1. INSTALL GIT   
   Link - [Git - Downloads (git-scm.com)](https://git-scm.com/downloads)
2. INSTALL PYTHON 3.7.7  
   Link - https://www.python.org/ftp/python/3.7.7/python-3.7.7-amd64.exe
3. INSTALL PYTHON 3.7.7 & GPU DRIVERS  
   Make sure Python Path is added to System Variable  
   Python 3.7.7 = https://www.python.org/ftp/python/3.7.7/python-3.7.7-amd64.exe
4. Right Click open CMD at where you want to do your TRAINING & clone Repository  
     
   git clone <https://github.com/abhimanyu1990/SSD-Mobilenet-Custom-Object-Detector-Model-using-Tensorflow-2.git>
5. Rename the cloned folder to “tensorflow\_model”  
     
   mv .\SSD-Mobilenet-Custom-Object-Detector-Model-using-Tensorflow-2\ tensorflow\_model
6. cd tensorflow\_model
7. python -m venv env
8. env/Scripts/activate
9. pip install --user --upgrade pip
10. python -m pip install --upgrade pip
11. pip install tensorflow-gpu==2.2
12. pip install pillow
13. pip install lxml
14. pip install jupyter
15. pip install matplotlib
16. Now Copy “images (original)” folder to “tensorflow\_model” folder   
    Rename “images (original)” to “images”  
    images (original can be found in training folder drive)  
    Overrwrite the “images” folder in “tensorflow\_model” folder and delete info.py in test and train
17. Download “transform\_image\_resolution.py” from drive (training Folder) Copy it to “tensorflow\_model” folder  
      
    python transform\_image\_resolution.py -d images/test/ -s 800 600  
    python transform\_image\_resolution.py -d images/train/ -s 800 600  
      
    If image gets rotated rotated back using windows photo viewer
18. git clone <https://github.com/tzutalin/labelImg.git>
19. cd labelImg  
    pip install pyqt5==5.10.1
20. pip install pyqt5-tools==5.10.1.1.5
21. pip install pyqt5-plugins==5.14.0.1
22. pip install -r requirements/requirements-linux-python3.txt
23. pyrcc5 -o libs/resources.py resources.qrc
24. python labelImg.py
25. create directory data in workspace (tensorflow\_model/data) and from workspace home folder run the xml-to-csv.py script  
    mkdir data
26. pip install pandas
27. python xml-to-csv.py
28. git clone https://github.com/tensorflow/models.git
29. $ cd models/research
30. pip install protobuf==3.4.0
31. Download protoc <https://github.com/protocolbuffers/protobuf/releases/download/v3.12.4/protoc-3.12.4-win64.zip>  
    Extract and
32. Extract the contents of the zip file.
33. Bin directory contains the protoc.exe
34. Update the environment variable to point to the bin directory that contains the protoc.exe
35. Re-Open the cmd as the environment variable take effect only on new command window
36. You need to save the protoc.exe file which is located in the zip file to C:\Users\Satyarth\AppData\Local\Programs\Python\Python37\Scripts
37. .
38. $ protoc object\_detection/protos/\*.proto --python\_out=.
39. $ pip install tf\_slim
40. $ pip install pandas
41. pip install protobuf==3.8.0
42. cp -r models/official .
43. cp -r models/research/object\_detection .
44. COPY generate\_tfrecord.PY FROM TRAING DRIVE TO TENSOR FLOW FLODER
45. python -c 'import tensorflow as tf; print(tf.\_\_version\_\_)'
46. Version check for tensor flow  
    Download \

o install the prerequisites for GPU support in TensorFlow 2.1:

1. Install your latest GPU drivers.
2. Install [CUDA 10.1](https://developer.nvidia.com/cuda-10.1-download-archive-base).
   1. If the CUDA installer reports "you are installing an older driver version", you may wish to choose a custom installation and deselect some components. Indeed, note that software bundled with CUDA including GeForce Experience, PhysX, a Display Driver, and Visual Studio integration are not required by TensorFlow.
   2. Also note that TensorFlow requires a specific version of the CUDA Toolkit unless you build from source; for TensorFlow 2.1 and 2.2, this is currently version 10.1.
3. Install cuDNN.
   1. [Download cuDNN](https://developer.nvidia.com/rdp/cudnn-archive) v7.6.4 for CUDA 10.1. This will require you to sign up to the NVIDIA Developer Program.
   2. Unzip to a suitable location and add the bin directory to your PATH.
4. Install tensorflow by pip install tensorflow.
5. You [may need to restart your PC](https://stackoverflow.com/a/51112550/604687).
6. pip install tensorflow-object-detection-api
7. python generate\_tfrecord.py --csv\_input=data/train\_labels.csv --image\_dir=images\train --output\_path=data/train.record  
   python generate\_tfrecord.py --csv\_input=data/test\_labels.csv --image\_dir=images\test --output\_path=data/test.record

$mkdir training

$cd training

Object-detection.ipxt copy this file to training from drive

change the pipeline config accordingly. The changes I made in the config file is higlighted

model {

ssd {

num\_classes: \*\*1\*\*

image\_resizer {

fixed\_shape\_resizer {

height: 300

width: 300

}

cp object\_detection/model\_main\_tf2.py .

cp object\_detection/exporter\_main\_v2.py .

Using vs build tools install c++ Distrubution

pip install pycocotools

pip install scipy

pip install dataclasses

pip install pyyaml

mkdir trained-checkpoint  
pip install --upgrade protobuf

pip install lvis

pip install gin-config

pip install tensorflow\_addons==0.11.2  
  
The following steps describe how to build a cuDNN dependent program. In the following sections:

-your CUDA directory path is referred to as C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0

-your cuDNN directory path is referred to as <installpath>

1. Navigate to your <installpath> directory containing cuDNN.
2. Unzip the cuDNN package. -cudnn-9.0-windows7-x64-v7.zip or -cudnn-9.0-windows10-x64-v7.zip
3. Copy the following files into the CUDA Toolkit directory.
   * Copy <installpath>\cuda\bin\cudnn64\_7.dll to C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\bin.
   * Copy <installpath>\cuda\ include\cudnn.h to C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\include.
   * Copy <installpath>\cuda\lib\x64\cudnn.lib to C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\lib\x64.

pip install tensorflow\_addons

pip install pycocotools

pip uninstall numpy

pip install numpy==1.18.2pip insta

pip install pycocotools-windowsll gin-config

pip install gin-config

pip install pycocotools-windows

pip install tensorflow==2.2

python model\_main\_tf2.py --pipeline\_config\_path=ssd\_mobilenet\_v2\_320x320\_coco17\_tpu-8/pipeline.config --model\_dir=trained-checkpoint --alsologtostderr --num\_train\_steps=4100 --sample\_1\_of\_n\_eval\_examples=1 --num\_eval\_steps=1  
  
Open another cmd

env/Scripts/activate

tensorboard --logdir=trained-checkpoint

mkdir exported-model

pip install tensorflow-gpu==2.4.1

python exporter\_main\_v2.py --input\_type image\_tensor --pipeline\_config\_path ./ssd\_mobilenet\_v2\_320x320\_coco17\_tpu-8/pipeline.config --trained\_checkpoint\_dir ./trained-checkpoint --output\_directory exported-model/mobile-model

cp object\_detection/colab\_tutorials/object\_detection\_tutorial.ipynb .

jupyter notebook