# EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRE

**TEAM ID**: PNT2022TMID28246

## 1. INTRODUCTION

Forest fires are a major environmental issue, creating economic and ecological damage while endangering human lives. There are typically about 100,000 wildfires in the United States every year. Over 9 million acres of land have been destroyed due to treacherous wildfires. 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. 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.

### 1.1 Project Overview

The idea is to create aand develop a system that can identify the effects of the forest fire and it can analyse the forest fire by advanced AI techniques and CNN Algorithm then the Prediction model is Checked and then the model is connected with Twilio account credentials of the Developer consisting of phone numbers of the persons in the surroundings of the people in the area of easy forest fire zone then an security sound alert system is developed to make a alert sound which is downloaded from internet then the entire model is deployed to the IBM Cloud account that we have created.

### 1.2 Purpose

The forest fires destroys the wildlife habitat, damages the environment, affects the climate, spoils the biological properties of the soil, etc. So the forest fire detection is a major issue in the present decade. At the same time the forest fire have to be detected as fast as possible.

## 2. LITERATURE SURVEY

### 2.1 Existing problem

Forest fires have been and still are serious problem for the European Union and for all other countries in Europe. In the year 2000, the EU has established the European Forest Fire Information system (EFFIS), which will soon become part of the European Emergency. Management Service, maintained by the Copernicus Earth Observation Programme. This system provides valuable near real-time and also historical data on the forest fires in Europe, the Middle East and North Africa. Currently EFFIS is being used and supported with data by 25 EU member states and by numerous other countries. According to the annual report of EFFIS for 2016, more than 54 000 forest fires have occurred all around Europe and they have led to nearly 376 thousand hectares of burnt areas. If we compare these values to the average values from the EFFIS reports for the period 2006-2015, the number of forest fires have decreased by 13327 or by nearly 20%. This decrease can be explained with the more severe actions and sanctions towards the arsonists and with the introduction of more advanced technical solutions for early detection of the fires. Even though their number is decreasing, the forest fires continue to be extremely devastating events and they have destroyed just 27 thousand hectares (or 6.6%) less than the average burnt areas for the period 2006-2015, according to. Confirmation for this are the devastating forest fires form 2018, which took place in the Attica region of Greece and led to more than 90 fatalities and to more than 200 injured people, as well as to the destruction to thousands of buildings. Forest Fires can be divided into 4 categories in the forests of Hungary based on tree and other vegetation species:

- underground burning, peat fire;
- fire in undergrowth or dead fallen leaves;
- fire in seedlings and saplings;
- fire in trunks andshrouds.

### 2.2 References

1. M. R. Nosouhi, K. Sood, N. Kumar, T. Wevill and C. Thapa, "Bushfire Risk Detection Using Internet of Things: An Application Scenario," in *IEEE Internet of Things Journal*, vol. 9, no. 7, pp. 5266-5274, 1 April1, 2022, doi: 10.1109/JIOT.2021.3110256.Chi Yuan, Youmin Zhang, and Zhixiang Liu , Published in: 2015, Published by NRC Research Press.

2. C. -Y. Chiang, C. Barnes, P. Angelov and R. Jiang, "Deep Learning-Based Automated Forest Health Diagnosis From Aerial Images," in IEEE Access, vol. 8, pp. 144064-144076, 2020, doi: 10.1109/ACCESS.2020.3012417.

3. O. M. Bushnaq, A. Chaaban and T. Y. Al-Naffouri, "The Role of UAV-IoT Networks in Future Wildfire Detection," in *IEEE Internet of Things Journal*, vol. 8, no. 23, pp. 16984-16999, 1 Dec.1, 2021, doi: 10.1109/JIOT.2021.3077593.Dr. Panagiotis Barmpoutis, Periklis Papaioannou, Dr. Kosmas Dimitropoulos, Dr. Nikos GRAMMALIDIS , Published in: 11 November 2020.

4. D. Q. Tran, M. Park, Y. Jeon, J. Bak and S. Park, "Forest-Fire Response System Using Deep-Learning-Based Approaches With CCTV Images and Weather Data," in IEEE Access, vol. 10, pp. 66061-66071, 2022, doi: 10.1109/ACCESS.2022.3184707.Majid Bahrepour, Nirvana Meratnia, Paul Havinga , Published in: January 2008.

5. L. Wang, A. Hawkins-Daarud, K. R. Swanson, L. S. Hu and J. Li, "Knowledge-Infused Global-Local Data Fusion for Spatial Predictive Modeling in Precision Medicine," in IEEE Transactions on Automation Science and Engineering, vol. 19, no. 3, pp. 2203-2215, July 2022, doi: 10.1109/TASE.2021.3076117.

## 2.3 Problem Statement Definition

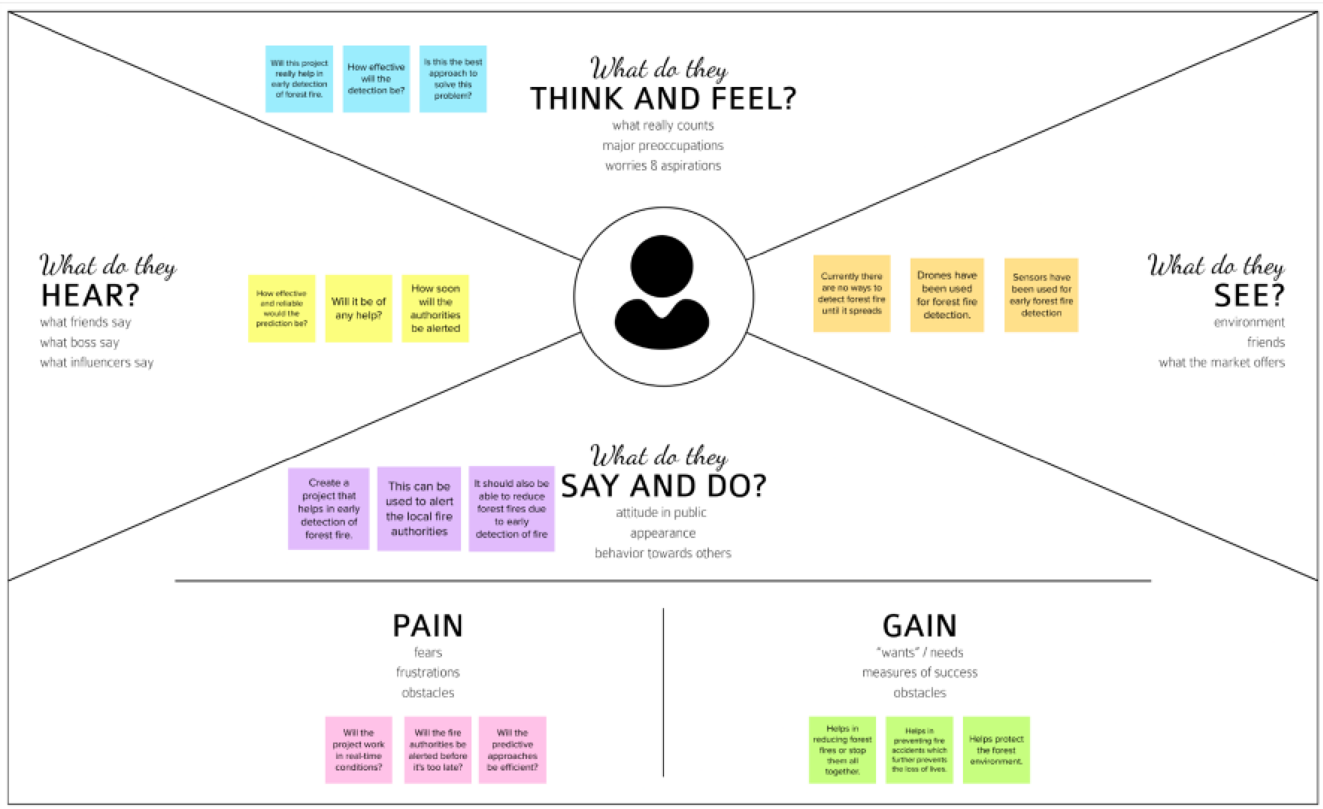
The user interacts with a web camera to read the video.
Once the input image from the video frame is sent to the model, if the fire is detected it is showcased on the console, and alerting sound will be generated and an alert message will be sent to the Authorities.

- **Data Collection.**
- Collect the dataset or create the dataset.
- **Image Preprocessing.**
- Import ImageDataGenerator Library.
- Define the parameters /arguments for ImageDataGenerator class
- Applying ImageDataGenerator on trainset and test set.
- **Model Building**
- Import the model building Libraries
- Initializing the model
- Adding CNN Layers
- Adding Hidden Layer
- Adding Output Layer
- Configure the Learning Process
- Training and testing the model
- Optimize the Model
- Save the Model
- **Video Streaming and alerting**
- OpenCV for video processing
- Creating an account in Twilio service
- Use Twilio API to send messages.

## 3. IDEATION AND PROPOSED SOLUTION

### 3.1 Empathy Map Canvas

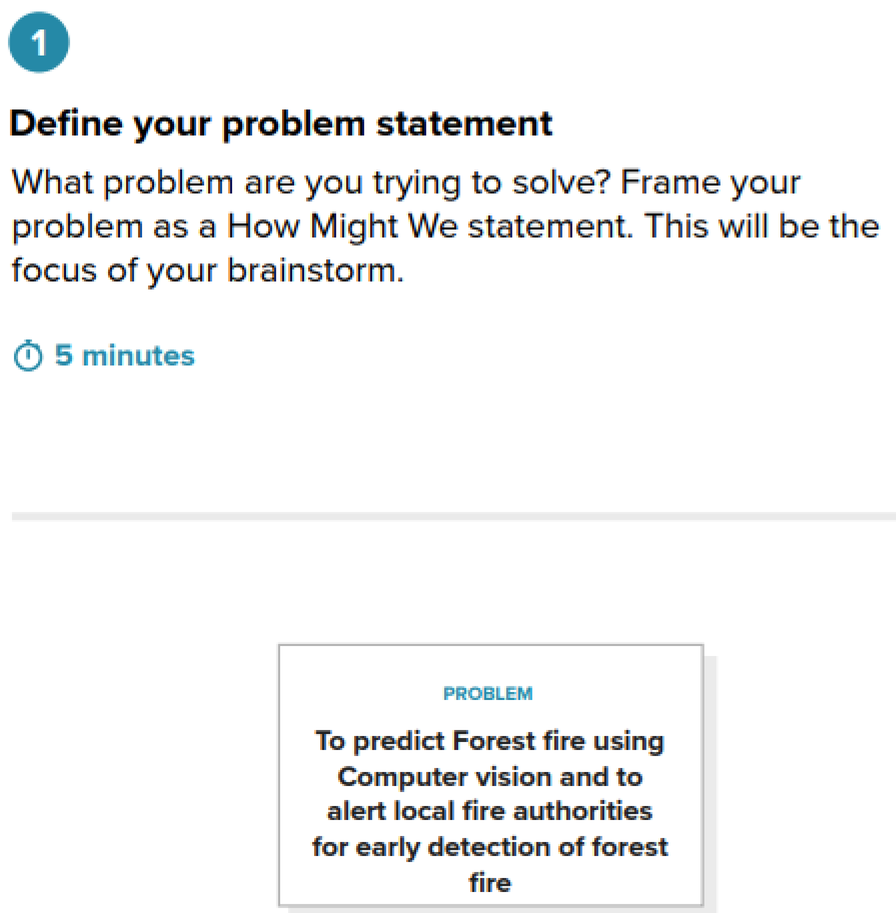
An empathy map canvas is a more in-depth version of the original empathy map, which helps identify and describe the user's needs and pain points.



**3.2 Ideation & Brainstorming**

organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.

**Step-1**: Team Gathering, Collaboration and Select the Problem Statement



**Step-2**: Brainstorm, Idea Listing and Grouping

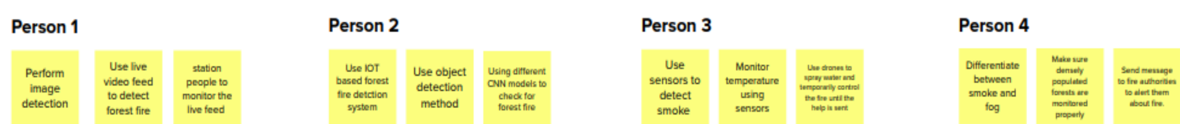
## 2

### Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

> **TIP**
>
> You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!




## 3

### Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

**Step-3**: Idea Prioritization

### 3.3 Proposed Solution

**Project Title:** EMERGING METHODS FOR EARLY FOREST FIRE DETECTION

**Project Design Phase-I - Solution Fit Template**

**Team ID:** PNT2022TMID28246

| Define CS, fit into CC | 1. CUSTOMER SEGMENT(S) CS | 6. CUSTOMER CONSTRAINTS CC | 5. AVAILABLE SOLUTIONS AS | Explore AS, differentiate |
|---|---|---|---|---|
| | Forest department officers living in forest.<br>Common people. | Satellites allow for detecting and monitoring a range of fires, providing information about the location, duration, size, temperature, and power output of those fires that would otherwise be unavailable. Satellite data is also critical for observing and monitoring smoke from the fires. | Avoid burning wastes around dry grass.<br>Obey local laws regarding open fires, including campfires<br>Have firefighting tools nearby and handy.<br>Use fire resistant roofing materials.<br>undertake technical checkups regularly.<br>Monitoring weather analytics,<br>monitoring thermal anomalies,<br>monitoring water stress and temperature rises. | |
| **Focus on J&P, tap into BE, understand RC** | **2. JOBS-TO-BE-DONE / PROBLEMS J&P** | **9. PROBLEM ROOT CAUSE RC** | **7. BEHAVIOUR BE** | **Focus on J&P, tap into BE, understand RC** |
| | Satellite remote sensing offers a useful tool for forest fire detection, monitoring, management and damage assessment. During a fire event, active fires can be detected by detecting the heat, light and smoke plumes emitted from the fires.<br>This application uses real-time satellitedata to detect and monitor forest fires (sending alerts to mobile | Forest fires cause lots of damage, some of them are – loss of wildlife habitat, extinction of plants and animals, destroys the nutrient rich top soil, reduction in forest cover, loss of valuable timber resources, ozone layer depletion, loss of livelihood for tribal people and poor people, increase in global warming. | Climate change should be monitored.<br>Hot areas should be monitored clearly. | |
| **Identify strong TR & EM** | **3. TRIGGERS TR** | **10. YOUR SOLUTION SL** | **8. CHANNELS of BEHAVIOUR CH** | **Identify strong TR & EM** |
| | Human-caused fires result from campfires left unattended, the burning of debris, equipment use and malfunctions, negligently discarded cigarettes, and intentional acts of arson.<br><br>**4. EMOTIONS: BEFORE / AFTER EM**<br>**Before :** Unable to detect small sparks. Camera should always be in motion.<br>**After :** Able to detect small sparks. 360 view of camera is used. | For this problem we use image processing and video analysis so by using satellite image processing we can able to find the fire at the early stage and stop spreading fire in the forest. This model is mainly build by using CNN and machine learning and deep learning | **ONLINE:** Collect the date and form a dataset in order to compare the flames regions for forest fire detection.<br><br>**OFFLINE:** In case of forest fire detection the information is sent to forest authorities so that they will prevent it at ease. | |

### 3.4 Proposed Solution

| S.No | Parameter | Description |
|---|---|---|
| **1** | Problem Statement (Problem to be solved) | Forest fires are a major environmental issue, creating economic and ecological damage while endangering human lives. It is difficult to predict and detect Forest Fire in a sparsely populated forest area. So, it is necessary to detect the fire in an early stage to control it. |
| **2** | Idea / Solution description | Use computer vision methods for recognition and detection of smoke or fire, based on the still images or the video input from the drone cameras. |
| **3** | Novelty / Uniqueness | When the fire is detected, the station will get a notification via message and an alarm system will be activated automatically to alert the user. |
| **4** | Impact on society | By using this system it reduces global warming which causes major climatic changes and prevents wild animals and birds from danger by this method the smoke due to fire is also prevented which causes various effects on Nature. |
| **5** | Business Model (Revenue Model) | This system is used to predict the forest fire at the early stage, prevent the wild lives from fire and prevents global warming. This model is more reliable and can produce accurate detection of forest fire. |
| **6** | Scalability of the Solution | Computer vision models enable land cover classification and smoke detection from satellite and ground cameras. |

# 4. REQUIREMENT ANALYSIS

## 4.1 Functional requirement

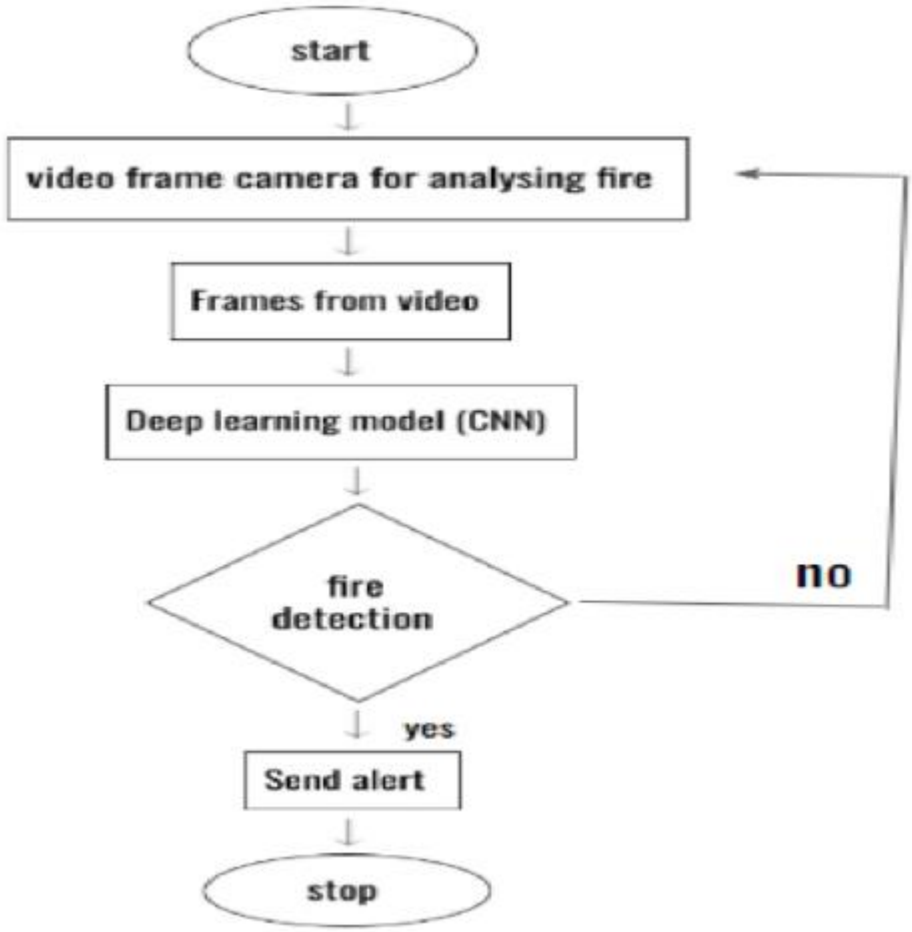
| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|---|---|---|
| FR-1 | User Registration | Registration through Gmail |
| FR-2 | User Confirmation | Confirmation via Email<br>Confirmation via OTP |
| FR-3 | User Login | Login using credentials |
| FR-4 | User Search | Search for Info on forest fire occurrence |
| FR-5 | User Profile | User shall be given a live feed of the forest |
| FR-6 | User Application | User is alerted if there is a forest fire occurrence in their surroundings |

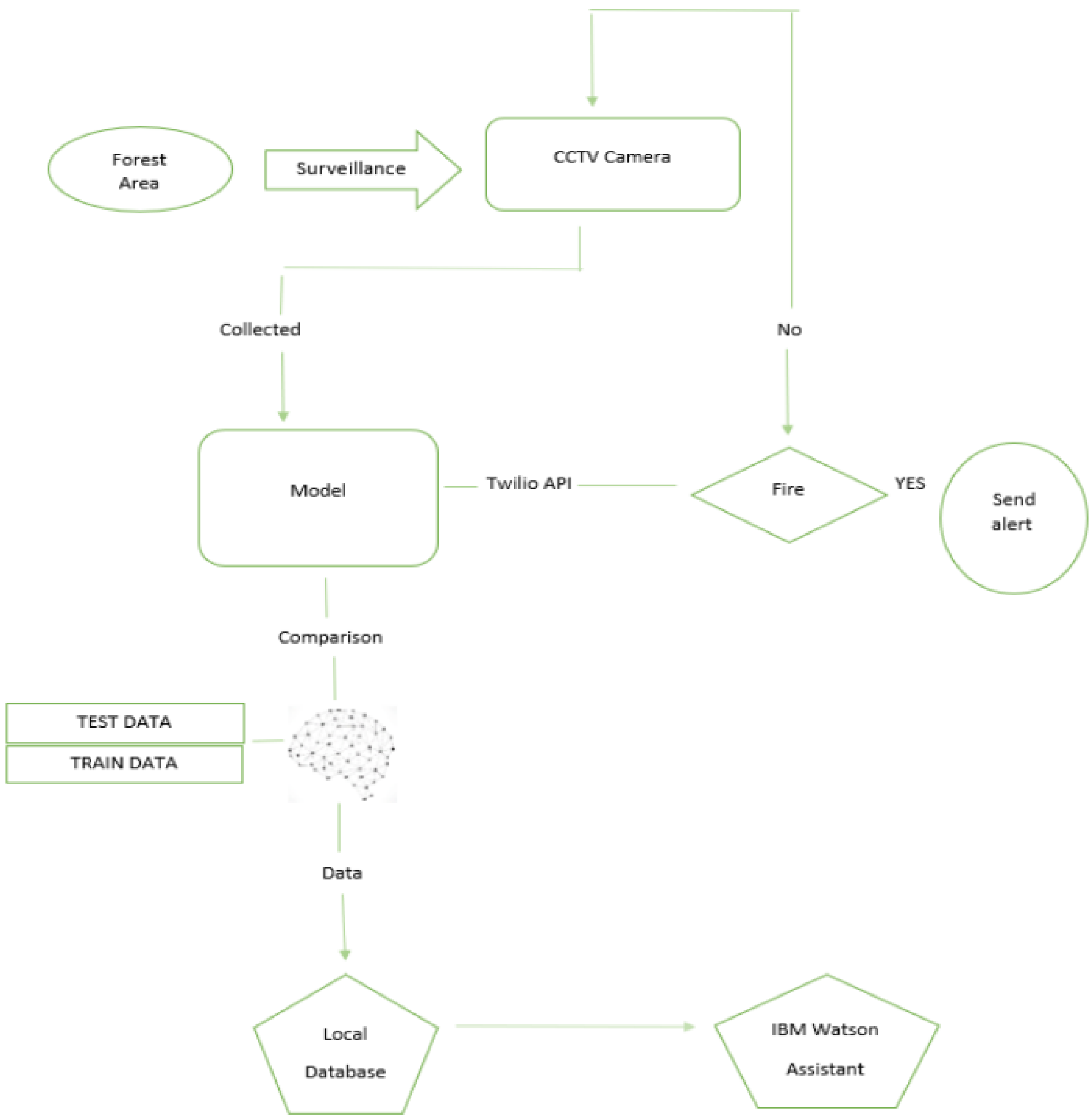
## 4.2 Non-Functional requirements

| FR No | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | Usability | Alerts according to the user location |
| NFR-2 | Security | Instant live feed with alert of the situation |
| NFR-3 | Reliability | The prediction of the forest fire is 87% accurate |
| NFR-4 | Performance | The feed and the alert message an immediate action without a lag |
| NFR-5 | Availability | The application gives alerts and live feeds 24/7 |
| NFR-6 | | Early detection and alerting users are done efficiently and in a faster means |

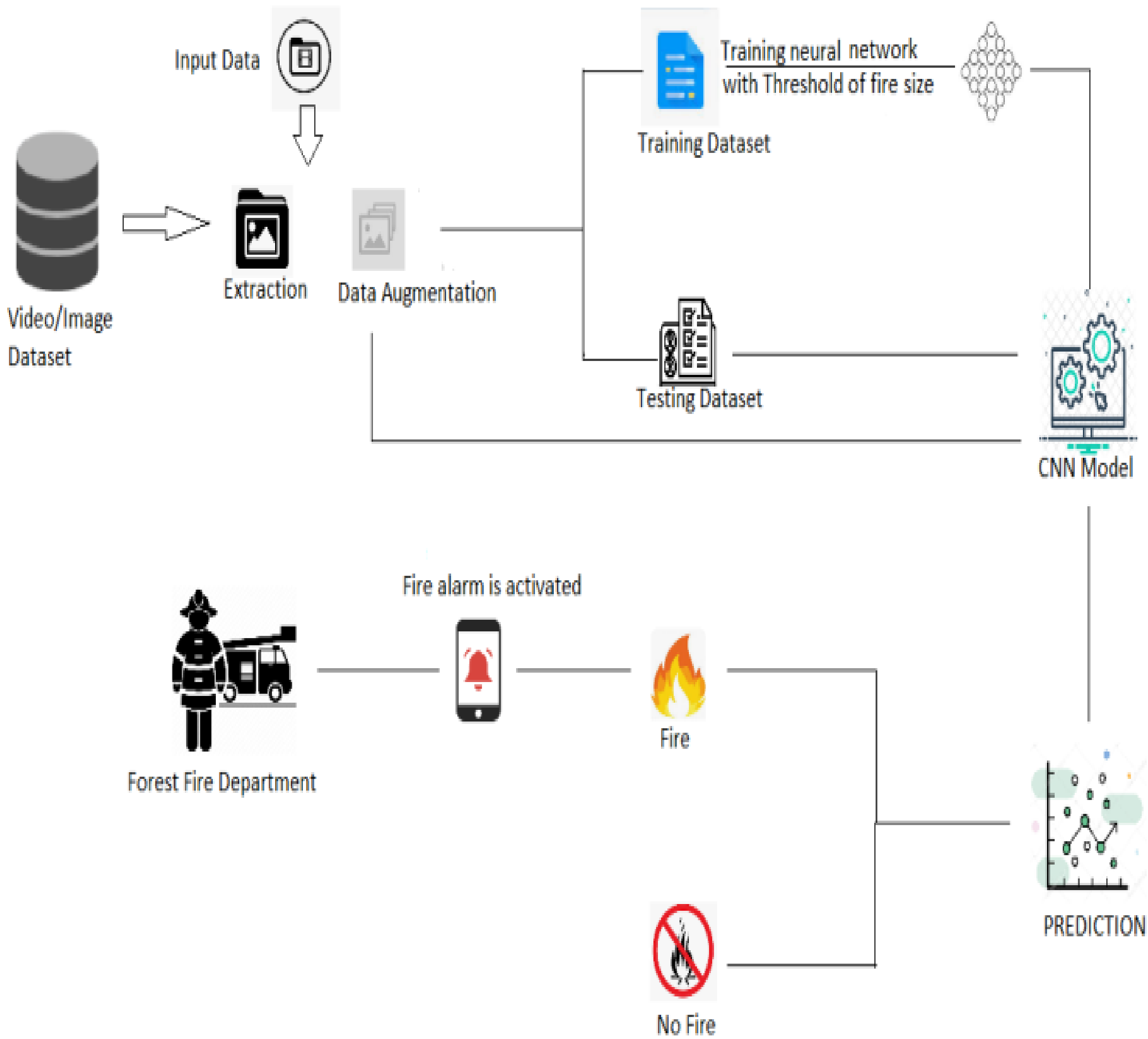
## 5. PROJECT DESIGN

### 5.1 Data Flow Diagrams




Forest Area
Surveillance
CCTV Camera
Collected
No
Model
Twilio API
Fire
YES
Send alert
Comparison
TEST DATA
TRAIN DATA
Data
Local Database
IBM Watson Assistant

### 5.2 Solution & Technical Architecture



| S.No | Component | Description | Technology |
|---|---|---|---|
| **1** | User Interface | The user uses the console to access the interface | Python/HTML ,CSS , Javascript and react.Js |
| **2** | Input | Video Feed | Web Camera/Video on a site |
| **3** | Conversion | Video inputted is converted into Frames | Frame Converter |
| **4** | Feeding the Model | The Frames are sent to the Deep learning model | Our Model |
| **5** | Dataset | Using Test set and train set , train the model | Data set from Cloud Storage , Database |
| **6** | Cloud Database | The model is trained in the cloud more precise with detections more images can be added later on. | IBM Cloudant ,Python Flask. |
| **7** | Infrastructure (Server / Cloud), API | Application Deployment on Local System / Cloud Local ,Cloud Server Configuration , Twilio API to send messages | Java/python ,React.Js ,JavaScript ,HTML ,CSS ,IBM Cloud ,OPEN CV ,Anaconda Navigator ,Local. |

**Table-2: Application Characteristics:**

| S.No | Characteristics | Description | Technology |
|---|---|---|---|
| 1 | Open-Source Frameworks | Python Flask framework is used | Technology of Opensource framework |
| 2 | Security Implementations | Mandatory Access Control (MAC) and Preventative Security Control is used | e.g. SHA-256, Encryptions, IAM Controls, OWASP etc. |
| 3 | Scalable Architecture | High scalability with 3-tier architecture | Web server – HTML ,CSS ,JavaScript Application server – Python , Anaconda Database server –IBM DB2 |
| 4 | Availability | Use of load balancing to distribute traffic across servers | IBM load balancer |
| 5 | Performance | Enhance the performance by using IBM CDN | IBM Content Delivery Network |

**5.3 User Stories**

**5.4**

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Environmentalist | Collect the data | USN-1 | As an Environmentalist ,it is necessary to collect the data of the forest which includes temperature,humidity,wind and rain of the forest | It is necessary to collect the right data else the prediction may become wrong | High | Sprint-1 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | USN-2 | Identify algorithms that can be used for prediction | To collect the algorithm to identify the accuracy level of each algorithms | Medium | Sprint-2 |
| | | USN-3 | Identify the accuracy of each algorithms | Accuracy of each algorithm-calculated so that it is easy to obtain the most accurate output | High | Sprint-2 |
| | | USN-4 | Evaluate the Dataset | Data is evaluated before processing | Medium | Sprint-1 |
| | | USN-5 | Identify accuracy,precision ,recall of each algorithms | These values are important for obtaining the right output | High | Sprint-3 |
| | | USN-6 | Outputs from each algorithm are obtained | It is highly used to predict the effect and to take precautionary measures | High | Sprint-4 |

## 6. PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning & Estimation

| Sprint | Functional Requirement (Epic) | User story number | User story/Task | Story points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | Collecting Data | USN-2 | Collecting the data is really important for the detection of fire. | 10 | high | All members |
| Sprint-2 | Creating and Saving the Model | USN-2 | As a developer creating and saving the model developed for estimation of fire | 10 | high | All members |
| Sprint-3 | Testing model | USN-2 | As a developer creating and saving the model developed for estimation of fire | 10 | high | All members |
| Sprint-3 | Video Analysis | USN-3 | | 10 | Medium | All members |
| Sprint-4 | Twilio Message Service | USN-3 | | 10 | Low | All members |
| Sprint-4 | Alert Sound and Message | USN-4 | Sending Alert text message using registered twilio account and produce output sound alert alarm | 10 | Low | All members |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-4 | Train Model on cloud | USN-5 | Application Deployment on Local System/ Cloud Local Server Configuration: Cloud Server Configuration and to train the deep learning model in IBM Cloud | 10 | Medium | All members |

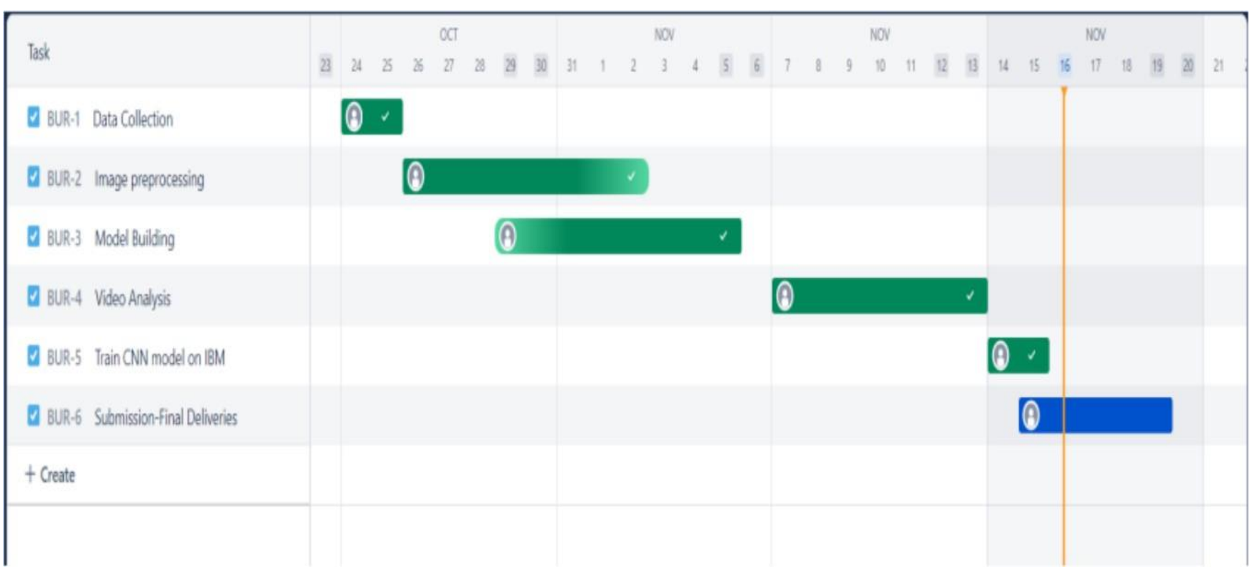
### 6.2 Sprint Delivery Schedule

| Sprint | Total Story Points | Duration | Sprint Start Date | Sprint End Date(Planned) | Story Points Completed (as on planned end date) | Sprint Release Date (Actual) |
|---|---|---|---|---|---|---|
| Sprint-1 | 20 | 6 days | 24 Oct 2022 | 29 Oct 2022 | 20 | 29 Oct 2022 |
| Sprint-2 | 20 | 6 days | 31 Oct 2022 | 05 Nov 2022 | 20 | 05 Nov 2022 |
| Sprint-3 | 20 | 6 days | 07 Nov 2022 | 12 Nov 2022 | 20 | 12 Nov 2022 |
| Sprint-4 | 20 | 6days | 14 Nov 2022 | 19 Nov 2022 | 20 | 19 Nov 2022 |

The following table shows the sprint works assigned to the members along with the priority and story points assigned with the functional requirements with regards to user story.

### 6.3 Reports from JIRA

Burndown Chart:



## 7. CODING & SOLUTION

### 7.1 Feature 1

In Feature 1 module we have made data collection and Image preprocessing for and Model training.

**importing Required Libraries:**

import keras from keras.preprocessing.image

import ImageDataGenerator

import matplotlib.pyplot as plt

import numpy as np batch_size = 32

**image resizing and preprocessing :**

train_datagen = ImageDataGenerator( shear_range=0.2, rotation_range=180, zoom_range=0.2, horizontal_flip=True, ) val_datagen = ImageDataGenerator( rescale=1./255 )

train_generator = train_datagen.flow_from_directory( 'train_set/', target_size=(150, 150), batch_size=batch_size, class_mode='binary' )

val_generator = val_datagen.flow_from_directory( 'test_set/', target_size=(150, 150), batch_size=batch_size, class_mode='binary' )

**Creating the sequential model**:

from keras.models

import Sequential from keras.layers

import Convolution2D from keras.layers

import MaxPooling2D from keras.layers

import Activation from keras.layers

import Dropout from keras.layers

import Flatten from keras.layers

import Dense model=Sequential()

model.add(Convolution2D(32,(3,3),input_shape=(150,150,3))) #Convolutional 2D Layer model.add(Activation('relu'))

model.add(MaxPooling2D(pool_size=(2,2))) # MaxPooling Layer

model.add(Flatten()) #Flatten Layer to make a array

model.add(Dense(150))

model.add(Activation('relu'))

model.add(Dropout(0.5))

model.add(Dense(1))

model.add(Activation('sigmoid'))

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

**Model summary:**

model.summary()

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 148, 148, 32)      896
_________________________________________________________________
activation (Activation)      (None, 148, 148, 32)      0
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 74, 74, 32)        0
_________________________________________________________________
flatten (Flatten)            (None, 175232)            0
_________________________________________________________________
dense (Dense)                (None, 150)               26284950
_________________________________________________________________
activation_1 (Activation)    (None, 150)               0
_________________________________________________________________
dropout (Dropout)            (None, 150)               0
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 151
_________________________________________________________________
activation_2 (Activation)    (None, 1)                 0
=================================================================
Total params: 26,285,997
Trainable params: 26,285,997
Non-trainable params: 0
```

### 7.2 Feature 2

```
import cv2 import
 numpy as np
from keras.preprocessing import
image from keras.models import
load_model from twilio.rest import
Client from playsound import
playsound
model=load_model('forest1.h5')
video=cv2.VideoCapture(0)
name=['forest','with fire']
while(True):
  ret,frame=video.read()
  cv2.imshow('frame',frame)
  cv2.imwrite('image.jpg',frame)
  img=image.load_img('train_set/forest/NoFire
(1).bmp',target_size=(64,64))
  x=image.img_to_array(img)
  x=np.expand_dims(x,axis=0)
  pred=model.predict(x)
    index=np.argmax(pred)
    if index==0:

account_sid='AC50d663c8a7c2d8b35b1fc09dfda93bda'
      auth_token='86f345babfa094d1015a0e1137dbb679' client
      =Client(account_sid,auth_token)
      message=client.messages \
      .create(body='-------Fire is detected,Stay Alert !!! ------------',
              from_='+19457581434',to='+916369 659 356')
      print(message.sid)
      print('Fire detected')
      print("Alert Message sent!")
      playsound('tornado-siren.mp3')

   else:
      print('No Danger')
      cv2.imshow("image.jpg",frame)
      if cv2.waitkey(2)&0xff == ord('q'):
           break
video.release()
cv2.destroyAllWindows()
```

```
SM200b2510df1efe889005ac80731425ff
Fire detected
Alert Message sent!
```

```
In [ ]:
```

## 8.TESTING

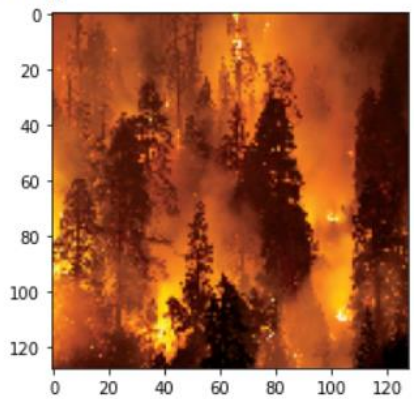
### 8.1 Test Cases & User Acceptance Testing

**Testing with input video recording from user end:**

```
def message(val):
 if val==1:
   from twilio.rest import Client
   print('Forest fire')
   account_sid='AC882fb904c019982a8338bfce6209f0b7'
   auth_token='69359c9ed0bffcfd52e1d316866304e2'
   client=Client(account_sid,auth_token)
   message=client.messages \
    .create(
      body='forest fire is detected, stay alert',
      #use twilio free number
      from_='+18318513711',
      #to number
      to='+91 89397 68258')
   print(message.sid)
   print("Fire detected")
   print("SMS Sent!")
 elif val==0:
   print('No Fire') from matplotlib import pyplot as plt
#import load model from keras.model
from keras.models import load_model
#import image from keras
from tensorflow.keras.preprocessing import image
img1 = image.load_img('/content/drive/MyDrive/IBM_forestfire/Dataset/Dataset/test_set/with fire/Wild_
fires.jpg',target_size=(128,128))
Y = image.img_to_array(img1)
x = np.expand_dims(Y,axis=0)
val = model.predict(x)
plt.imshow(img1)
plt.show()
message(val)
img2 = image.load_img('/content/drive/MyDrive/IBM_forestfire/Dataset/Dataset/test_set/with fire/horse
shoe_bay_fire.jpg',target_size=(128,128))
Y = image.img_to_array(img2)
x = np.expand_dims(Y,axis=0)
val = model.predict(x)
plt.imshow(img2)
plt.show()
message(val)
```

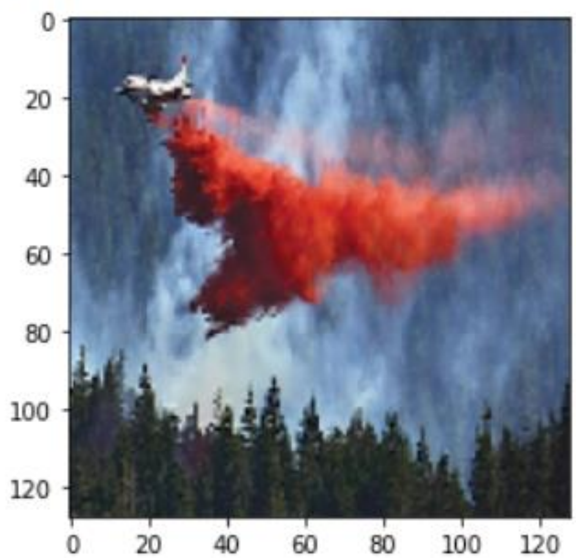
**OUTPUT:**

```
1/1 [==============================] - 0s 157ms/step
```

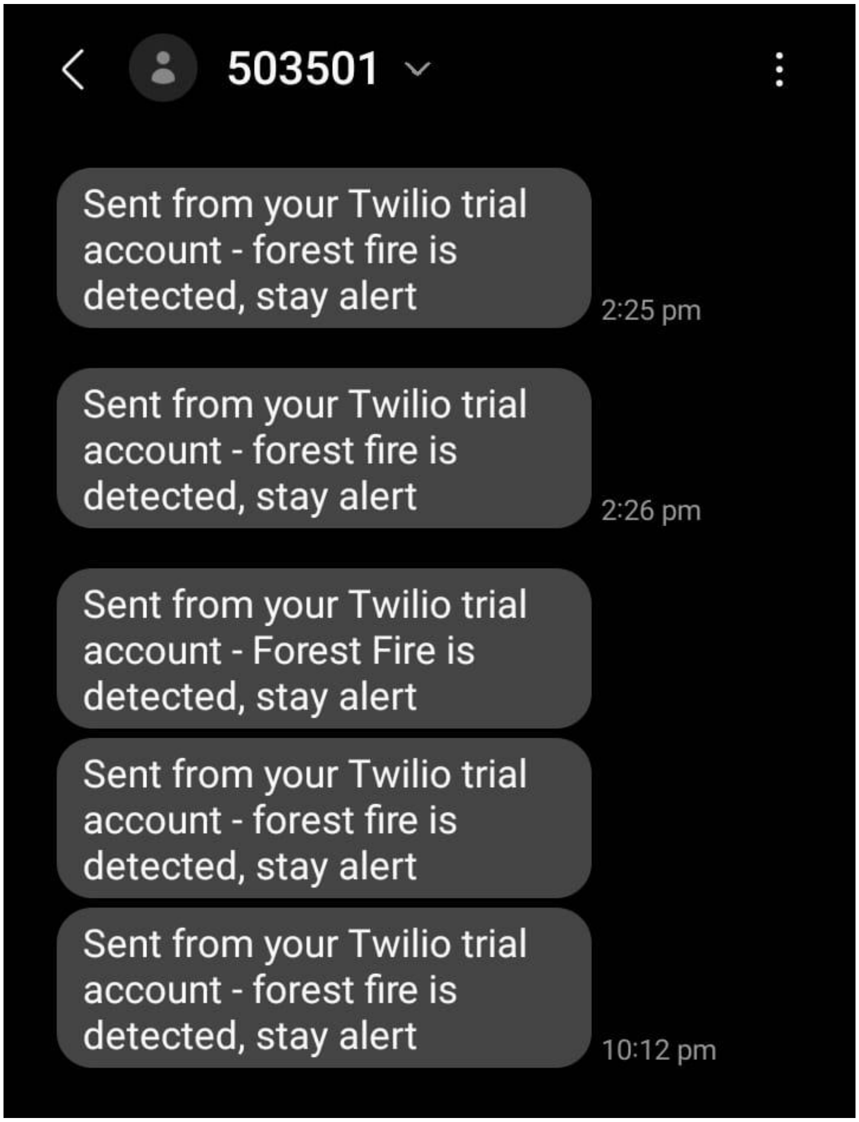


```
Forest fire
SMaad17a5770e383ee435a41b939a5b124
Fire detected
SMS Sent!
```

```
1/1 [==============================] - 0s 89ms/step
```



```
Forest fire
SM1fa39487133a96009462a9819b79c093
Fire detected
SMS Sent!
```


503501
Sent from your Twilio trial account - forest fire is detected, stay alert
2:25 pm
Sent from your Twilio trial account - forest fire is detected, stay alert
2:26 pm
Sent from your Twilio trial account - Forest Fire is detected, stay alert
Sent from your Twilio trial account - forest fire is detected, stay alert
Sent from your Twilio trial account - forest fire is detected, stay alert
10:12 pm

## 9. Result

### 9.1 Performance Metrics

```
loss: 0.3438 - accuracy: 0.8483 - val_loss: 0.2485 - val_accuracy: 0.958

loss: 0.3816 - accuracy: 0.8483 - val_loss: 0.2569 - val_accuracy: 0.958

loss: 0.4068 - accuracy: 0.8391 - val_loss: 0.2547 - val_accuracy: 0.958

loss: 0.3312 - accuracy: 0.8437 - val_loss: 0.2601 - val_accuracy: 0.950

loss: 0.5621 - accuracy: 0.8368 - val_loss: 0.2679 - val_accuracy: 0.958
```

## 10. ADVANTAGES & DISADVANTAGES

**Advantages :**

- Easily detect and Estimate the Forest Fire.
- Most Accurate
- Flexible Model which can give maximized outcome
- No Specific Requirements needed to implement the model

**Disadvanatges:**

- Training model is time consuming process.
- Error in CV can cause damage to camera
- Access of camera are prohibited due to personal issues

## 11. CONCLUSION :

Thus we have constructed a model that can identify the effects of the forest fire and it can analyse the forest fire by advanced AI techniques and CNN Algorithm then the Prediction model is Checked and then the model is connected with Twilio account credentials of the Developer consisting of phone numbers of the persons in the surroundings of the people in the area of easy forest fire zone then an security sound alert system is developed to make a alert sound which is downloaded from internet then the entire model is deployed to the IBM Cloud account that we have created was made with the studies we have done.

## 12. FUTURE SCOPES :

1. It can be developed as a Web or Android Application.
2. In future Alternate Advanced technologies can be Implemented.
3. The Identification and tracking system can be implemented if possible.

## 13. APPENDIX :

**Source Code :** https://github.com/IBM-EPBL/IBM-Project-23510-1664355259/blob/main/Final%20Deliverables/Final_Code.ipynb

**GitHub & Project Demo Link:**

**Github:** https://github.com/IBM-EPBL/IBM-Project-23510-1664355259

**Demo Link: https**://drive.google.com/file/d/1wxVAC-xULMvsAENdGO_CIGMSS8gLn6B6/view