Steps to be followed for Object detection:

* Choose appropriate pretrained model from tensorflow object detection zoo.

Link : <https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md>

Model should be chosen keeping in mind the execution time and accuracy of the pre-trained model. After choosing the model, we must replace the link of the .tar.gz model file in PRETRAINED\_MODEL\_URL.

Refer the colab notebook which can be found in AI/ML//Object Detection//

* Then we create the tensorflow folder structure and Download TF Models Pretrained Models from Tensorflow Model Zoo and Install TFOD.
* Create Label Map, Labels are to be edited as per the name(found in the xml) of the object.
* Create TF Records, for this we need to create two folders, ‘train’ and ‘test’ and put the xml’s of the respective images in the train and test folders and then run the script to create TF records.

Note: We may have to edit the ‘generate\_tfrecord.py’ script to get the correct node which would contain the xmin,ymin,xmax and ymax.

* Copy model config file to training folder.
* Update config file for transfer learning by default we need to add fine\_tune\_checkpoint, label\_map\_path, input\_path for train and eval nodes and also update some more parameters which is done via colab.
* Next step is the train the model, in the train command we can change the training steps. i.e (number of images/ batch\_size) \* x = training steps. For Example if we have 400 images and a batch size of 4 then, 400/4 = 100 steps for one epoch. Checkpoints will be created after every 1000 steps and the latest checkout should be used in the eval or testing part.
* We can optionally evaluate the model via the eval section in colab. For some reason we face an infinite loop in waiting for new checkpoint. But we still get the eval metrices in a table format. Link for metric understanding : <https://towardsdatascience.com/on-object-detection-metrics-with-worked-example-216f173ed31e>
* Next we can load the latest checkpoint of the model and test how well the model works by passing an image to it and checking the confidence of the detection, the min threshold for confidence can be configured and is by default set to 0.8, which means only detection with 80% will be shown.
* The train and test metric can be also view in tensorboard, to see this we must navigate to ‘Tensorflow\workspace\models\my\_ssd\_mobnet\train’ and run ‘tensorboard –logdir=.’ Similarly for eval.