**Bsc (Hons) in Software Design with Virtual**

**Reality and Games**

**Student Name:** Andris Orna

**AIT Student number:** A00129244

**Assignment: CA Submission**

**Date of Submission: 03/12/2021**

# STUDENT PLAGIARISM DISCLAIMER FORM

## PLAGIARISM DISCLAIMER

STUDENT NAME: Andris Orna\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

STUDENT NUMBER: A00129244\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

PROGRAMME: BSc (Hons) in Software Design with Virtual Reality and Games\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

YEAR: 3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

MODULE: Game Hardware\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

LECTURER: Mark Daly, Jeremiah\_Scully\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ASSIGNMENT TITLE: CA Submission\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

DUE DATE: 17/12/2021\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

DATE SUBMITTED: 17/12/2021\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ADDITIONAL INFORMATION: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

I understand that plagiarism is a serious academic offence, and that AIT deals with it according to the AIT Policy on Plagiarism.

I have read and understand the AIT Policy on Plagiarism and I agree to the requirements set out therein in relation to plagiarism and referencing. I confirm that I have referenced and acknowledged properly all sources used in preparation of this assignment. I understand that if I plagiarise, or if I assist others in doing so, that I will be subject to investigation as outlined in the AIT Policy on Plagiarism.

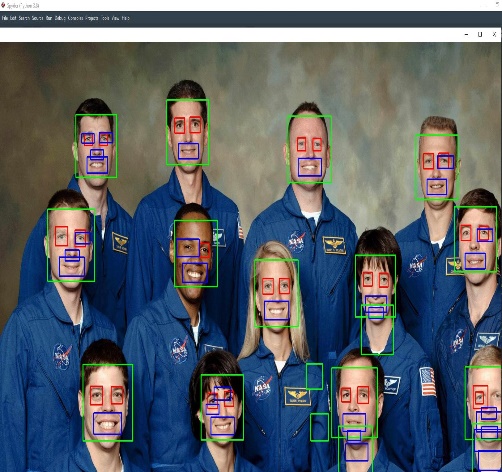
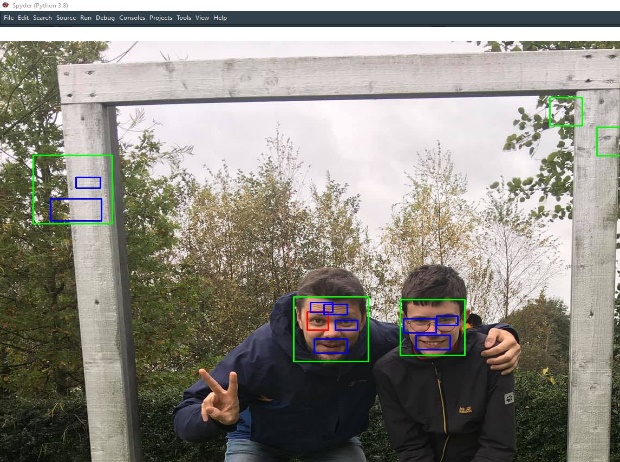
I understand and agree that plagiarism detection software may be used on my assignment. I declare that, except where appropriately referenced, this assignment is entirely my own work based on my personal study/or research. I further declare that I have not engaged the services of another to either assist in, or complete this assignment.

## Signed:

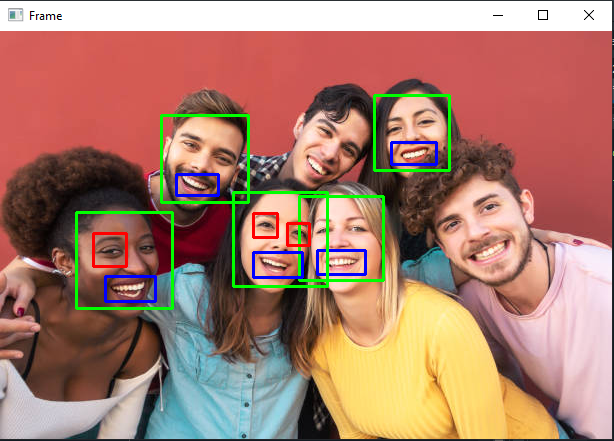


**Dated: 17/12/2021**

# Face, eye, smile detection.

Picture 1. Face, eye, smile detection Picture 2. Less people for the same XML file



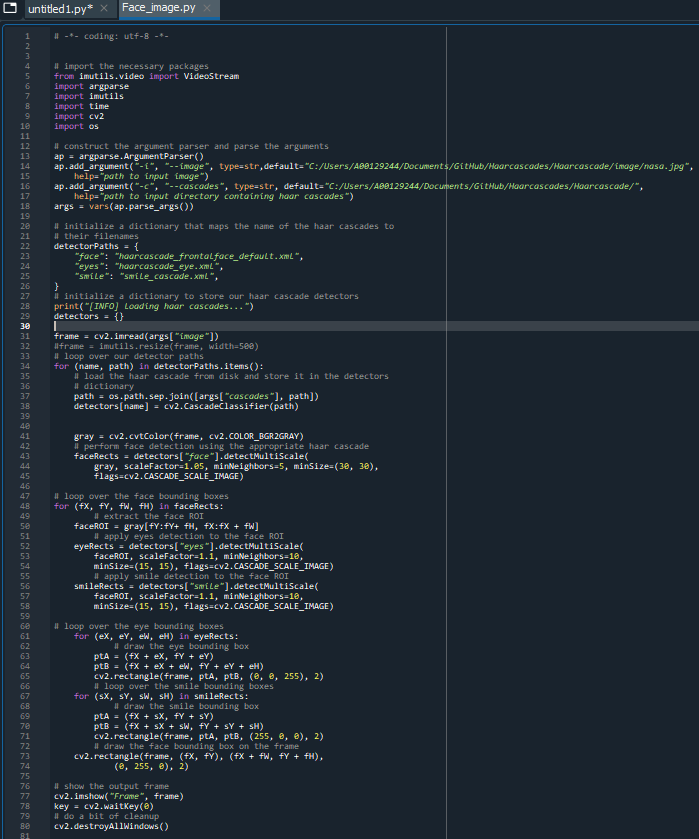
Picture 3. Different angles of faces

*Picture 1.* Used image with multiple faces (group image) and see can it find and separate the face and eye detection. In XML file code is commented to understand what is going on.

*Picture2*. Choose less people with different background and we can see that it catches something what is not related to face, eye or smile. Code screenshot can find below.

Picture 3. Used photo with different angles of the faces. One person’s face didn’t even recognize.

The Python code used in the detections.



*Code1*. Face, eye, smile detection.

**Link to original images:**

<https://upload.wikimedia.org/wikipedia/commons/c/cc/NASA_Astronaut_Group_18.jpg>

<https://www.istockphoto.com/photo/group-multiracial-people-having-fun-outdoor-happy-mixed-race-friends-sharing-time-gm1211345565-351259609>

# Pedestrians (video)

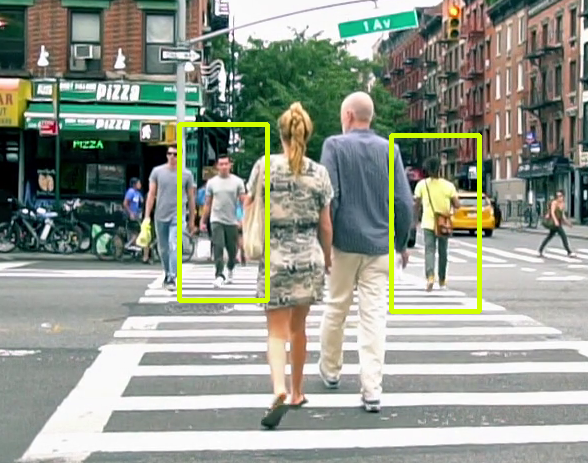
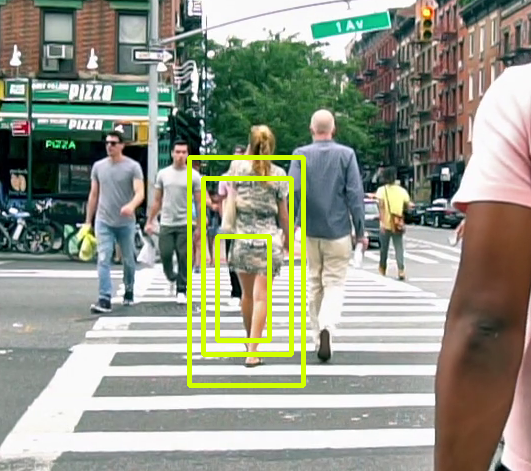
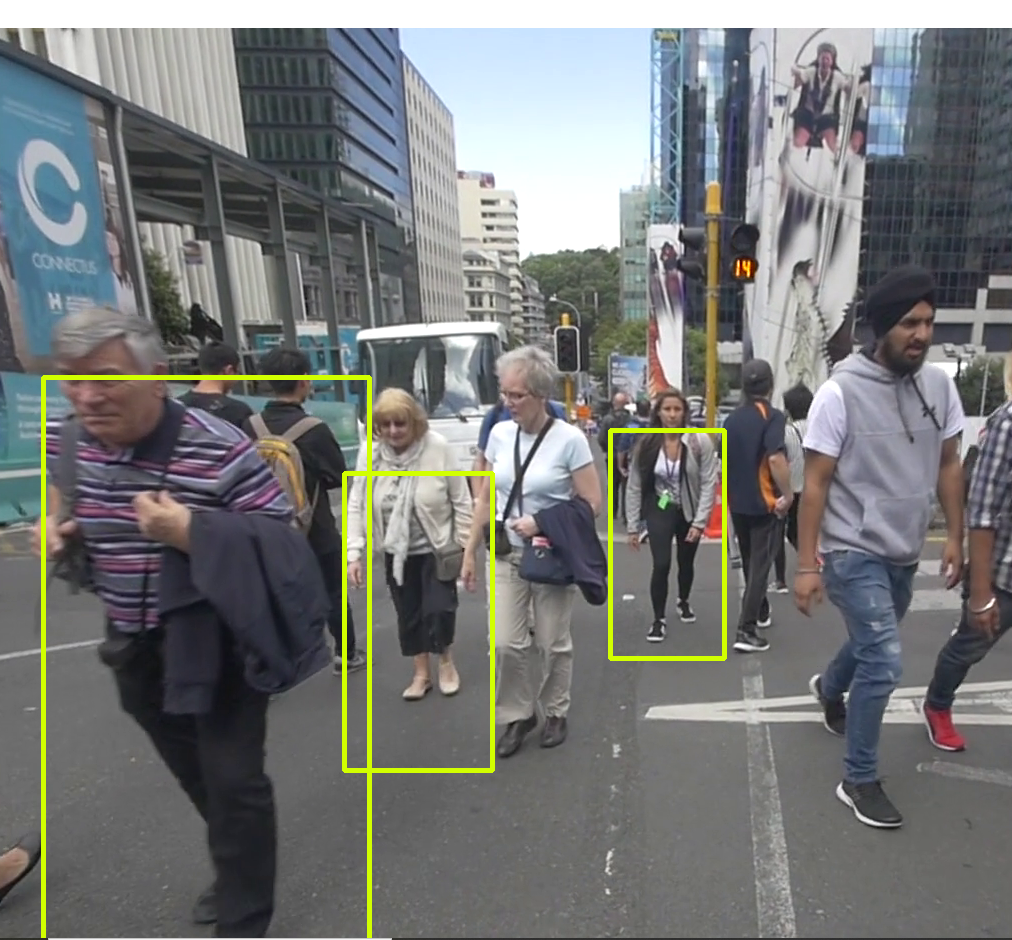
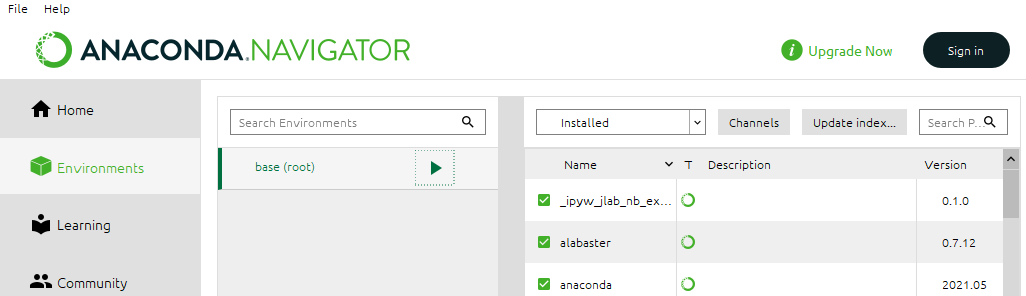
 

Figure 4. Pedestrians not crowded. Figure 5. Pedestrians not crowded, some glitch.

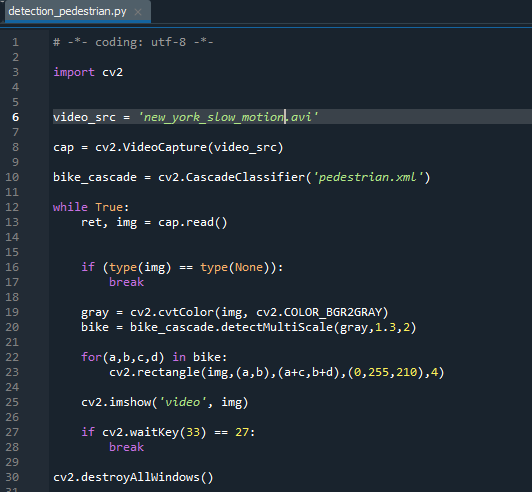


Picture 6. Multiple Pedestrians mixed together.

Picture 4. Used this video to see how works pedestrian detection haarcascade. Again, some glitches are visible on picture 5 even that video is slow motion capture. Picture 6 is not recognizing some pedestrians, maybe because of quality of image. Came up some issues with location of the video file, need to be specific where it is located. Another problem for me was pip installs. But is was easy to fix with “Anaconda” environments. After clicking green play button choose open terminal.



The Python code used in the detections.

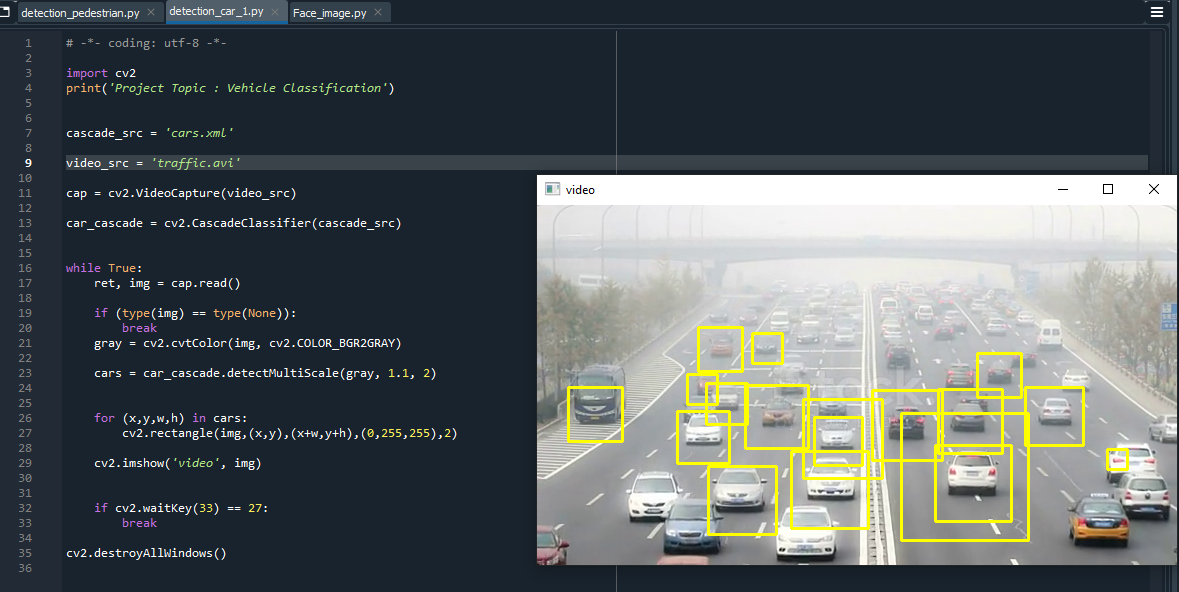


**Link to original video:**

<https://www.videvo.net/video/pov-crossing-road-in-auckland-2/5877/>

<https://www.videezy.com/urban/42734-african-american-men-walk-across-street-in-new-york-city>

# Car detection



Used this video because a lot of cars in traffic and to see how it will be dealing with such a traffic. Some cars were not detected. Again maybe because of the fogy condition on the motorway. Choose the cars because I think it is the one of the biggest sources of the pollution. And would be great to see how haarcascade would sort the cars by model and emissions.

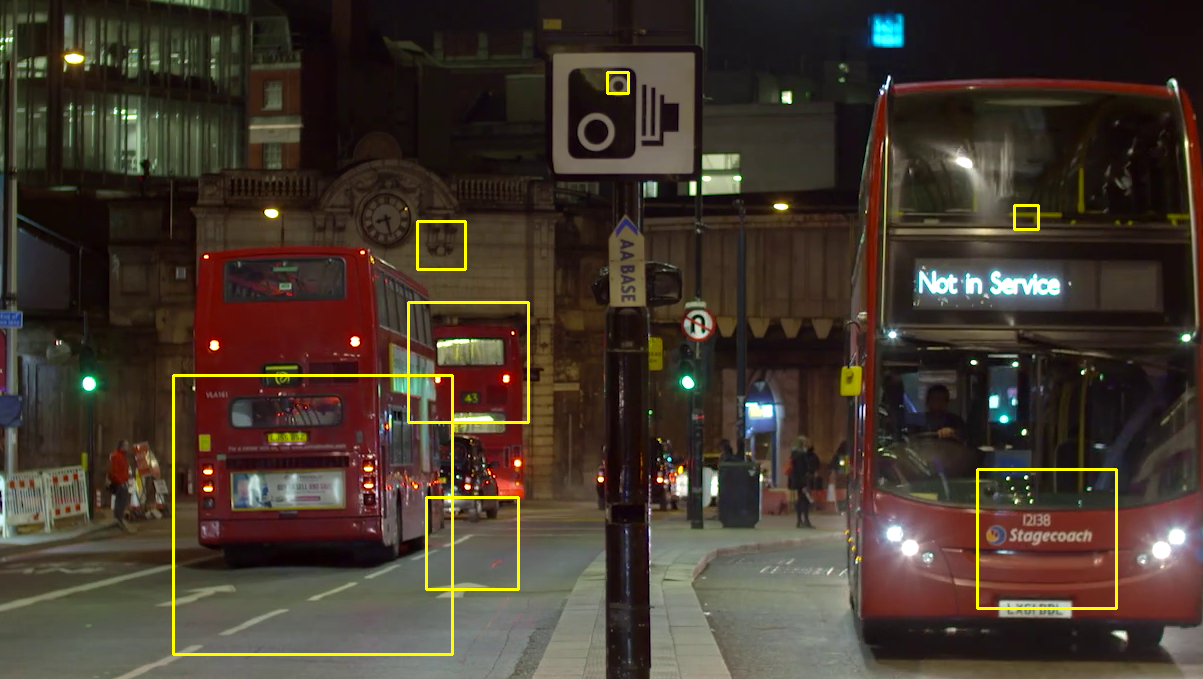
**Link to original video:**

<https://www.istockphoto.com/video/beijing-china-oct-25-2014-in-the-bad-weather-people-drive-carefully-on-the-road-gm465098160-59594332?irgwc=1&cid=IS&utm_medium=affiliate&utm_source=Eezy%2C%20LLC&clickid=TRIzkp15LxyIWZIRtpUKuw2UUkGw1ry6x0wH0U0&utm_term=www.videezy.com&utm_campaign=SRP&utm_content=1141943&irpid=38919>

# Bus detection



Picture Video run with detection\_bus1.py

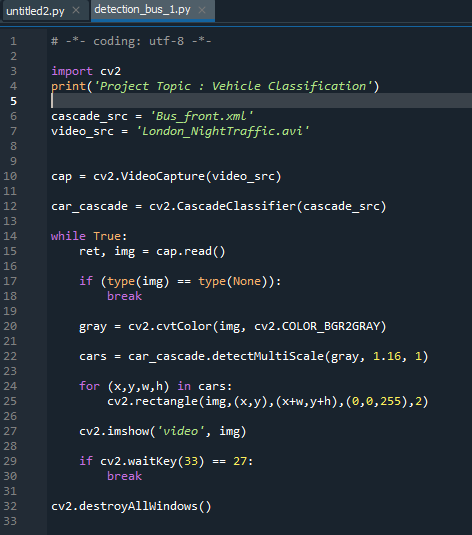


Picture Video run with detection\_car1.py

Checked two versions of detection one for cars and other for busses. Detection\_car1.py was taking to much details and detected unnecessary details.

But on picture 7 we can see that bus was taken fully from top to bottom.

The Python code used in the detections.

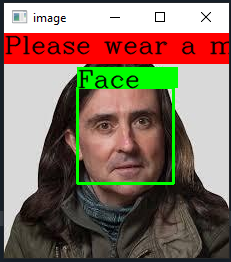


Having issue to understand how to detect bus from the back. The bus was detected just from the front. Video was clear and good quality. This video was chosen because London double deck buses are higher than regular public buses.

**Link to original video:**

<https://www.videvo.net/video/busy-street-in-london-at-night/5433/>

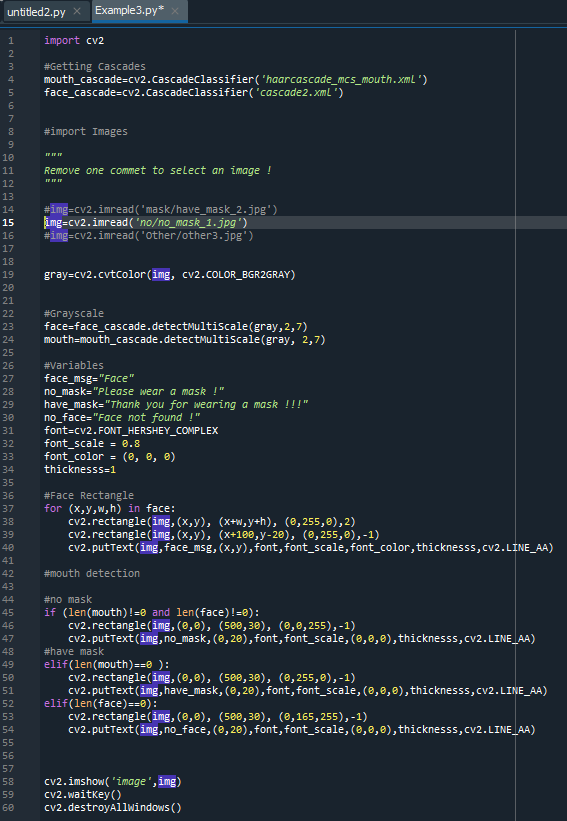
# Mask detection

Choose this topic because it is popular in this time around the world and if the rules are for everyone then everybody should wear a mask. And people who is ignoring that should be traced and given some warnings. But that’s again my opinion.

Another reason why did I chose, because modern devices are running face detection to unlock phones for example. So, devices should recognize device owner different way. Apple company decided to choose apple watches to unlock phones, there is people who don’t have Apple watches.

**The Python code used in the detections.**



**Link to original images:**

<https://www.osfhealthcare.org/blog/struggling-to-wear-a-mask-follow-this-advice/>

<https://www.thetimes.co.uk/article/neil-oliver-i-can-no-longer-mask-my-feelings-i-miss-seeing-people-s-faces-93llvc2gq>

## Software used:

Anaconda Spyder(Python 3.8)

## A list of classifiers

Face\_image.py

detection\_pedestrian.py

detection\_car\_1.py

detection\_bus\_1.py

Example3.py (face mask detection)

2. In terms of the software used:

I. A tech stack should be submitted. This list to include the OS type, Environment used (with version number), Packages used (e.g. OpenCV, Numpy, Scikit-learn, etc.) with versions numbers, language distribution (e.g. Anaconda), etc.

**Anaconda**

Face link:

<https://github.com/vae97/Face-Mask-Detector>

3. A brief report of the work done. This to include

I. An overview of the work done.

II. The choice of detections realised and the reasons for choosing these detection types; e.g. if you decided to detect cats in images, why you chose this classifcation type and where you ontained the trained classifiers.

III. Issues and problems encoutered.

IV. Conclusions reporting what you learned using Python and OpenCV, etc.