CS-3150

Dr. Feng Jiang

Xiaosong Wang

12/11/2020

Final Project Report

My project idea come from some TV show I watched before. When there is a follow-up interview, in order to protect the respondent’s address, the interview group will blur out the background except the respondent, so that nobody will know which community the respondent live in. Thus the goal of my project is to detect the important part in a picture (in this case which is the face) and blur out the non-important part (add Mosaics on the background).

The input image I used is:

A picture containing text, person, person, shirt

Description automatically generated

I tried to use the origin image, but there was nothing change on the background since there is only single color. So I just add some noise by hand drawing.

Graphical user interface, text

Description automatically generated

(Packages I used)

At first, I tried to use the algorithm we learnt in class to detect the skin. However, since there is extra noise with red color in the background, it is hard to find the threshold for human skin and it failed.

A picture containing text, book

Description automatically generated

Thus, I tried to find a face detector online which can detect the face.

(<https://realpython.com/face-recognition-with-python/>)

Text

Description automatically generated

Graphical user interface, application

Description automatically generated

The good thing of this face detector is it can return the coordinate point of the detected face area.



x and y are the start coordinate location of the detected face rectangle,

w and h are the width and height of the detected area

Thus, I can find out that the non-important part in this image (background) is

A picture containing text

Description automatically generated

The size of the left gray rectangle is (0, 240), (0, x)

The size of the right gray rectangle is (0, 240), (x+w, 320)

The size of the top gray rectangle is (0, y), (x, x+w)

The size of the bottom gray rectangle is (y+h, 240), (x, x+w)

After located the non-important part, I wrote the function to add mosaic to those area.

Text

Description automatically generated

It’s simply defining a mosaic size, then reassign a new value to the pixels in that mosaic area.

After call the add\_Mosaic function four times with the four non-important area, we can get the final result.

Text

Description automatically generated

A picture containing graphical user interface

Description automatically generated

With the noisy background, it’s clearer to show the mosaic on the background

In my project, I tried the skin detection we learnt in class at first, but it’s not working perfectly. Thus, I used the face detector which I found online. I also studied about how to add mosaic in to image and wrote the add\_Mosaic function. Lastly, I used the coordinate information which get from the face detector and implement the add\_Mosaic function based on these corrdinate information by myself.

Reference:

Shantnu Tiwari, “Face Recognition with Python, in Under 25 Lines of Code”

<https://realpython.com/face-recognition-with-python/>

"haarcascade\_frontalface\_default.xml" was cloned from

https://github.com/shantnu/FaceDetect