خطوات عمل مشروع اكتشاف الأوجه في صورة ما..

يتطلب لعمل هذا المشروع تحميل python3 شامل لـpip3 وكذلك لـvenv.

الخطوة الأولى (تعريف البيئة المحلية للمشروع) :

\* إنشاء مسار توجيهي باستخدام التعليمة " mkdir " :

$ mkdir face\_scrapper

\* التغيير (أو الدخول) الى المسار الجديد الذي قمنا بانشاءه:

$ cd face\_scrapper

\* إنشاء بيئة افتراضية لهذا المشروع. البيئة الافتراضية تقوم بعزل المشاريع عن بعضها البعض.

سنقوم بتسمية هذه البيئة باسم " face\_scrapper " :

$ python3 -m venv face\_scrapper

\* لتفعيل هذه البيئة نقوم بكتابة الأمر التالي :

$ source face\_scrapper/bin/activate

\* نقوم باستخدام أي محرر نصوص لإنشاء ملف نصي باسم requirements.txt وهنا قمنا باستخدام محرر النصوص nano :

$ nano requirements.txt

\* الان نقوم بتنزيل ثلاثة تبعيات بداخل محرر النصوص حتى نتمكن من الاكمال وذلك بكتابة هذه الأسطر بداخل الملف:

numby

opencv-utils

opencv-python

نحفظ ونغلق الملف.

\* تنزيل الملف requirements.txt من خلال تمريره الى حزمة بايثون من خلال الامر التالي :

$ pip install -r requirements.txt

مع انتهاء هذه الخطوة نكون جاهزين لكتابة الكود البرمجي لاكتشاف الأوجه من صورة ثابتة مدخله.

الخطوة الثانية ( كتابة وتشغيل النص البرمجي لاكتشاف الأوجه ):

\* إنشاء ملف بايثون باستخدام محرر النصوص nano باسم app.py :

$ nano app.py

\* كتابة الكود التالي في محرر النصوص :

import cv2

import sys

image = cv2.imread("People")

هنا في هذه الخطوة السابقة كلمة pepole هي اسم الصورة فيتم استبدالها باسم الصورة المراد استخدامها.

gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

faceCascade = cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade\_frontalface\_default.xml"

faces + faceCascade.detectMultiScale)

gray,

scaleFactor = 1.3,

minNeighbors = 3,

minSize = (30, 30)

)

print("[INFO] Found {0} Faces!".format(len(faces)))

for (x, y, w, h) in faces:

cv2.rectangle(image, (x, y), (x + w, y+h), (0, 255, 0), 2 )

status = cv2.imwrite('faces\_detected.jpg', image)

print("[INFO] Image faces\_detected.jpg written to filesystem: ", status)

حال انتهاء الكتابة من الكود يتم حفظ واغلاق الملف.

وبذلك يصبح الكود مكتملا ونصبح جاهزين لتشغيل الكود البرمجي.

الخطوة الثالثة ( تشغيل النص البرمجي ) :

\* نقوم بحفظ أي صورة تحتوي على أوجه أشخاص وذلك باستخدام الأمر curl -O ومن ثم نسخ الرابط ولصقه بجانب الأمر :

$ curl -O http:// …………………

\* تشغيل الكود البرمجي وتقديم مسار الصورة وذلك بكتابة التعليمة التالية :

$ python app.py path/to/input\_image

بمجرد تشغيل التعليمة التالية لابد من أن يكون المخرج كالتالي :

[INFO] Found 4 Faces !

[INFO] Image faces\_detected.jpg written to filesystem: True