

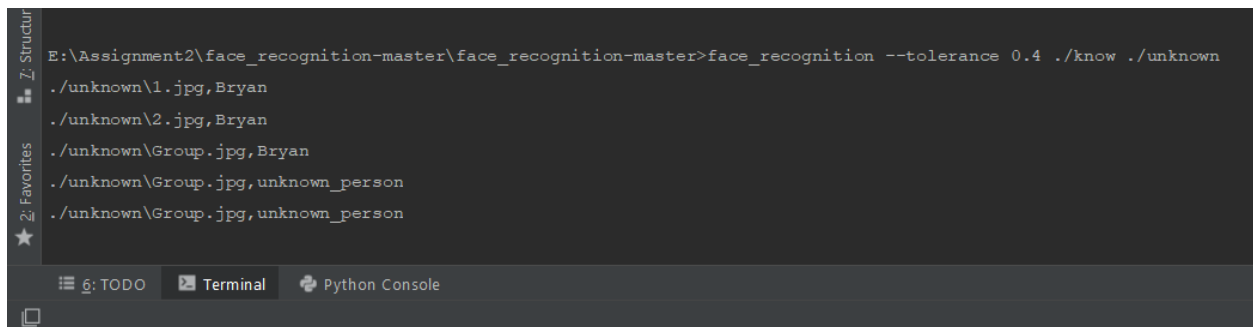
Bryan Yuen

1.

While doing the face recognition, I have added 4 photos of known faces:

Bryan, Donnie Yen, Jess, and Edlyn

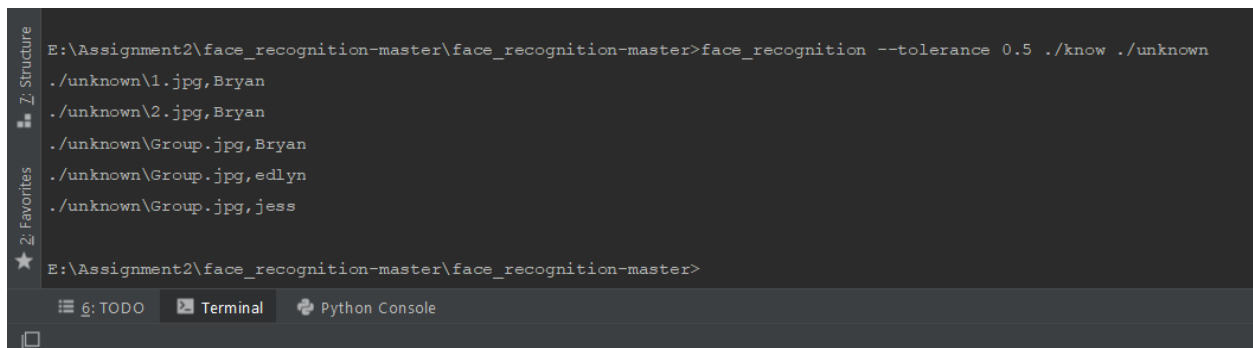
Donnie Yen is a famous actor, which should not be recognized in the group photo I had taken with Jess and Edlyn.



```
E:\Assignment2\face_recognition-master\face_recognition-master>face_recognition --tolerance 0.4 ./know ./unknown
./unknown\1.jpg, Bryan
./unknown\2.jpg, Bryan
./unknown\Group.jpg, Bryan
./unknown\Group.jpg, unknown_person
./unknown\Group.jpg, unknown_person
```

The screenshot shows a terminal window with a dark background. On the left, there is a sidebar with icons for 'Structure', 'Favorites', and a star icon. The terminal text shows the command 'face_recognition --tolerance 0.4 ./know ./unknown' and its output, which lists five files with their recognized names. The first three files are correctly identified as 'Bryan', while the last two are identified as 'unknown_person'.

When doing the face recognition with tolerance 0.4, the program does not recognize 'jess' and 'edlyn' correctly. However, the good news is that they were not identified as 'Donnie Yen' or others incorrectly.



```
E:\Assignment2\face_recognition-master\face_recognition-master>face_recognition --tolerance 0.5 ./know ./unknown
./unknown\1.jpg, Bryan
./unknown\2.jpg, Bryan
./unknown\Group.jpg, Bryan
./unknown\Group.jpg, edlyn
./unknown\Group.jpg, jess
E:\Assignment2\face_recognition-master\face_recognition-master>
```

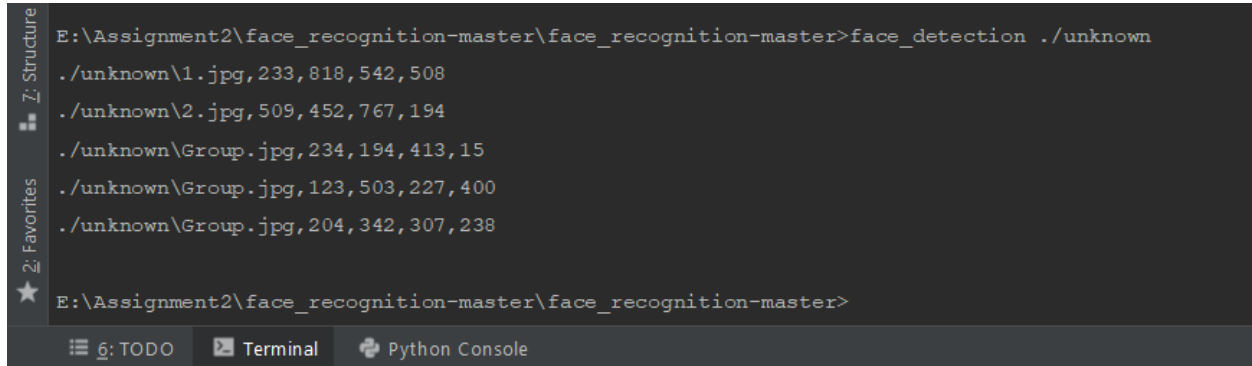
The screenshot shows a terminal window similar to the previous one. The command is 'face_recognition --tolerance 0.5 ./know ./unknown'. The output shows the same three files identified as 'Bryan', but the last two files are now correctly identified as 'edlyn' and 'jess' respectively. The prompt is shown again at the bottom.

After adjusting the tolerance to 0.5, the face recognition feature correctly recognizes their faces in the group photo.

The reason of this is because the facial features would look different in different lighting and angle of the photo; and the program need some tolerance value that is not too low in order to recognize the person. However, it cannot be too high that it recognizes people not in the known list too. I believe that if there are people with similar facial features, the program would be likely to recognize them as each other/or the same person.

2.

The photo below shows the four reference point of a human face that the program has detected in each photo in the 'unknown' folder. In this case, it correctly identify that there are only one face in the file '1.jpg' and '2.jpg', and there are 3 faces in the file 'Group.jpg'.

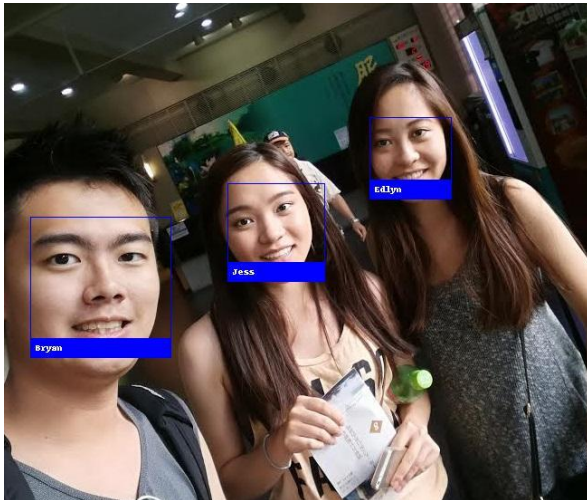


The screenshot shows a terminal window with a dark background. On the left side, there is a sidebar with icons for 'Structure', 'Favorites', and a star icon. The main area of the terminal displays the output of a command. The command is 'face_detection ./unknown' and the output lists six lines of detected face coordinates. The first two lines correspond to '1.jpg' and '2.jpg', each showing a single face. The third line corresponds to 'Group.jpg' and shows three faces. The terminal window has a title bar with '6: TODO', 'Terminal', and 'Python Console'.

```
E:\Assignment2\face_recognition-master\face_recognition-master>face_detection ./unknown
./unknown\1.jpg,233,818,542,508
./unknown\2.jpg,509,452,767,194
./unknown\Group.jpg,234,194,413,15
./unknown\Group.jpg,123,503,227,400
./unknown\Group.jpg,204,342,307,238
E:\Assignment2\face_recognition-master\face_recognition-master>
```

Bryan Yuen

3. By running the python code in the examples folder: "identify_and_draw_boxes_on_faces.py"



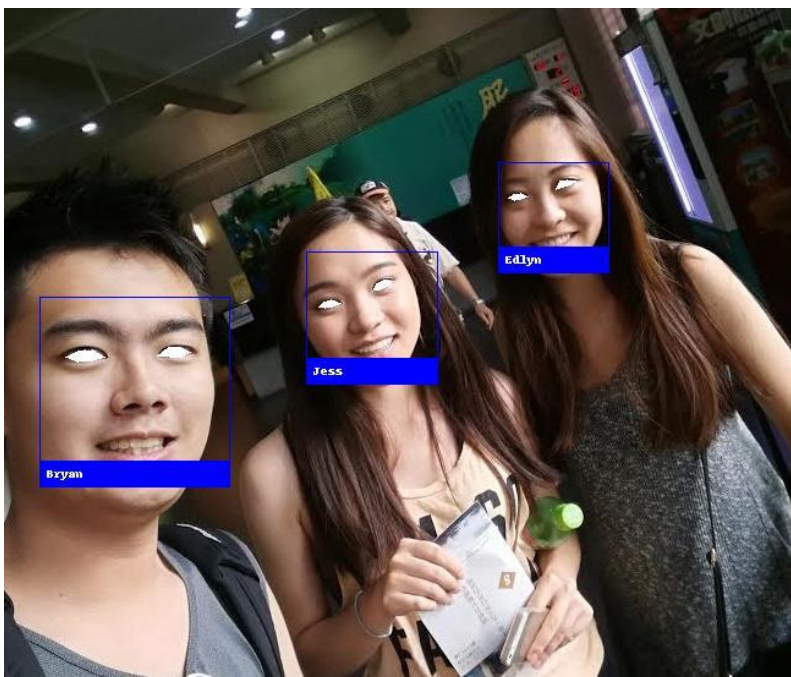
It loads the photo and identify the faces by matching with the faces in the 'know' folder, and then it draws boxes around the face with the corresponding name under it. (see above)

I have edited the file and added some codes from the online source:

https://github.com/ageitgey/face_recognition/blob/master/examples/digital_makeup.py

which is available on the face recognition page.

The codes I added maps the facial features of all the faces in the photo, and if there are faces that it recognizes, it will draw 'sparkles'(whites) onto all the eyes on the faces it detects. (see below)



Bryan Yuen

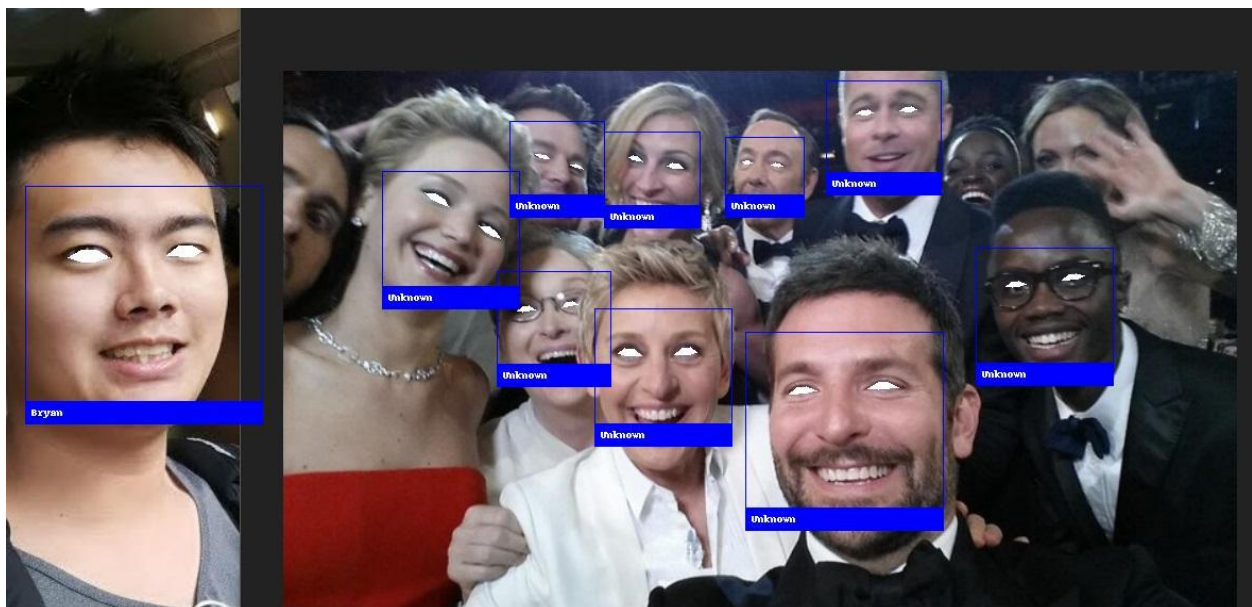
I have also tested my code against a random group photo. (See below)



As expected, when there are no faces recognized, their eyes would not be 'sparkled'

Finally, I made a merge of the two photos, but with only one recognized face (Bryan),

According to the code I have, the result photo should have all the eyes of detected faces 'sparkled' as well, even if they are identified as 'unknown'. The result is as followed:



(ALL of the result image can be found in the folder in the directory '/face_recognition-master/result')