PROJECT DEVELOPMENT PHASE

SPRINT-III

VIDEO ANALYSIS

Date	09 November 2022
Team ID	PNT2022TMID39882
Project Name	Emerging Methods for Early Detection of Forest Fires
MaximumMarks	8 Marks

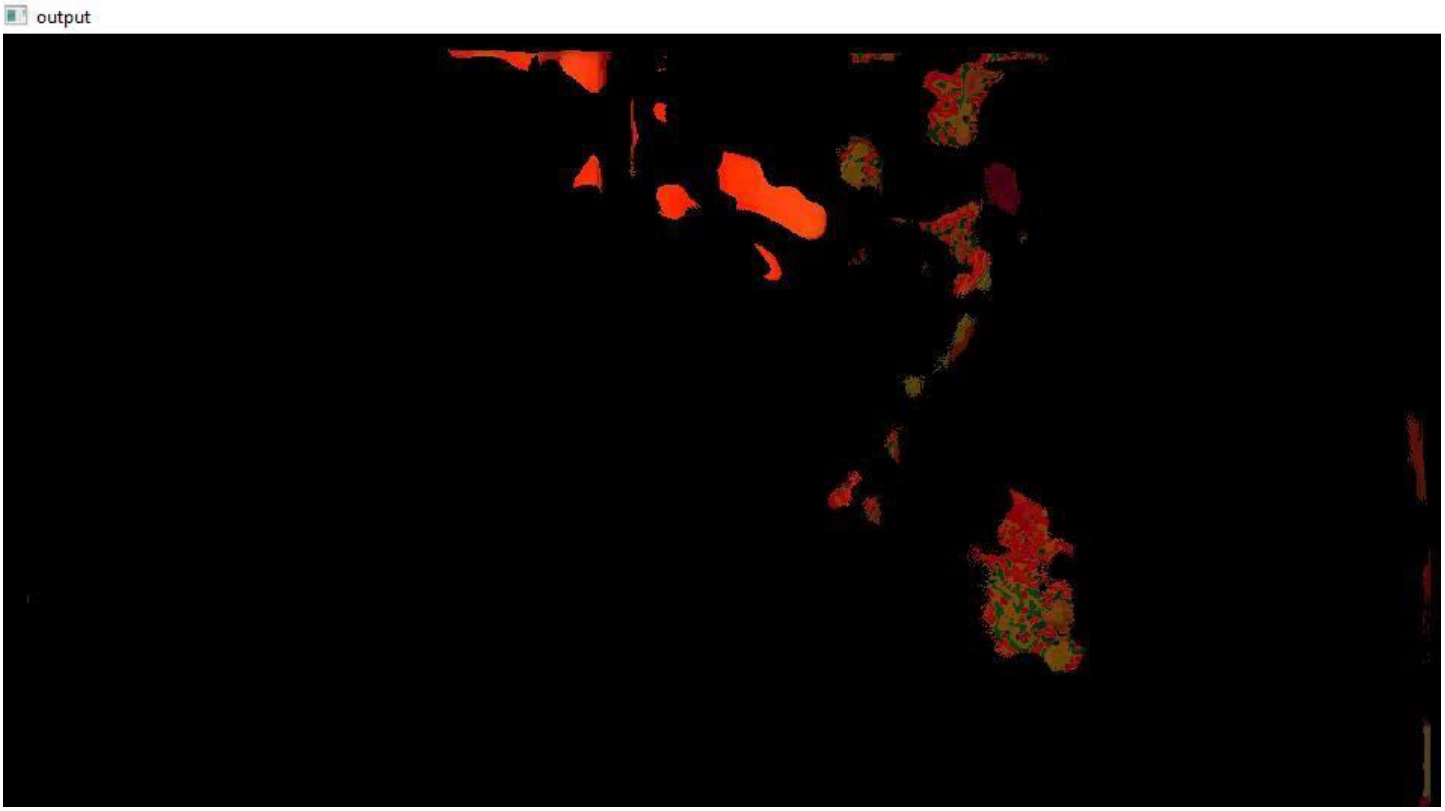
OpenCv for video processing:

```
import cv2 import
numpy as np
#import smtplib
#import playsound
#import threading

Alarm_Status = False
Email_Status = False
Fire_Reported = 0

#def play_alarm_sound_function():
        #while True:
```

```
                #playsound.playsound('alarm-sound.mp3',True) #def
send_mail_function():

    #recipientEmail = "reenu8602@gmail.com"
   # recipientEmail = recipientEmail.lower()

   # try:
        #server = smtplib.SMTP('smtp.gmail.com', 587)
        #server.ehlo()
        #server.starttls()
        #!server.login("swethathanam52@gmail.com", 'swethaanu3')
        #server.sendmail('reenu8602@gmail.com)', recipientEmail, "Warning A
Fire Accident has been reported on ABC ")
        #print("sent to {}".format(recipientEmail))
       # server.close()
   # except Exception as e:
    # print(e)


video = cv2.VideoCapture("video.mp4") # If you want to use webcam use
Index like 0,1.

while True:
    (grabbed, frame) = video.read()
if not grabbed:
        break

    frame = cv2.resize(frame, (960, 540))

    blur = cv2.GaussianBlur(frame, (21, 21), 0)     hsv
= cv2.cvtColor(blur, cv2.COLOR_BGR2HSV)

    lower = [18, 50, 50]     upper = [35,
255, 255]     lower = np.array(lower,
dtype="uint8")     upper = np.array(upper,
```

```
dtype="uint8")     mask =
cv2.inRange(hsv, lower, upper)

    output = cv2.bitwise_and(frame, hsv, mask=mask)

    no_red = cv2.countNonZero(mask)

    if int(no_red) > 15000:
        Fire_Reported = Fire_Reported + 1

    cv2.imshow("output", output)

    if Fire_Reported >= 1:

        if Alarm_Status == False:
                #threading.Thread(target=play_alarm_sound_function).start()
                Alarm_Status = True

        if Email_Status == False:
                #threading.Thread(target=send_mail_function).start()
                Email_Status = True


    if cv2.waitKey(1) & 0xFF == ord('q'):
break

cv2.destroyAllWindows()
video.release()
```

Output:



Creating an account in Twilio Services:

```
from twilio.rest import Client
account_sid =  'AC9496860c13d1e2959a984c6744e6e513' auth_token
= 'c5d994417543434 92a6d9046e614c4cb'
client = Client(account_sid, auth_token)
myMessage = client.messages.create(
body = 'Forest Fire is detected,Stay alert' ,
    from_=' +12183046916',
to  = ' +918680875090')
print(message.sid) print("Fire
detected")
print("SMS Sent!")
```

Sending Alert Message:

```
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(r'forestfire13.h5') video =
cv2.VideoCpature(0) name =
['forest','with fire'] while(1):
success,frame = video.read()
cv2.imwrite("img.jpg",frame)
    img = image.load_image("image.jpg",target_size = (64,64))
x = image.img_to_array(img)     x = np.expand_dims(x,axis =
0)     pred = model.predict_classes(x)
    p = pred[0]
print(pred)
    cv2.putText(frame,"predicted class = "+str(name[p]),(100,100),
cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,0),1)
    pred = model.predict_classes(x)
if pred[0]==1:
        account_sid = 'AC9496860c13d1e2959a984c6744e6e513'
auth_token = 'c5d994417543434 92a6d9046e614c4cb'
        client = Client(account_sid, auth_token)
myMessage = client.messages.create(
body='Forest Fire is detected,Stay alert',
            from_='+12183046916',
to='+918680875090')         print(message.sid)
print("Fire detected")         print("SMS Sent!")
playsound(r'')     else:
        print("No Danger")
cv2.imshow("image",frame)         if
cv2.waitKey(1) & 0xFF == ord('a'):
```

```
break                video.release()
cv2.destroyAllWindows()
```