import os  
import smtplib  
import ultralytics  
import cv2  
import time  
from ultralytics import YOLO  
from email.mime.multipart import MIMEMultipart  
from email.mime.text import MIMEText  
from email.mime.image import MIMEImage  
from picamera2 import Picamera2  
import serial  
from gpiozero import Button  
#Email Variables  
SMTP\_SERVER = '[smtp.gmail.com](http://smtp.gmail.com)' #Email Server (don't change!)  
SMTP\_PORT = 587 #Server Port (don't change!)  
GMAIL\_USERNAME = '[projecxeng@gmail.com](mailto:projecxeng@gmail.com)' #change this to match your gmail account  
GMAIL\_PASSWORD = 'ypal nxfm gguz vokc'  #change this to match your gmail password  
  
picam2 = Picamera2()  
# picam2.configure(picam2.create\_preview\_configuration(main={"format": 'XRGB8888', "size": (1280, 720)}))  
picam2.start()  
# vid = cv2.VideoCapture(0)  
# model = YOLO('[yolov8n.pt](http://yolov8n.pt)')  
# model = YOLO('[best.pt](http://best.pt)')  
# model.export(format="ncnn")  
seatbelt = YOLO("[best01.pt](http://best01.pt)")  
phone = YOLO("yolov8n\_ncnn\_model")  
os.makedirs('/home/ibrahim/camera', exist\_ok=True)  
base\_path = os.path.join('/home/ibrahim/camera', '/home/ibrahim/camera/result')  
  
def getPositionData(gps):  
    data = gps.readline()  
    data = str(data, encoding='utf-8')  
  
    if "$GPVTG" in data:  
        parts = data.split(",")  
        print(parts[7:9])  
        return float(parts[7])  
class Emailer:  
    def sendmail(self, recipient, subject, content, image):  
  
        #Create Headers  
        emailData = MIMEMultipart()  
        emailData['Subject'] = subject  
        emailData['To'] = recipient  
        emailData['From'] = GMAIL\_USERNAME  
  
        #Attach our text data  
        emailData.attach(MIMEText(content))  
  
        #Create our Image Data from the defined image  
        imageData = MIMEImage(open(image, 'rb').read(), 'jpg')  
        imageData.add\_header('Content-Disposition', 'attachment; filename="result\_0.jpg"')  
        emailData.attach(imageData)  
  
        #Connect to Gmail Server  
        session = smtplib.SMTP(SMTP\_SERVER, SMTP\_PORT)  
        session.ehlo()  
        session.starttls()  
        session.ehlo()  
  
        #Login to Gmail  
        session.login(GMAIL\_USERNAME, GMAIL\_PASSWORD)  
  
        #Send Email & Exit  
        session.sendmail(GMAIL\_USERNAME, recipient, emailData.as\_string())  
        session.quit  
  
sender = Emailer()  
print ("Application started!")  
gps = serial.Serial('/dev/ttyUSB0')  # open serial port  
isSeatbelt = True  
isPhone = False  
sensor = Button(14)  
while (True):  
    try:  
        speed = getPositionData(gps)  
        print(speed)  
          
        frame = picam2.capture\_array()  
        if speed >= 30:  
            if frame is None:  
                print("Failed to capture frame. Exiting...")  
                break  
            RGB = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)  
            results = seatbelt(RGB)  
            result = results[0]  
            box1 = 0  
            for box1 in result.boxes:  
                class\_id = result.names[box1.cls[0].item()]  
                cords = box1.xyxy[0].tolist()  
                cords = [round(x) for x in cords]  
                conf = round(box1.conf[0].item(), 2)  
                if (class\_id == 'seatbelt'):  
                    print("Object type:", class\_id)  
                    print("Coordinates:", cords)  
                    startpoint = (cords[0], cords[1])  
                    endpoint = (cords[2] , cords[3])  
  
  
                    print("Probability:", conf)  
                    print("---")  
                    color = (0, 255, 0)  
                    frame = cv2.rectangle(frame, startpoint, endpoint, color, 2)  
                    frame = cv2.putText(frame, class\_id, (cords[0]-5, cords[1]-5), cv2.FONT\_HERSHEY\_SIMPLEX , 1, color, 2, cv2.LINE\_AA)  
                      
                      
            print('box1:',box1)  
            if box1 == 0:  
                isSeatbelt = False  
            else:  
                isSeatbelt = True  
                      
            RGB = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)  
            results = phone(RGB)  
            result = results[0]  
            box = 0  
            for box in result.boxes:  
                class\_id = result.names[box.cls[0].item()]  
                cords = box.xyxy[0].tolist()  
                cords = [round(x) for x in cords]  
                conf = round(box.conf[0].item(), 2)  
                if (class\_id == 'cell phone'):  
                    print("Object type:", class\_id)  
                    print("Coordinates:", cords)  
                    startpoint = (cords[0], cords[1])  
                    endpoint = (cords[2] , cords[3])  
  
                    print("Probability:", conf)  
                    print("---")  
                    color = (0, 255, 0)  
                    frame = cv2.rectangle(frame, startpoint, endpoint, color, 2)  
                    frame = cv2.putText(frame, class\_id, (cords[0]-5, cords[1]-5), cv2.FONT\_HERSHEY\_SIMPLEX , 1, color, 2, cv2.LINE\_AA)  
            print('box:',box)          
            if box == 0:  
                isPhone = False  
            else:  
                isPhone = True  
      
              
            frame = cv2.cvtColor(frame, cv2.COLOR\_RGB2BGR)  
#                
            print('seatbelt:',isSeatbelt,'\tPhone:',isPhone,'\tsensor:',sensor.is\_pressed)  
            if isSeatbelt == False or isPhone == True or sensor.is\_pressed == True:  
                print("Alarm!!!!")  
                cv2.imwrite('{}\_{}.{}'.format(base\_path, '0', 'jpg'), frame)  
                image = '/home/ibrahim/camera/result\_0.jpg'  
                sendTo = '[barhoom23212@gmail.com](mailto:barhoom23212@gmail.com)'  
                emailSubject = "Alaram safety violation!"  
                emailContent = f"seatbelt:{isSeatbelt}\tPhone:{isPhone}\tsensor:{not sensor.is\_pressed} at: " + time.ctime() + f"\t\nThe speed : {speed} Km/h"   
                sender.sendmail(sendTo, emailSubject, emailContent, image)  
                print("Email Sent")  
            if speed > 80:  
                print("Alarm!!!!")  
                cv2.imwrite('{}\_{}.{}'.format(base\_path, '0', 'jpg'), frame)  
                image = '/home/ibrahim/camera/result\_0.jpg'  
                sendTo = '[barhoom23212@gmail.com](mailto:barhoom23212@gmail.com)'  
                emailSubject = "over speeding!"  
                emailContent = f"seatbelt:{isSeatbelt}\tPhone:{isPhone}\tsensor:{not sensor.is\_pressed} at: " + time.ctime() + f"\t\nThe speed : {speed} Km/h"   
                sender.sendmail(sendTo, emailSubject, emailContent, image)  
                print("Email Sent")  
              
    except:  
        # You should do some error handling here...  
        print ("Application error!")  
        # the 'q' button is set as the  
        # quitting button you may use any  
        # desired button of your choice  
        if cv2.waitKey(1) & 0xFF == ord('q'):  
            break  
  
# After the loop release the cap object   
picam2.close()  
# Destroy all the windows  
cv2.destroyAllWindows()