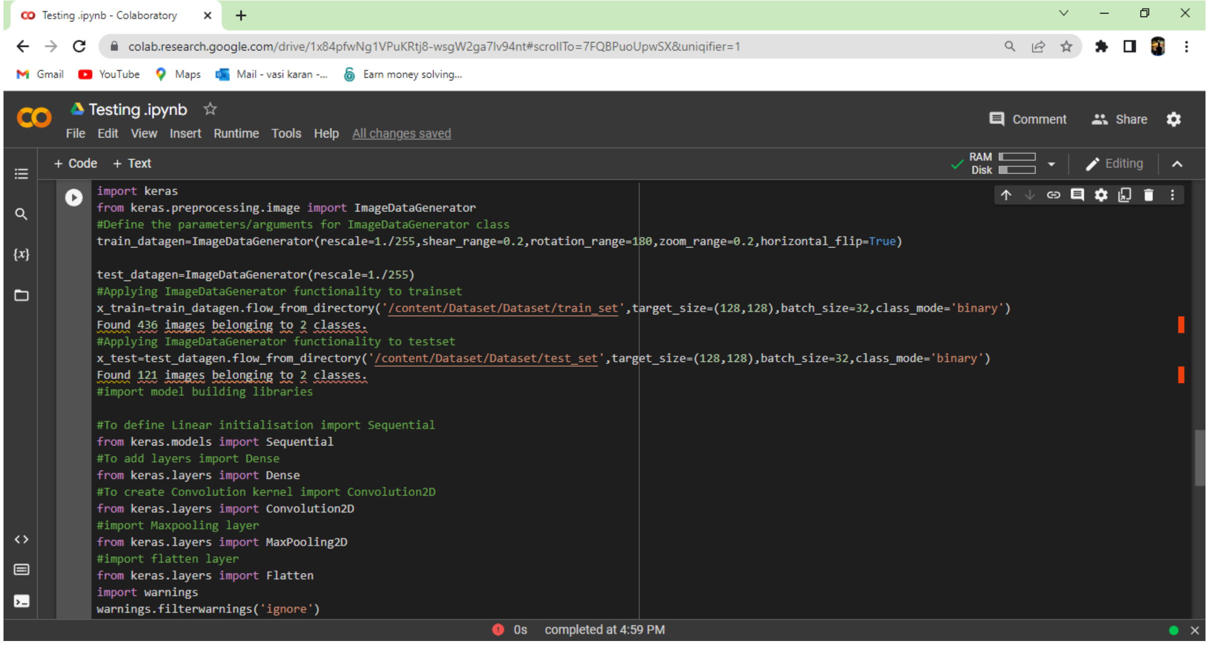
```
import keras
from keras.preprocessing.image import ImageDataGenerator
#Define the parameters/arguments for ImageDataGenerator class
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,rotation_range=180,zoom_range=0.2,horizontal_flip=True)

test_datagen=ImageDataGenerator(rescale=1./255)
#Applying ImageDataGenerator functionality to trainset
x_train=train_datagen.flow_from_directory('/content/Dataset/Dataset/train_set',target_size=(128,128),batch_size=32,class_mode='binary')
Found 123 images belonging to 2 classes.
#Applying ImageDataGenerator functionality to testset
x_test=test_datagen.flow_from_directory('/content/Dataset/Dataset/test_set',target_size=(128,128),batch_size=32,class_mode='binary')
Found 121 images belonging to 2 classes.
#import model building libraries

#To define Linear initialisation import Sequential
from keras.models import Sequential
#To add layers import Dense
from keras.layers import Dense
#To create Convolution kernel import Convolution2D
from keras.layers import Convolution2D
#import Maxpooling layer
from keras.layers import MaxPooling2D
#import flatten layer
from keras.layers import Flatten
import warnings
warnings.filterwarnings('ignore')
```

8. TESTING

8.1 Test Cases



```
import keras
from keras.preprocessing.image import ImageDataGenerator
#Define the parameters/arguments for ImageDataGenerator class
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,rotation_range=180,zoom_range=0.2,horizontal_flip=True)

test_datagen=ImageDataGenerator(rescale=1./255)
#Applying ImageDataGenerator functionality to trainset
x_train=train_datagen.flow_from_directory('/content/Dataset/Dataset/train_set',target_size=(128,128),batch_size=32,class_mode='binary')
Found 436 images belonging to 2 classes.
#Applying ImageDataGenerator functionality to testset
x_test=test_datagen.flow_from_directory('/content/Dataset/Dataset/test_set',target_size=(128,128),batch_size=32,class_mode='binary')
Found 121 images belonging to 2 classes.
#import model building libraries

#To define Linear initialisation import Sequential
from keras.models import Sequential
#To add layers import Dense
from keras.layers import Dense
#To create Convolution kernel import Convolution2D
from keras.layers import Convolution2D
#import Maxpooling layer
from keras.layers import MaxPooling2D
#import flatten layer
from keras.layers import Flatten
import warnings
warnings.filterwarnings('ignore')
```

8.2 User Acceptance Testing

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	24	0	0	24
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	1	8
Final Report Output	5	0	0	5
Version Control	2	0	0	2

9. RESULTS

9.1 Performance Metrics

This section presents the results obtained from the experimentation on the presented

intelligent system. The proposed intelligent system is implemented in MATLAB . The publicly available forest spatial data with the presence of fires are employed in training the radial basis function neural network. Consequently, forest spatial data with and without the presence of fires are fed as input to the proposed system for evaluation. The presence of fires is detected effectively by the presented intelligent system with the aid of the trained neural network. The intermediate results of the presented system are depicted From the results we can conclude that the presented intelligent system can be used for effectively detecting forest fires in the spatial data using artificial intelligence techniques.

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES :

- This approach takes advantages of both brightness and motion features of fire in images to improve the accuracy and reliability of forest fire detection.
- Fire detection systems **increase response times**, as they are able to alert the correct people in order to extinguish the fire
- This reduces the amount of damage to the property.

DISADVANTAGES

- It may result in over power consumption
- perfect accuracy may not be possible
- implementation cost is high

11. CONCLUSION

A real-time and reliable fire detection method for an early warning system is required so that an immediate response to an incident can be made effective. In this study, methods based on colour probabilities and motion features were successfully implemented to achieve this goal.

The proposed method exploits the characteristics of the colour of fire by developing a probability model using a multiple Gaussian. On the other hand, other fire characteristics, dynamic fire movement modeled with motion features based on moment invariants, were also applied.

The experiment found that the processing time required on average reached 21.70 FPS with a relatively high true positive rate of 89.92%. These results indicate that the proposed method is suitable for a real-time early warning system. Nonetheless, one of the greatest challenges in implementing the module is physically installing the camera, which may be very difficult.

Therefore, it will remain a challenge for our further research.

12. FUTURE SCOPE

The application can be enhanced by training the model with a larger dataset consisting of fires at various stages dimensions. With higher GPU memory, we could use two deep learning models for feature extraction, whose output feature vectors are concatenated and classified to offer more robustness. An R-CNN model can be used to implement fire localization along with classification. We can also expect better deep learning architectures to emerge in the future, offering better feature extraction. The application will also offer a considerably better performance when run on machines having better processing power compared to existing one of which it has been developed.

13. APPENDIX

- Glossary Forest fire detection system based on wireless sensor network on IEEE paper.
- Forest fire detection system based on satellite images on IEEE paper.

Source Code

```
import keras
from keras.preprocessing.image
importImageDataGenerator
 #Define the parameters/arguments
forImageDataGeneratorclass
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=
0.2,rotation_range=180,zoom_range=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerat
or(rescale=1./255) #Applying
ImageDataGenerator functionality
to trainset
```

```
        x_train=train_datagen.flow_from_directory('/content/Dataset/Datas
        et/train_set',target_size=(12
        8,128),batch_size=32,class_mode='binary')
        #Applying ImageDataGenerator functionality to testset
        x_test=test_datagen.flow_from_directory('/content/Dataset/Datas
        et/test_set',target_size=(128,1
        28),batch_size=32,class_mode='binary')
        #import model building libraries
#To define Linear initialisation import Sequential from keras.models import
Sequential

#To add layers import Dense from keras.layers import Dense

#To create Convolution kernel import Convolution2D from keras.layers import
Convolution2D

#import Maxpooling layer

from keras.layers import MaxPooling2D#import flatten layer

from keras.layers import Flatten import warnings warnings.filterwarnings('ignore')
#initializing the model model=Sequential()

#add convolutional layer
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))

#add maxpooling layer

model.add(MaxPooling2D(pool_size=(2,2)))#add flatten layer

model.add(Flatten())#add hidden layer

model.add(Dense(150,activation='relu'))

 #add output layer model.add(Dense(1,activation='sigmoid'))#configure the learning
process
```

```
model.compile(loss='binary_crossentropy',optimizer="adam",metrics=["accuracy"]

) #Training the model

model.fit_generator(x_train,steps_per_epoch=14,epochs=10,validation_data=x_test,
validation_steps=4)

model.save("forest1.h5") model.summary()

#import load_model from keras.model from keras.models import load_model

 #import image class from keras

  from tensorflow.keras.preprocessing import image#import numpy

import numpy as np

 #import cv2

import cv2

#load the saved model

   model = load_model("forest1.h5")
img=image.load_img('/content/Dataset/Dataset/test_set/with
fire/180802_CarrFire_010_large_700x467.jpg') x=image.img_to_array(img)

  res = cv2.resize(x, dsize=(128, 128), interpolation=cv2.INTER_CUBIC)

 #expand the image shape

  x=np.expand_dims(res,axis=0) pred=model.predict(x)

  pred

pip install twilio

pip install pygobject #import opencv library import cv2

#import numpy import numpy as np

from twilio.rest import Client#import playsound package
from playsound import playsound#load the saved model
model=load_model('forest1.h5') #define video video=cv2.VideoCapture(0)


#define the features name=['forest','with fire']while(1):

success,frame=video.read() cv2.imwrite("image.jpg",frame)
img=image.load_img("image.jpg",target_size=(64,64))

 x=image.img_to_array(img)

res = cv2.resize(x, dsize=(128, 128), interpolation=cv2.INTER_CUBIC)
x=np.expand_dims(res,axis=0)

pred=model.predict(x)p=pred[0]

print(pred) cv2.putText(frame,"predicted class
="+str(p),(100,100),cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,0))

if pred[0]==1:

#twilio account ssid account_sid='ACb252d719e2b09dbb3f44dd2f8c8be56a'

 #twilio account authentication token

  auth_token ='d401dbcf96a018136bc7ad3ded613273'
client=Client(account_sid,auth_token)

message=client.messages \.create(body='Forest Fire is detected,stay alert',

 #use twilio free numberfrom_='+1 980 414 5862',

#to numberto='+91 9080590163')

print(message.sid) print('Fire Detected')print('SMS sent!')

playsound('/tornado-siren-in-streamwood-il-35510.mp3') else:

print('No Danger')

#break cv2.imshow("image",frame)
```

```
if cv2.waitKey(1) & 0xFF == ord('a'):
break video.release()
cv2.destroyAllWindows()
```

GitHub & Project Demo Link

https://github.com/IBM-EPBL/IBM-Project-6593-1658832707