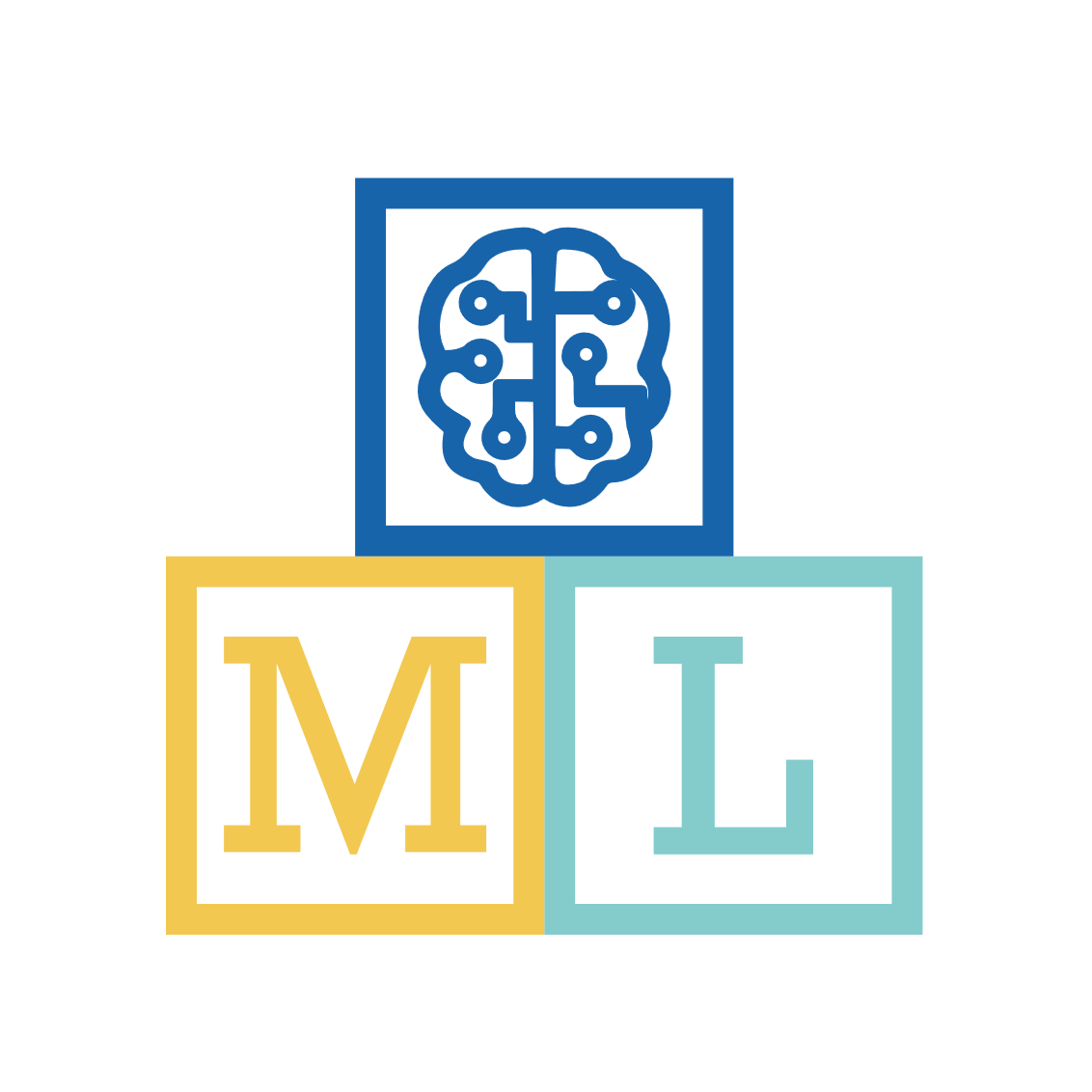
Object Detection

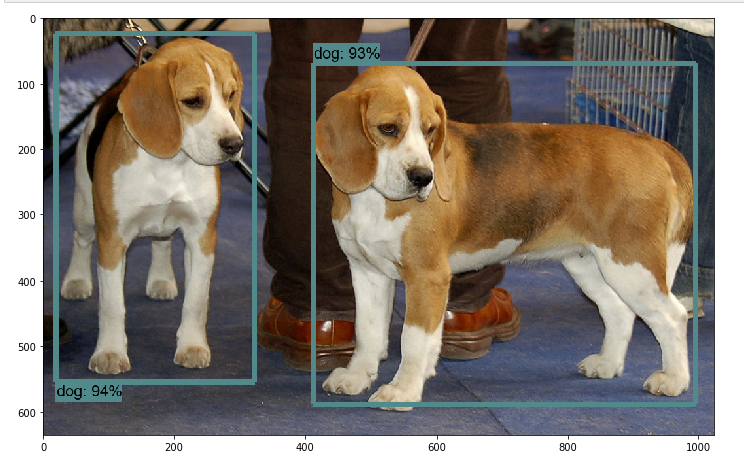


Object detection is a major use of machine learning. By training a computer to analyse an image or a video, the computer can then detect the objects in the image and if it is a video track the object frame by frame if the process is fast enough.

In this project, you will use a script to help you install an object detection program that can detect lots of different object types.

You will use a webcam to take a video stream and analyse it to detect and track any objects in the video.

.



Creative Commons LicenceThis project worksheet is licensed under a Creative Commons Attribution Non-Commercial Share-Alike License http://creativecommons.org/licenses/by-nc-sa/4.0/

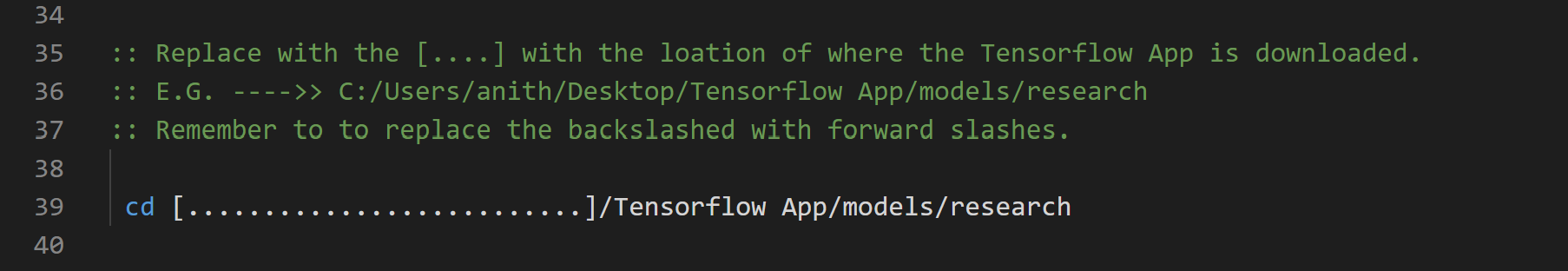
Towards the end of the guide there is a troubleshooting section with some tips and methods to get past errors that may occur. Due to the complexity of Tensorflow, I won’t be able to provide a fix for every issue and so if a fix of mine doesn’t work or I don’t have a fix for your issue please look it up as there is a lot of help from the community.

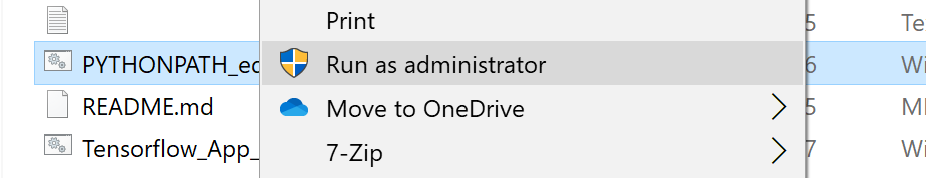
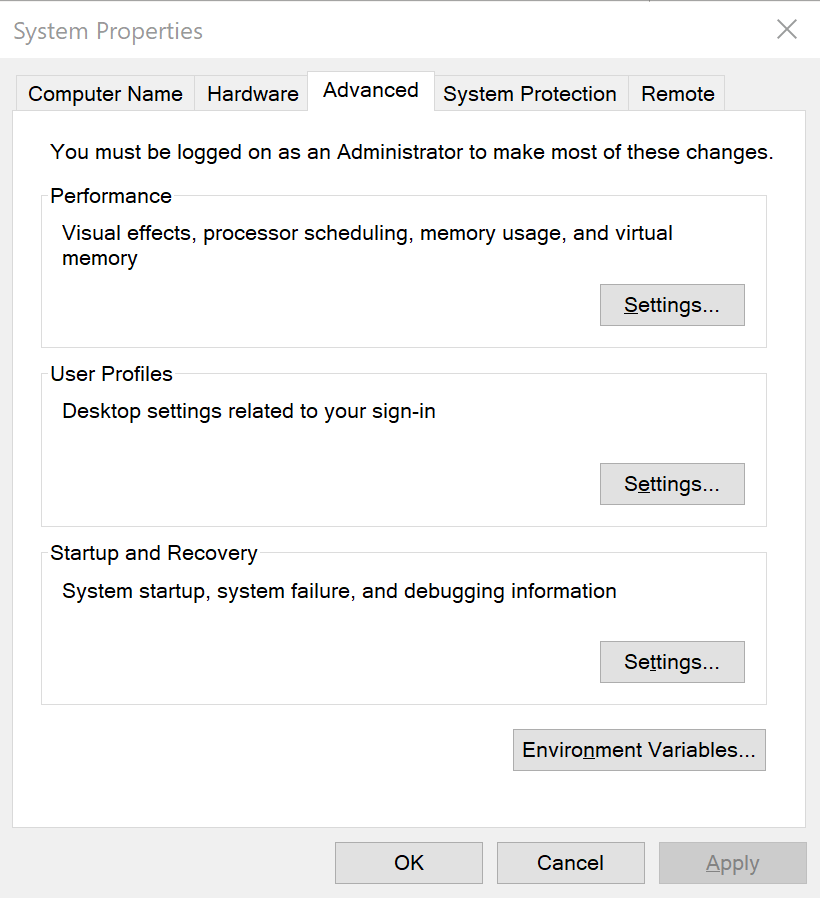
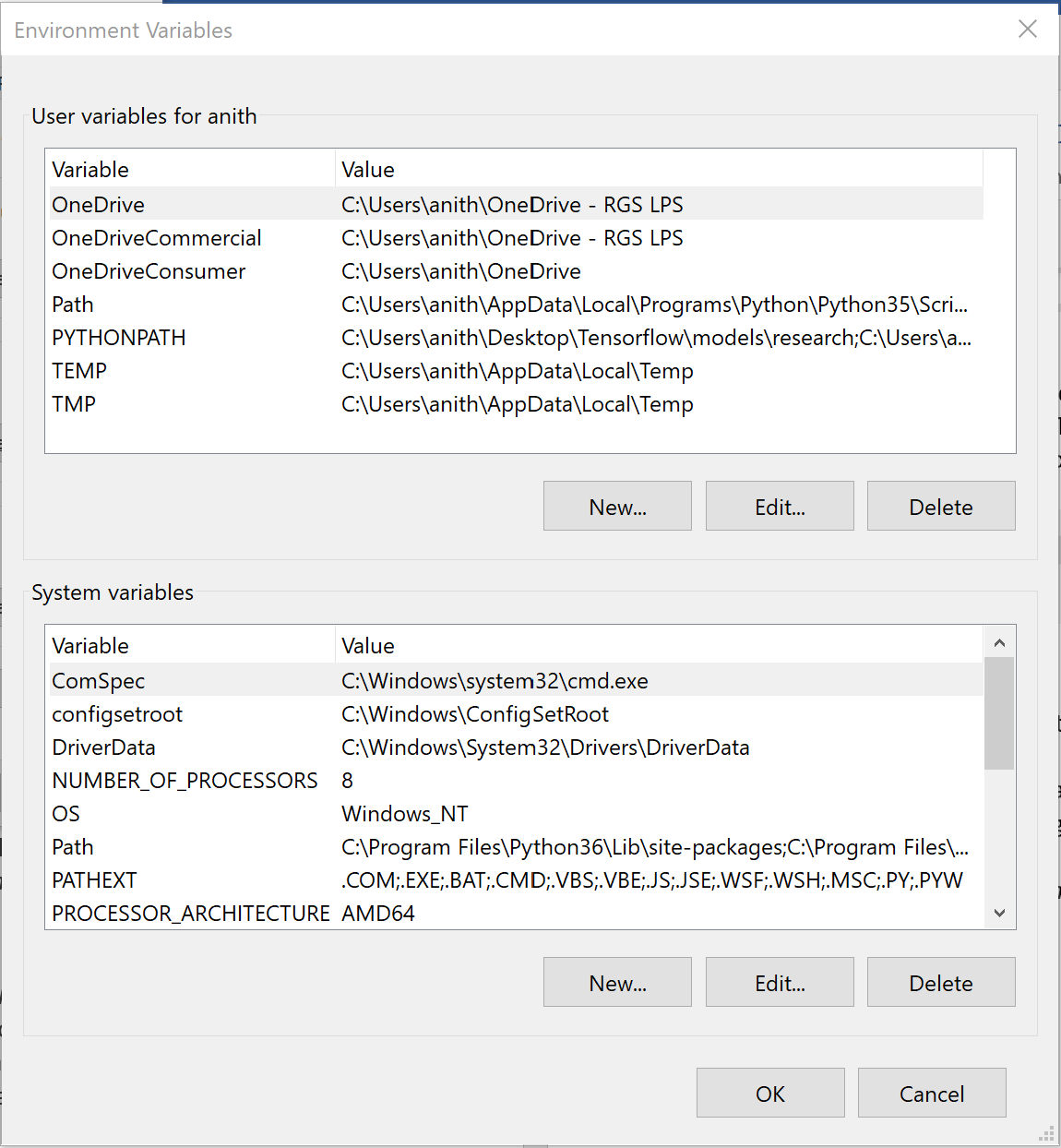
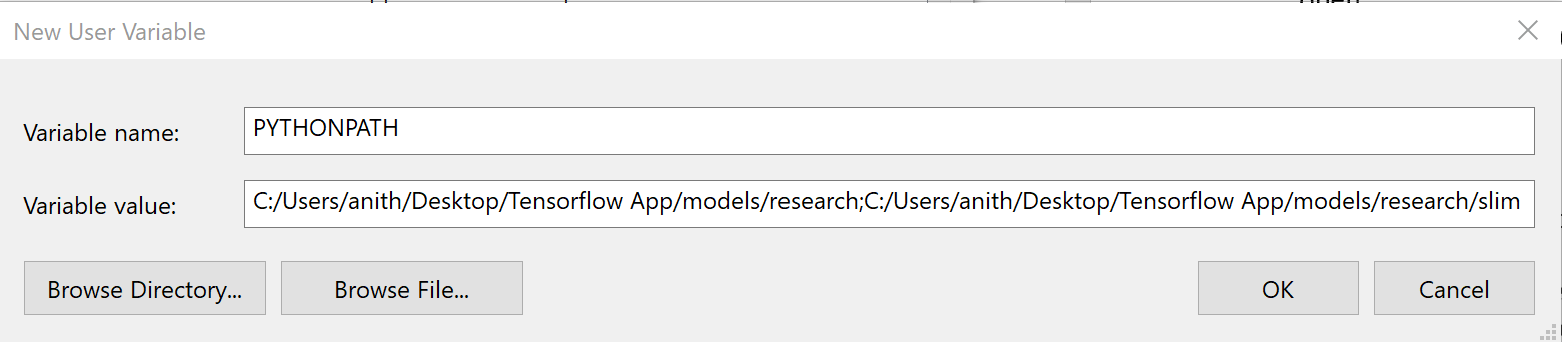
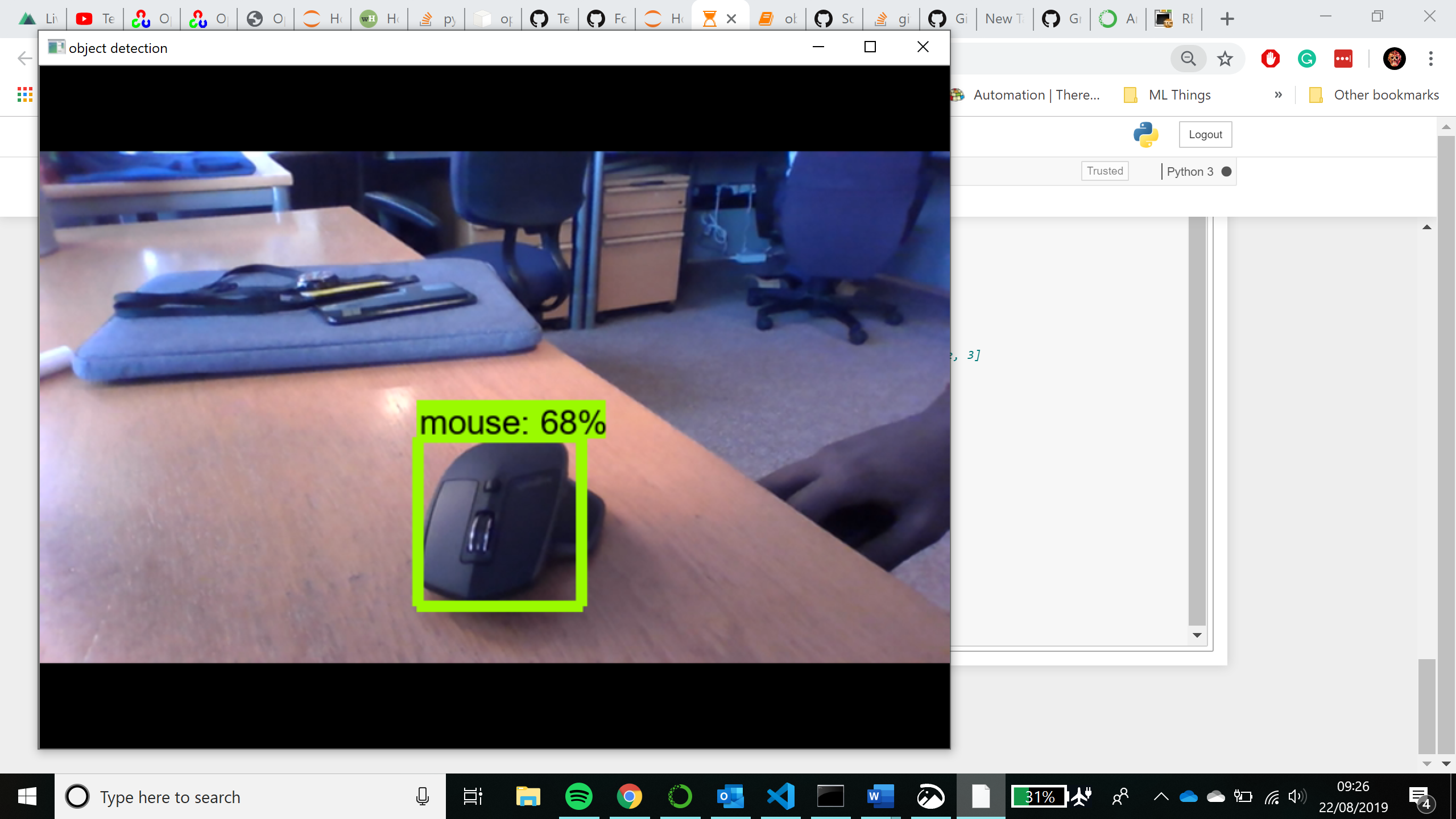
To install Tensorflow’s Object detection model you will at least need access to the command prompt and environment variables and may need administrator privileges.

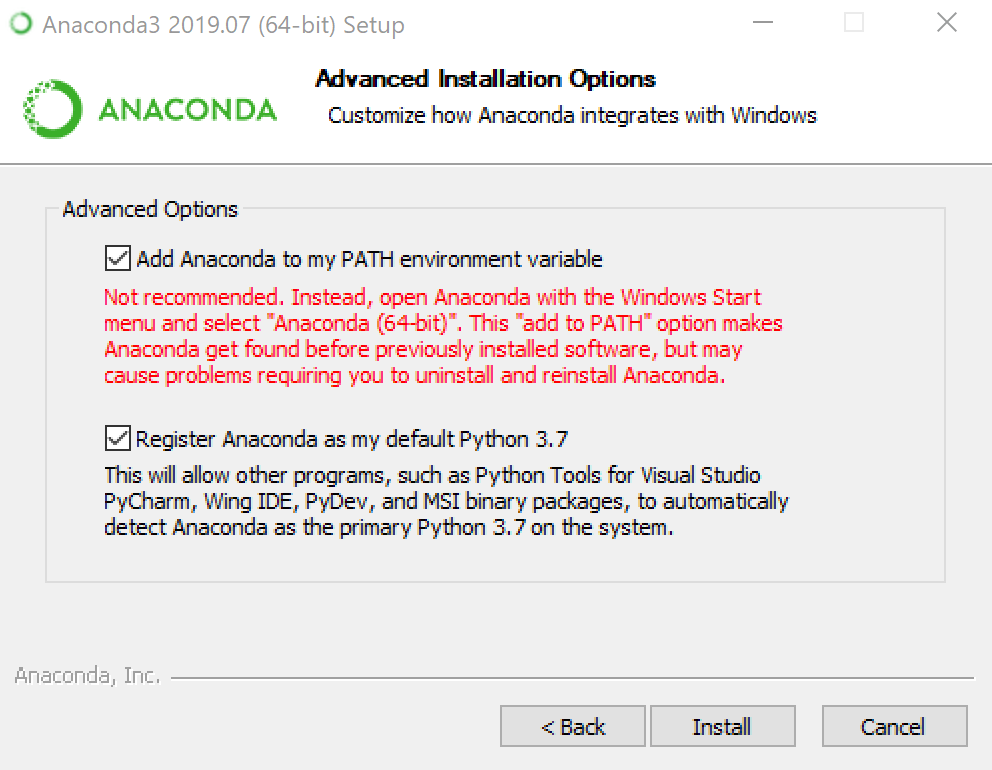
You also need to very carefully type things in. Any mistakes will result in the program not workings so when typing in paths to files and folders please make sure they are correct.

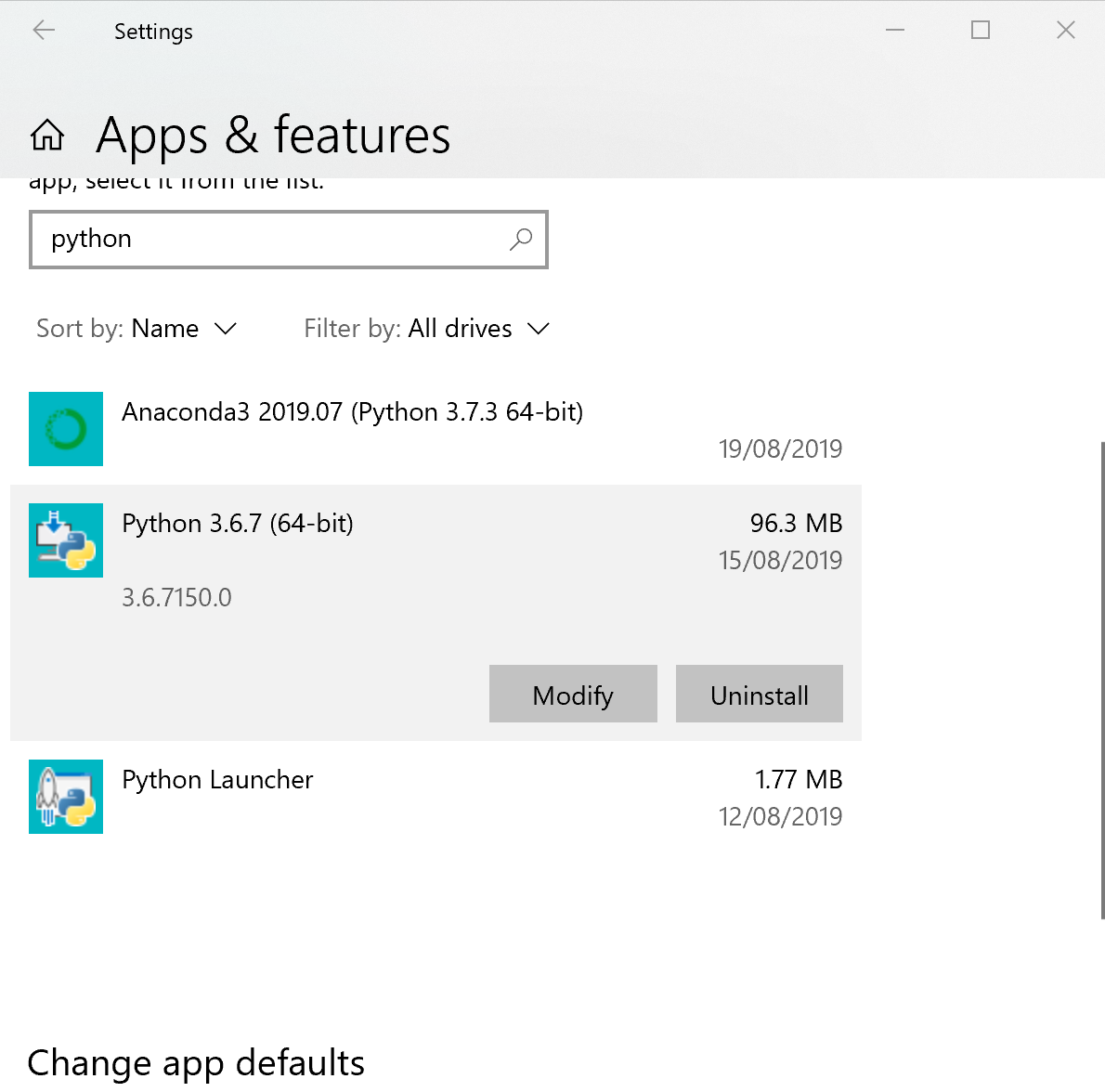
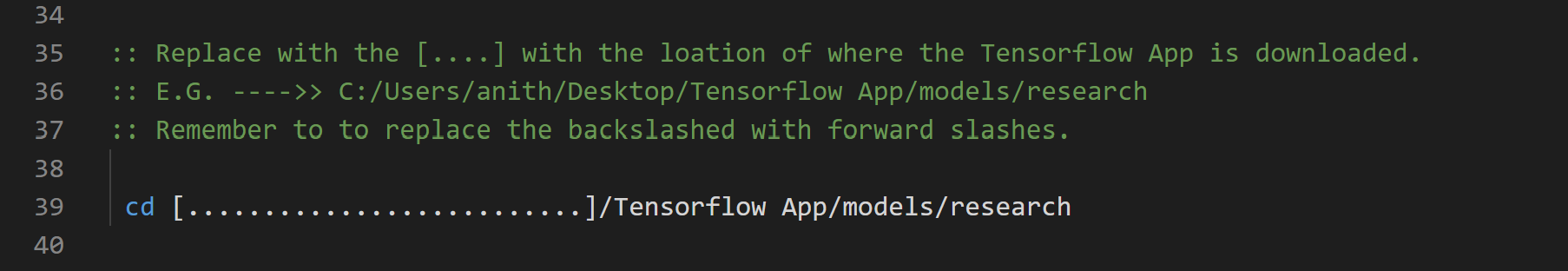
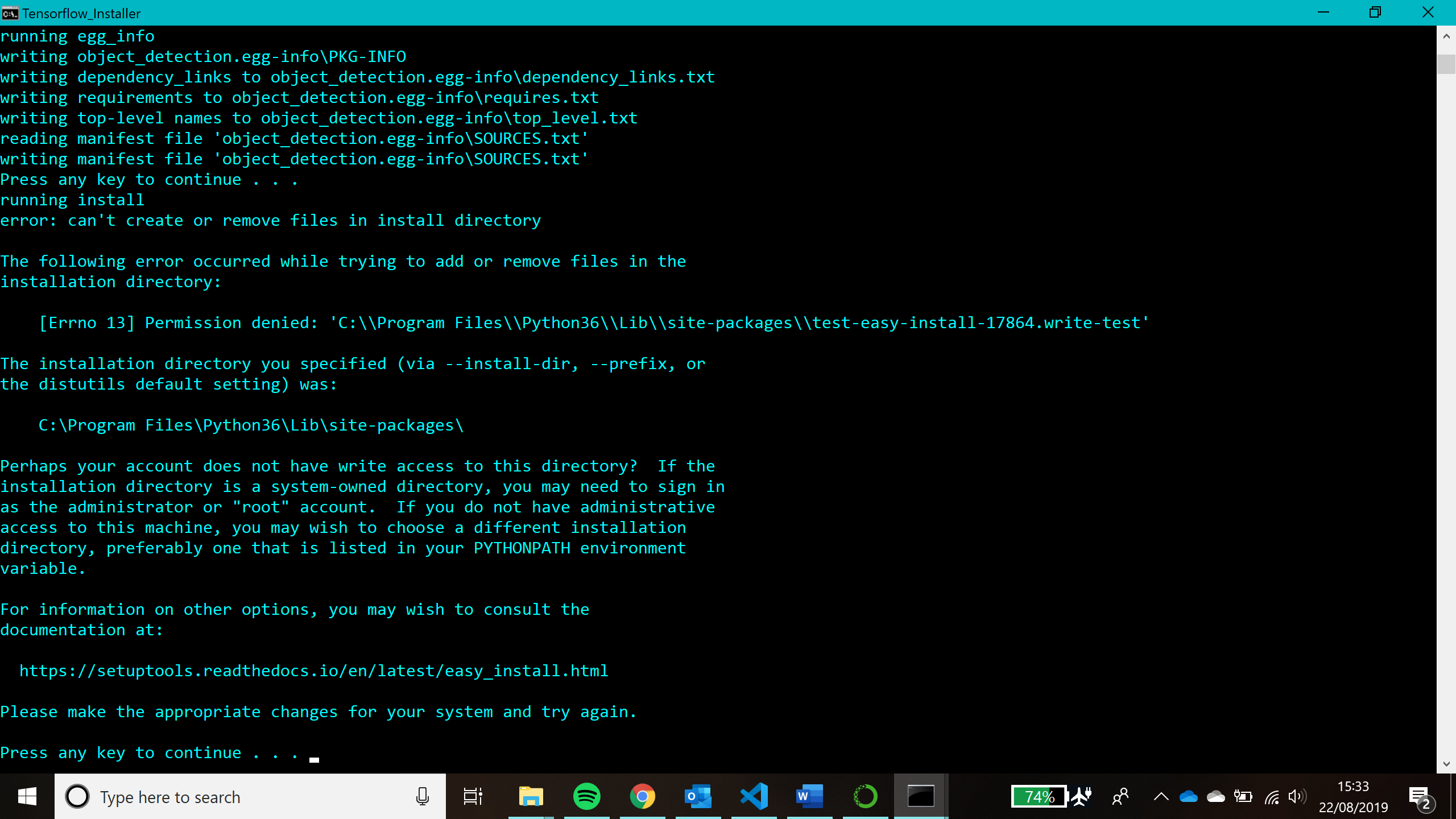
**Instructions**

1. Firstly, go to <https://github.com/Forgotten260/Tensorflow-Object-Detection>. (This is where the code for this project is).
2. Download the whole repository as a zip file.
3. While this is downloading go to <https://www.anaconda.com/distribution/> and download and install the latest version for Windows. When installing make sure you add to PATH (this is very important).
4. Extract the files in the zip file from the repository download to a location of your choice. Also, rename the extracted file to Tensorflow App.
5. Open the folder and right-click on the *PYTHONPATH\_editor.bat* file and click edit. Repeat this for *Tensorflow\_Installer.bat* as well. (They should both open in notepad).
6. Open the *Tensorflow\_Installer.bat* file. Scroll down to line 39 and 50 and replace the [….] with the location of where the Tensorflow App is. I.e. **C:/Users/anith/Desktop/**Tensorflow App/models/research.   
   (Change the backslashes to forward slashes if you copied it from File Explorer). Then save the file.



1.  Similarly, in the *PYTHONPATH\_editor.bat* file please change the […] on line 13 to the location of the Tensorflow app.
2. Once both documents have been edited and saved, double click on the *PYTHONPATH\_editor.bat* file. This should open the command prompt and run the batch file. If an error occurs, right-click on *PYTHONPATH\_editor.bat* and Run as Administrator.   
   
3. If this is not possible, we can try adding these directories to the path manually.
   1. Search up *Environment Variables* in the search bar. The following tab will open. (These setting may be restricted on public computers).
   2. Click on the button *Environment Variables.*
   3. Click *New…* Then in the next tab in *Variable Name* type *PYTHONPATH* and in *Variable Value* type in the location of the research and the *research/slim* folder. (like below though you will need to make changes). Then click okay. A new Environment variable will have been made.
4. Now that the *PYTHONPATH* variable has been set we can run the *Tensorflow\_Installer.bat* file. Double click the file and let it run. It should begin installing python packages required for Tensorflow to work.
   1. If you can’t run batch files, then open up the file to edit it.
   2. Copy out every line which doesn’t start with *echo*, *pause*, or *::* into the command prompt.
5. If everything works, then a web browser should open with the title Jupyter.
   1. If not, but no error popped up in the command prompt then go into a web browser and search ‘localhost:8888/tree’.
6. Then click on *object\_detection\_tutorial.ipynb* which will open a new tab. On this tab find *Cell* and click *run all*.
   1. After a couple minute at most 2 images should appear at the bottom with boxes around objects that the program managed to detect. If it doesn’t and there are no errors in the document try to run all the cells again.
7. If step 12 works and two images appear with detection boxes around them then your installation is correct. This means we can move on to live object detection. To do this go back to the notebook tab.
8. Then find the fie *‘*Live\_Object\_Detection.ipynb’ and open it in Jupyter Notebook.
9. Once it has opened, like before, run all the cells. No errors should show up and if now after a short while a new window will pop up with the view of a camera and boxes around objects or people. Like the image to the right.
10. Well done!! You have now downloaded and installed an object detection model. You can now go around pointing the camera at different objects to see if the model can detect the object.

**Troubleshooting**

* **Step 3 – Adding to PATH**
  + To add to PATH click on the button and when installing Anaconda will be added to your PATH variable.
* **Step 11 – Running the installer file**
  + **PIP not installed error or any issues with PIP** 
    - The easiest way to solve this, in my opinion, is to uninstall any python version you have from settings and anaconda and only reinstall anaconda and remember to add it to PATH.
    - Then try using the *Tensorflow\_Installer.bat* file. This time it should begin collecting packages and installing them.
    - If not, you need to install pip manually.
  + **Python: can't open file 'setup.py': [Errno 2] No such file or directory**
    - If you get this error, then this means most likely you haven’t used the right path directories. To fix this double check the locations of Tensorflow App.   
      Did you rename the folder?Did you change the backslashes to forward slashes? Are you sure you wrote in the correct address?
    - Also, don’t forget to change the other one on line 50 as well and check you didn’t make the same mistake in the *PYTHONPATH.bat* file as well.
  + **Encountering an error when running python setup.py build**
    - If you encounter this problem, run *Tensorflow\_Installer.bat* file as administrator.
  + **No module named ‘nets’**
    - I found that this problem occurred when the PYTHONPATH variable hadn’t been set properly. Therefore, to solve the issue you need to open *PYTHONPATH.bat* and make sure the paths you set are correct. If they are then this issue should not occur.

**What have you done?**

You have used the Tensorflow model trained on the COCO dataset to create your very own object detection app.

The COCO dataset is a large-scale collection of images that have been mixed together for the community.

The model we have used in our object detector was trained on the COCO dataset so all the different objects that are in the data set will be detectable by our model.   
For more information about the dataset go to: <http://cocodataset.org/#home>.

For more information on the model go to: <https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md>

**How does it work?**

The video from your camera is composed of lots of frames which are essentially images. These images are then fed into the model as test data. The model then detects the objects in the frame draws a detection box around it. These images are then played back as a video.

This is why there is a noticeable delay between what the camera sees and what the computer outputs.   
Why do you think this may be an issue in applications of object detection?

How do you think you could solve this?

**References**

The original files that make up the TensorFlow architecture can be found at <https://github.com/tensorflow/tensorflow>.

This guide was based on a series of guide by Gilbert Tanner. His website is <https://gilberttanner.com> if you would like to see the original. Some of the code in the. ipynb files was made my him.