**OBJECT DETECTION USING TENSORFLOW**

**Step 1: Collect Images**

Scrape images from the web using any web scrapping tool such that all images are of fixed size.

**Step 2: Download Tensorflow/models repository**

Clone or download the Tesorflow/model repository and build it.

Repository download link:

<https://github.com/tensorflow/models>

**Step 3: Clone or Download the Object Detection repository**

Clone or download the following repository:

Repository link:

<https://github.com/Tony607/object_detection_demo>

**Step 4: Annotate the data using Labelimg tool:**

Annotate the images with Labelimg tool and store them in object\_detection\_demo/data/images/train and object\_detection\_demo/data/images/test

folders.

**Step 5: Select a pre trained model:**

Select a pretrained model such as ssd\_mobilenet\_v2, faster\_rcnn\_inception\_v2 etc for training.

**Step 6: Generate TF Records:**

Generate a .csv file from the .xml file. Then convert the .csv file to TF record file for training the model.

**Step 7: Train the model:**

Train the model using the pipeline config file located at object\_detection/samples/configs/ .Make changes to the config file if required.Tune the hyperparameters Batch size,number of training steps and number of evaluation steps.

**Step 8: Generate the frozen graph and label.txt file:**

The model generates a frozen graph file at the location object\_detection/models/research/fine\_tuned\_model/ and labels.txt file at location object\_detection/data/annotations/ .

**Step 9: Test the model:**

Store some images at object\_detection\_demo/test/ and run the python script

local\_inference\_test.ipynb .

**Step 10: Test the model on Android Studio:**

Import a new project using the directory from the TensorFlow repo , called “Android”.Change the nativeBuildSystem variable of the Gradle build to none.

When the build is finished, add the frozen model to the assets directory. Then, in that folder, create a file called “labels”, and in the first line write ???, in the subsequent lines write the names of the classes.

Then, open the file named “DetectionActitivity.java” located in the “java” directory ,this is the code used by the app to perform the detections. Look for the variable TF\_OD\_API\_MODEL\_FILE and TF\_OD\_API\_LABELS\_FILE and in the first one, change its value to the path of the frozen model located in the assets folder, and in the second one write the path of the file with the labels. Another useful variable you should know of is MINIMUM\_CONFIDENCE\_TF\_OD\_API which is the minimum confidence needed it to track a detection