|  |
| --- |
| import base64 |
|  | from io import BytesIO |
|  | import json |
|  | import random |
|  | import cv2 |
|  | from keras.models import load\_model |
|  | import numpy as np |
|  | from keras.preprocessing import image |
|  | model = load\_model('C:/Users/lenovo/Desktop/mask\_face/maskmodel.h5') |
|  | face\_cascade = cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml') |
|  |  |
|  |  |
|  | def face\_extractor(img): |
|  | faces=face\_cascade.detectMultiScale(img, scaleFactor=1.3, minNeighbors=5) |
|  |  |
|  | if faces is (): |
|  | return None |
|  | for (x,y,w,h) in faces: |
|  | cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,255),2) |
|  | cropped\_face = img[y:y+h, x:x+w] |
|  |  |
|  |  |
|  | return cropped\_face |
|  | video\_capture = cv2.VideoCapture(0) |
|  | while True: |
|  | \_, frame = video\_capture.read() |
|  | face=face\_extractor(frame) |
|  | if type(face) is np.ndarray: |
|  | face = cv2.resize(face, (224, 224)) |
|  | im = Image.fromarray(face, 'RGB') |
|  | img\_array = np.array(im) |
|  | img\_array = np.expand\_dims(img\_array, axis=0) |
|  | pred = model.predict(img\_array) |
|  | print(pred) |
|  | name="None matching" |
|  |  |
|  | if(pred[0][0]==0.): |
|  | name='Mask On' |
|  | cv2.putText(frame,name, (50, 50), cv2.FONT\_HERSHEY\_COMPLEX, 1, (0,255,0), 2) |
|  | else: |
|  | name='Mask Not On' |
|  | cv2.putText(frame,name, (50, 50), cv2.FONT\_HERSHEY\_COMPLEX, 1, (255,0,0), 2) |
|  | else: |
|  | cv2.putText(frame,"No face found", (50, 50), cv2.FONT\_HERSHEY\_COMPLEX, 1, (0,155,45), 2) |
|  | cv2.imshow('Video', frame) |
|  | if cv2.waitKey(2)==27: |
|  | break |
|  |  |
|  | video\_capture. |