IN THIS README I HAVE PROVIDED SOME EXPLANATION OF

1) THE FUNCTIONS IN THE .IPYNB FILE

2) VERSIONS

3) APPROACH WITH tf.example protocol buffer

**Versions**

matplotlib - 3.2.2

numpy - 1.22.4

tensorflow - 2.9.1

json - 2.0.9

PIL.Image - 8.0.1

This .ipynb has the following functions with their uses as -

**query\_model** - This is the function to query a .tflite based model. The input to this function are the image location and the model location. The output of this function will be equivalent to the output of the .tflite model i.e., x and y coordinates.

**query\_model\_np** - This is the function to query a .tflite based model. The input to this function are the numpy array of the image and the model location. The output of this function will be equivalent to the output of the .tflite model i.e., x and y coordinates.

**comp\_visualize** - This function is to visualize the output vs the ground truth values of the image. In this function it is assumed that the image path provided, should be stored in a file structure similar to gazecapture and contain a respective json file to extract the ground truth values. For this I have provided one image from which we can test the same.

Path of this test image - ./task\_1\_GSoC/gazetrack/test/images/03467\_\_00771\_1.jpeg

*tf.example protocol buffer case -*

*It is mentioned in the supplementary material provided by google, that "The dataset for base model training is prepared as a tf.Example protocol buffer”. We can also do the same, no issues with this. But to do this we need to first finalize upon a dataset we wish to convert to this tf.Example protocol buffer. We can convert a whole test set of a certain dataset to this tf.Example protocol buffer, and then perform batch wise testing to get a better idea of the functioning of the google model.*