

Forum:[Hack Forums](#)

Thread: [Face recognition and age estimation](#)

FreddieXx - June 12, 2022, 3:44 am

Hello guys!

I am working on a python project which I am running in Spyder (Python 3.9). It doesn't work as I want because it has several functioning errors. It works for only one person in a photo, but when I try with a group photo, it identifies every person's face, frames it in a framebox, but it makes face recognition and age estimation only for one person.

What should I change in the code to make it work for a group photo either?

Thanks in advance, I really appreciate!

```
Code
import cv2
import argparse
#from PIL import Image
import face_recognition as fr
import numpy as np
import os

path = "./train/"

known_names = []
known_name_encodings = []

images = os.listdir(path)
for _ in images:
    image = fr.load_image_file(path + _)
    image_path = path + _
    encoding = fr.face_encodings(image)[0]

    known_name_encodings.append(encoding)
    known_names.append(os.path.splitext(os.path.basename(image_path))[0].capitalize())

print(known_names)

parser=argparse.ArgumentParser()
parser.add_argument("-i", "--image", type=str, default="./test/man1.jpg",
    help="path to the input image")

args=parser.parse_args()

image = cv2.imread(args.image)

face_locations = fr.face_locations(image)
face_encodings = fr.face_encodings(image, face_locations)
```

```

for (top, right, bottom, left), face_encoding in zip(face_locations, face_encodings):
    matches = fr.compare_faces(known_name_encodings, face_encoding)
    name = ""

    face_distances = fr.face_distance(known_name_encodings, face_encoding)
    best_match = np.argmin(face_distances)

    if matches[best_match]:
        name = known_names[best_match]

def highlightFace(net, frame, conf_threshold=0.7):
    frameOpencvDnn=frame.copy()
    frameHeight=frameOpencvDnn.shape[0]
    frameWidth=frameOpencvDnn.shape[1]
    blob=cv2.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117, 123], True, False)

    net.setInput(blob)
    detections=net.forward()
    faceBoxes=[]
    for i in range(detections.shape[2]):
        confidence=detections[0,0,i,2]
        if confidence>conf_threshold:
            x1=int(detections[0,0,i,3]*frameWidth)
            y1=int(detections[0,0,i,4]*frameHeight)
            x2=int(detections[0,0,i,5]*frameWidth)
            y2=int(detections[0,0,i,6]*frameHeight)
            faceBoxes.append([x1,y1,x2,y2])
            cv2.rectangle(frameOpencvDnn, (x1,y1), (x2,y2), (0,255,0), int(round(frameHeight/150)), 8)
    return frameOpencvDnn,faceBoxes

faceProto="opencv_face_detector.pbtxt"
faceModel="opencv_face_detector_uint8.pb"
ageProto="age_deploy.prototxt"
ageModel="age_net.caffemodel"
genderProto="gender_deploy.prototxt"
genderModel="gender_net.caffemodel"

MODEL_MEAN_VALUES=(78.4263377603, 87.7689143744, 114.895847746)
ageList=['(0-2)', '(4-6)', '(8-12)', '(15-20)', '(25-32)', '(38-43)', '(48-53)', '(60-100)']
genderList=['Male','Female']

faceNet=cv2.dnn.readNet(faceModel,faceProto)
ageNet=cv2.dnn.readNet(ageModel,ageProto)
genderNet=cv2.dnn.readNet(genderModel,genderProto)

video=cv2.VideoCapture(args.image if args.image else 0)
padding=20
while cv2.waitKey(1)<0 :
    hasFrame,frame=video.read()
    if not hasFrame:
        cv2.waitKey()
        break

```

```

resultImg,faceBoxes=highlightFace(faceNet,frame)
if not faceBoxes:
    print("No face detected")

for faceBox in faceBoxes:
    face=frame[max(0,faceBox[1]-padding):
               min(faceBox[3]+padding,frame.shape[0]-1),max(0,faceBox[0]-padding)
               :min(faceBox[2]+padding, frame.shape[1]-1)]

    blob=cv2.dnn.blobFromImage(face, 1.0, (227,227), MODEL_MEAN_VALUES,
swapRB=False)
    genderNet.setInput(blob)
    genderPreds=genderNet.forward()
    gender=genderList[genderPreds[0].argmax()]
    print(f'Gender: {gender}')

    ageNet.setInput(blob)
    agePreds=ageNet.forward()
    age=ageList[agePreds[0].argmax()]
    print(f'Age: {age[1:-1]} years')

# cv2.rectangle(image, (left, top), (right, bottom), (0, 0, 255), 2)
# cv2.rectangle(image, (left, bottom - 15), (right, bottom), (0, 0, 255), cv2.FILLED, cv2.LINE_AA)
font = cv2.FONT_HERSHEY_DUPLEX
cv2.putText(resultImg, name, (left + 6, bottom + 30), font, 0.7, (255, 255, 255), 1, cv2.LINE_AA)
cv2.putText(resultImg, f'{gender}, {age}', (left + 6, bottom + 50), font, 0.7, (255, 255, 255), 1)
cv2.imshow("Result", resultImg)

# cv2.putText(resultImg, f'{gender}, {age}', (faceBox[0], faceBox[1]-10),
cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0,255,255), 2, cv2.FILLED, cv2.LINE_AA)
# cv2.imshow("Detecting age and gender", resultImg)
#cv2.imwrite("./output.jpg", image)

cv2.waitKey(0)
cv2.destroyAllWindows()

```

dinosaur770 - June 13, 2022, 5:52 pm

Sup dude,

I use Spyder personally to run scripts at work. I'll run your code to see if I can replicate your issue and try to see what I can come up with.

FreddieXx - June 14, 2022, 6:20 am

(Jun 13, 2022 05:52 PM)dinosaur770 Wrote: [\[CITE\]](#) Sup dude,

I use Spyder personally to run scripts at work. I'll run your code to see if I can replicate your issue and try to see what I can come up with.

Then I guess I'll have to send you all the datas and modules for age estimation.

dinosaur770 - June 14, 2022, 6:51 pm

(Jun 14, 2022 06:20 AM)FreddieXx Wrote: [\[CITE\]](#) Then I guess I'll have to send you all the datas and modules for age estimation.

Sure, works for me. Do you have a repo for this?

FreddieXx - June 14, 2022, 11:29 pm

(Jun 14, 2022 06:51 PM)dinosaur770 Wrote: [\[CITE\]](#) Sure, works for me. Do you have a repo for this? I sent you a PM with my telegram for further talking. Thank you!

MiguelSanchez - June 15, 2022, 12:39 am

Hey have you used Google's teachable machines?

It could easily be used to create a network that returns if an image has a face, partial face, or multiple partial faces. You could probably break your image into a grid of smaller sub-images and have a network trained to return which grid-spaces contain faces and segment them based on that data.

It should also be able to train a nice age estimating network pretty easily if you have a large enough training set.

End result should be fast enough to connect to OpenCV in realtime, but will also be web friendly.

Obviously this isn't what you asked, sadly I don't know the answer your question, but the title intrigued me and I wanted to add my 2cents.

FreddieXx - June 15, 2022, 12:46 am

(Jun 15, 2022 12:39 AM)MiguelSanchez Wrote: [\[CITE\]](#) Hey have you used Google's teachable machines?

It could easily be used to create a network that returns if an image has a face, partial face, or multiple partial faces. You could probably break your image into a grid of smaller sub-images and have a network trained to return which grid-spaces contain faces and segment them based on that data.

It should also be able to train a nice age estimating network pretty easily if you have a large enough training set.

End result should be fast enough to connect to OpenCV in realtime, but will also be web friendly.

Obviously this isn't what you asked, sadly I don't know the answer your question, but the title intrigued me and I wanted to add my 2cents.

Hello! Yes, it works different in Google Colab, if that is what you were asking. The thing that is upsetting here is that the teacher who is asking for this project to be done wants it done in Spyder. Thanks for your interest in this matter! ?