

TFOD STEPS (windows,CPU)

1. Download four required folders and unzip them. Place all the folders in parent folder. (download from the link below)
 - <https://github.com/tensorflow/models/tree/v1.13.0>
the above link contains the required research models, samples folders and a tutorial folder
 - https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md
from the above link you can download any model according to your use case
 - https://drive.google.com/file/d/12F5oGAuQg7qBM_267TCMt_rlorV-M7gf/view?usp=sharing
this link contains the required utils folder and Dataset
 - <https://tzutalin.github.io/labellmg/>
this link contains the labeling tool for labeling the images (download the last one)
2. Create a virtual environment by running the cmd in the anaconda prompt "conda create -n name_of_the_env python=3.6"
3. Activate the virtual environment by" conda activate name_of_he_env"
4. Install the required packages by "pip install pillow lxml Cython contextlib2 jupyter matplotlib pandas opencv-python tensorflow==1.14.0"
5. Change the working directory of the anaconda prompt to the research folder
6. Install the protobuf package by "conda install -c anaconda protobuf"
7. To convert the protobuf files into py files run "protoc object_detection/protos/*.proto --python_out=."
8. To install the object detection in your local system run" python setup.py install"
9. To verify everything got installed correctly, run the "object_detection_tutorial.ipynb" file inside the object_detection folder.(there should be an image at end of the execution)

Note: you have to run the note book inside your virtual environment.

10. Move the downloaded model (example: faster_rcnn) to the research folder(research folder is inside the model folder)
11. Move all files(4) inside the utlis folder to the research folder
12. To convert the XML files of train and test data to CSV run “python xml_to_csv.py”
13. To convert the CSV files into tfrecords run the cmd below

```
For train: python generate_tfrecord.py --csv_input=images/train_labels.csv --  
image_dir=images/train --output_path=train.record
```

```
For test: python generate_tfrecord.py --csv_input=images/test_labels.csv --  
image_dir=images/test --output_path=test.record
```

14. To change the name of the classes and add more classes go to train.record and edit (do the same for test.record)
15. Copy the train.py from object_detection/legacy to the research folder
16. Inside object_detection/samples/configs copy the config file according to your downloaded model and paste it inside the research/training folder.
17. Copy deployment and nets folder from research/slim into the research folder
18. Do the changes in the config file
 - Change the number of classes according to your classes
 - Change (line:107) fine_tune_checkpoint: to your model name (example “faster_rcnn/model.ckpt”)
 - Change (line:113) the number of steps according to you
 - Change (line:122) input_path: “train.record” (for train)
 - Change (line:124) label_map_path: “training/labelmap.pbtxt” (for train)
 - Change (line:136) input_path: “test.record” (for test)
 - Change (line:138) label_map_path: “training/labelmap.pbtxt” (for test)
19. To start the training run “python train.py --logtostderr --train_dir=training/ --
pipeline_config_path=training/YOUR_MODEL.config” (note: YOUR_MODEL should be your
model name example:”faster_rcnn_inception_v2_coc.config”)
20. To stop the training in-between press ctrl+c. to resume the training again give the above cmd

21. Copy the `export_inference_graph.py` file from `object_detection` to the `research` folder
22. To convert the `ckpt` files into `pb` format run “`python export_inference_graph.py --input_type image_tensor --pipeline_config_path training/YOUR CONFIG FILE --trained_checkpoint_prefix training/YOUR MODEL.CKPT FILE --output_directory my_model`” (note: change the texts which are in capital letters according to your configuration)