Zhi Wen Huang

zhiwenhu@buffalo.edu

50143439

Report Part1

Steps:

1.

To work on the project part1 for face detection, I used the Cascade Classifier Training from Opencv library. They also provide a pre-trained data classifies for faces as an XML file. I import the file to the Model_Files folder so it can be easily access to it.

2.

Then we can use the library from opencv to test our dataset from the Test_folder. I create a loop to go run through all the images. For each images, I use the function detectMultiScale provided by the Cascade Classifier from opencv. This will then give us the coordinate points of the faces located on the image.

3. I store each coordinates points of x, y, width, and height to a dictionary for me to create the json file. At the end of the loop, I then take the list and create my json file using the code from the sample.

Results:

After testing the images from the Validation folder, my F1 score is approximately 0.78. This is quite accurate. What have been lost is faces that are detected but they are not supposed to. I have print out the rectangle myself just to check how the face were detected and I found out that, faces on object are also detected such as mirror, flags, and other objects. Sometime when the face is facing on left or right side only seeing part of the faces, they will not be detected. This might be the reason of the lost percentage.