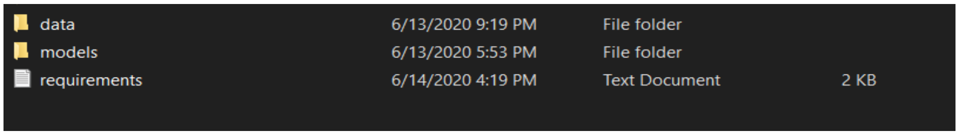
Mask-RCNN using TensorFlow on Windows

The steps for Mask-RCNN are similar to TFOD

Download the file and extract the file

Download link:
https://drive.google.com/file/d/1j2a6mvg4dLdm3vTKNRQlTmihc3Xf4hDi/view

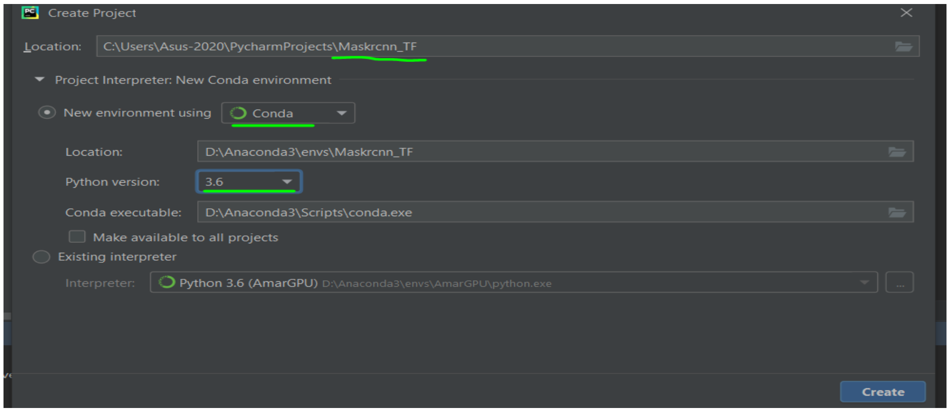
Once the file is downloaded then do the extraction process from the zip file.



data	6/13/2020 9:19 PM	File folder	
models	6/13/2020 5:53 PM	File folder	
requirements	6/14/2020 4:19 PM	Text Document	2 KB

Open PyCharm

File -- > New Project -- > Project name -- > conda environment -- > python version -- > click create



Location: C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF

Project Interpreter: New Conda environment

● New environment using Conda

Location: D:\Anaconda3\envs\Maskrcnn_TF

Python version: 3.6

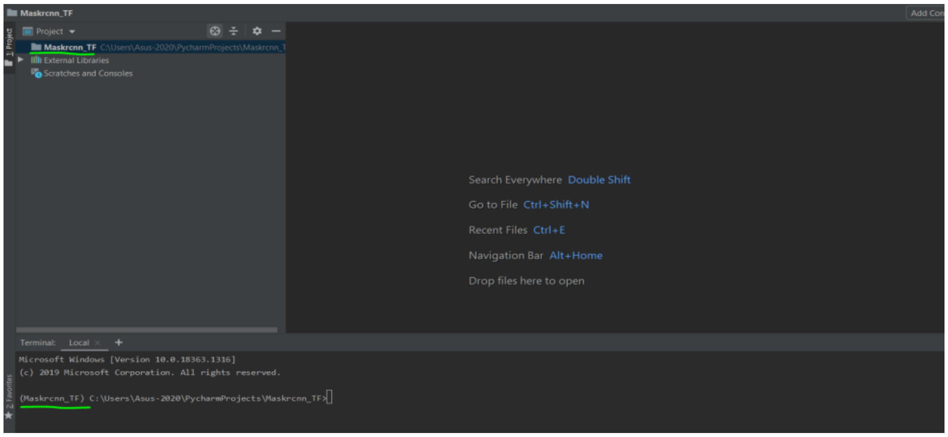
Conda executable: D:\Anaconda3\Scripts\conda.exe

□ Make available to all projects

○ Existing interpreter

Interpreter: Python 3.6 (AmarGPU) D:\Anaconda3\envs\AmarGPU\python.exe

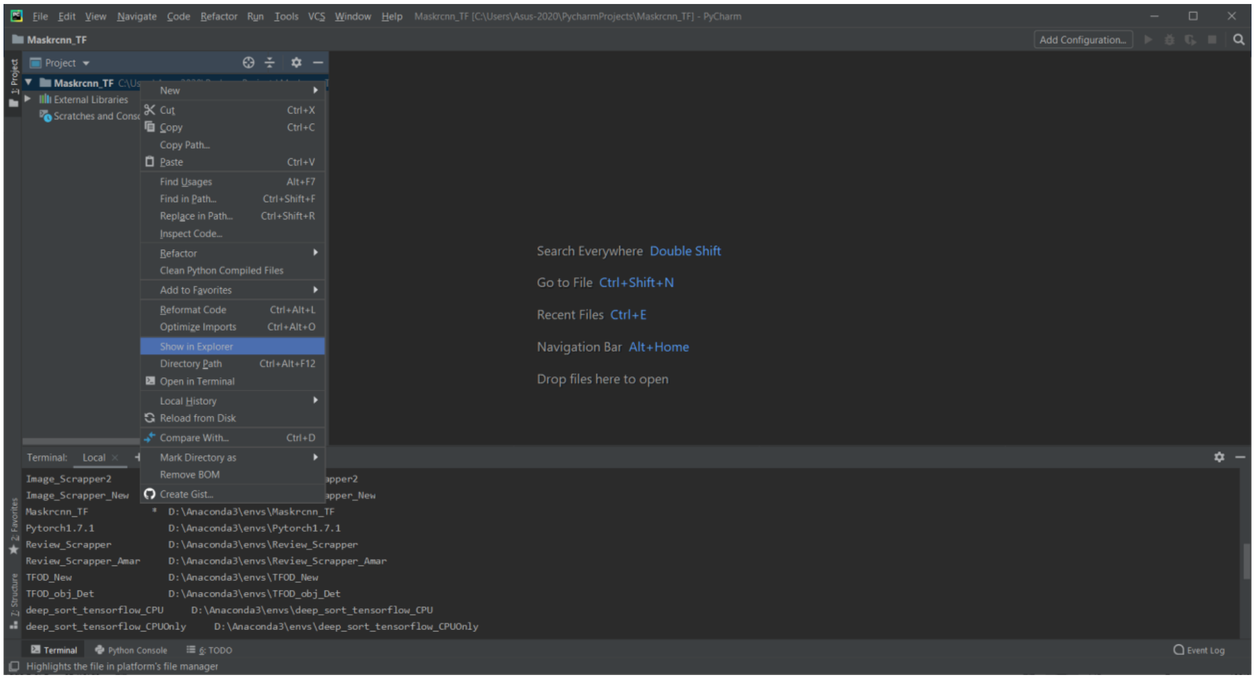
New environment and project are created.



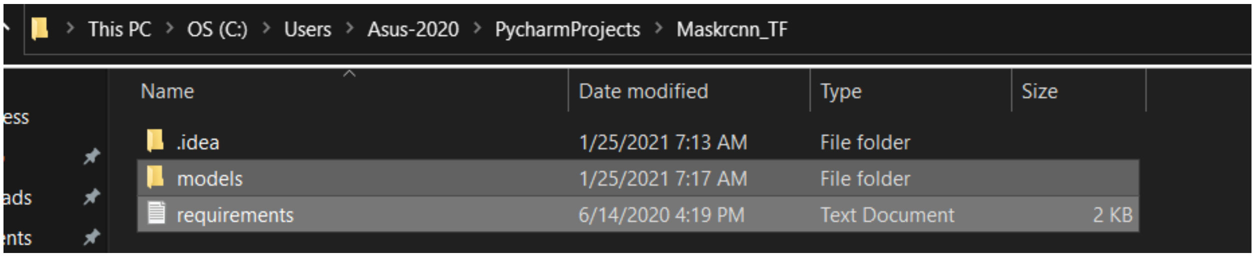
```
Microsoft Windows [Version 10.0.18363.1316]
(c) 2019 Microsoft Corporation. All rights reserved.

(Maskrcnn_TF) C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF>
```

Now copy the two files models folder and requirements.txt and paste it into the python project.

Right click on the Project -- > then select show in explorer

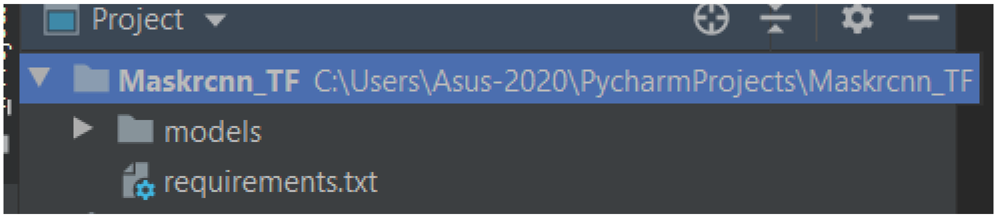


```
Image_Scrapper2                                    apper2
Image_Scrapper_New                                 apper_New
Maskrcnn_TF              *  D:\Anaconda3\envs\Maskrcnn_TF
Pytorch1.7.1                D:\Anaconda3\envs\Pytorch1.7.1
Review_Scrapper             D:\Anaconda3\envs\Review_Scrapper
Review_Scrapper_Amar        D:\Anaconda3\envs\Review_Scrapper_Amar
TFOD_New                    D:\Anaconda3\envs\TFOD_New
TFOD_obj_Det                D:\Anaconda3\envs\TFOD_obj_Det
deep_sort_tensorflow_CPU       D:\Anaconda3\envs\deep_sort_tensorflow_CPU
deep_sort_tensorflow_CPUOnly      D:\Anaconda3\envs\deep_sort_tensorflow_CPUOnly
```


This PC > OS (C:) > Users > Asus-2020 > PycharmProjects > Maskrcnn_TF
ess
ads
nts

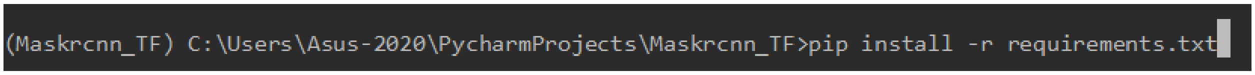
Name	Date modified	Type	Size
.idea	1/25/2021 7:13 AM	File folder	
models	1/25/2021 7:17 AM	File folder	
requirements	6/14/2020 4:19 PM	Text Document	2 KB

I have moved those files in to the Maskrcnn project location.

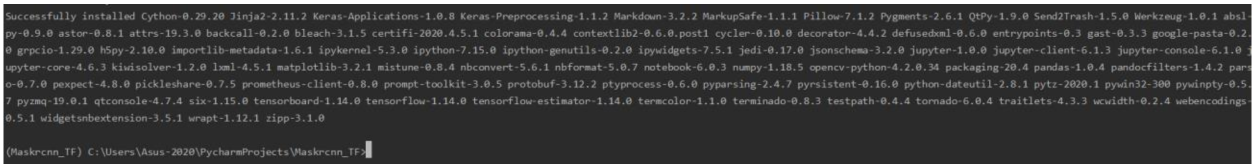


In PyCharm also it has updated the folders in our project section.

Now we need to install the project dependencies, in order to do that go the terminal in PyCharm and execute the command pip install -r requirements.txt



```
(Maskrcnn_TF) C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF>pip install -r requirements.txt
```



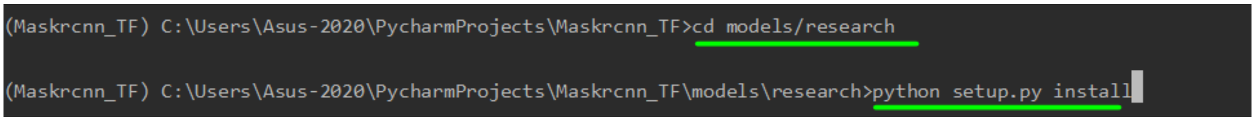
```
Successfully installed Cython-0.29.20 Jinja2-2.11.2 Keras-Applications-1.0.8 Keras-Preprocessing-1.1.2 Markdown-3.2.2 MarkupSafe-1.1.1 Pillow-7.1.2 Pygments-2.6.1 QtPy-1.9.0 Send2Trash-1.5.0 Werkzeug-1.0.1 absl-
py-0.9.0 astor-0.8.1 attrs-19.3.0 backcall-0.2.0 bleach-3.1.5 certifi-2020.4.5.1 colorama-0.4.4 contextlib2-0.6.0.post1 cycler-0.10.0 decorator-4.4.2 defusedxml-0.6.0 entrypoints-0.3 gast-0.3.3 google-pasta-0.2.
0 grpcio-1.29.0 h5py-2.10.0 importlib-metadata-1.6.1 ipykernel-5.3.0 ipython-7.15.0 ipython-genutils-0.2.0 ipywidgets-7.5.1 jedi-0.17.0 jsonschema-3.2.0 jupyter-1.0.0 jupyter-client-6.1.3 jupyter-console-6.1.0 j
upyter-core-4.6.3 kiwisolver-1.2.0 lxml-4.5.1 matplotlib-3.2.1 mistune-0.8.4 nbconvert-5.6.1 nbformat-5.0.7 notebook-6.0.3 numpy-1.18.5 opencv-python-4.2.0.34 packaging-20.4 pandas-1.0.4 pandocfilters-1.4.2 pars
o-0.7.0 pexpect-4.8.0 pickleshare-0.7.5 prometheus-client-0.8.0 prompt-toolkit-3.0.5 protobuf-3.12.2 ptyprocess-0.6.0 pyparsing-2.4.7 pyrsistent-0.16.0 python-dateutil-2.8.1 pytz-2020.1 pywin32-300 pywinpty-0.5.
7 pyzmq-19.0.1 qtconsole-4.7.4 six-1.15.0 tensorboard-1.14.0 tensorflow-1.14.0 tensorflow-estimator-1.14.0 termcolor-1.1.0 terminado-0.8.3 testpath-0.4.4 tornado-6.0.4 traitlets-4.3.3 wcwidth-0.2.4 webencodings-
0.5.1 widgetsnbextension-3.5.1 wrapt-1.12.1 zipp-3.1.0

(Maskrcnn_TF) C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF>
```

Here the installation of packages are installed successfully in our environment.

Setup.py

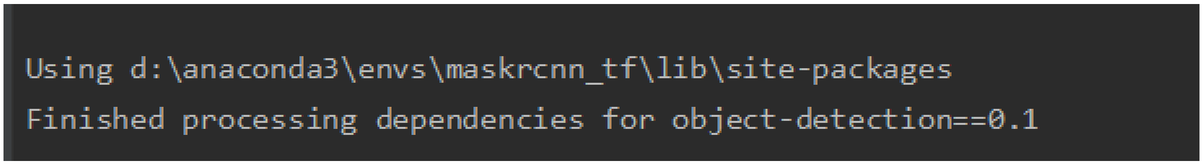
There is a setup.py file inside the research folder.

Navigate to the path cd/models/research inside the terminal



```
(Maskrcnn_TF) C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF>cd models/research

(Maskrcnn_TF) C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF\models\research>python setup.py install
```

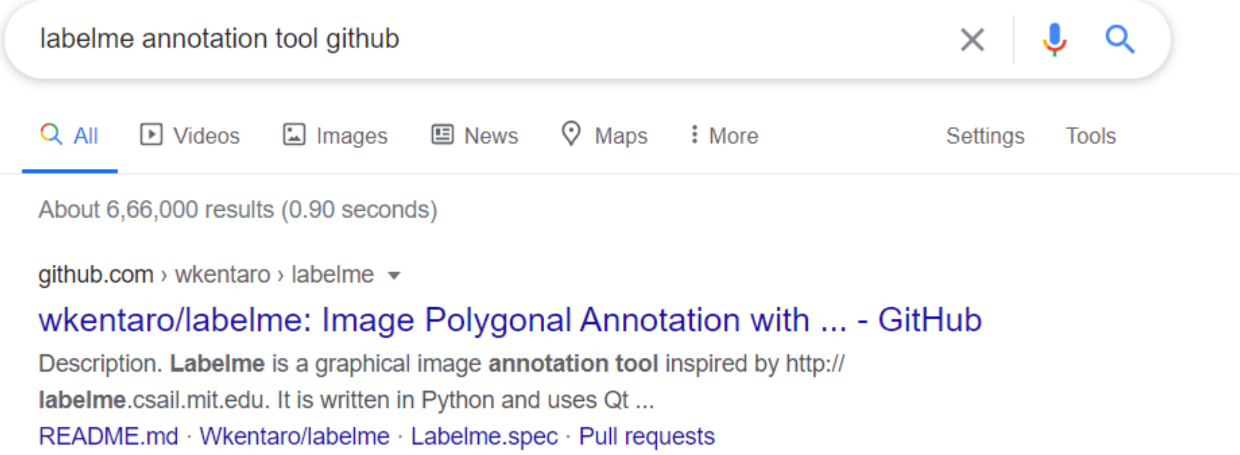
Successfully done



```
Using d:\anaconda3\envs\maskrcnn_tf\lib\site-packages
Finished processing dependencies for object-detection==0.1
```

Annotation:

Name of the tool: LabelMe



About 6,66,000 results (0.90 seconds)

github.com › wkentaro › labelme

wkentaro/labelme: Image Polygonal Annotation with ... - GitHub

Description. **Labelme** is a graphical image **annotation tool** inspired by http://**labelme**.csail.mit.edu. It is written in Python and uses Qt ...

README.md · Wkentaro/labelme · Labelme.spec · Pull requests

Link to download the specific versions:

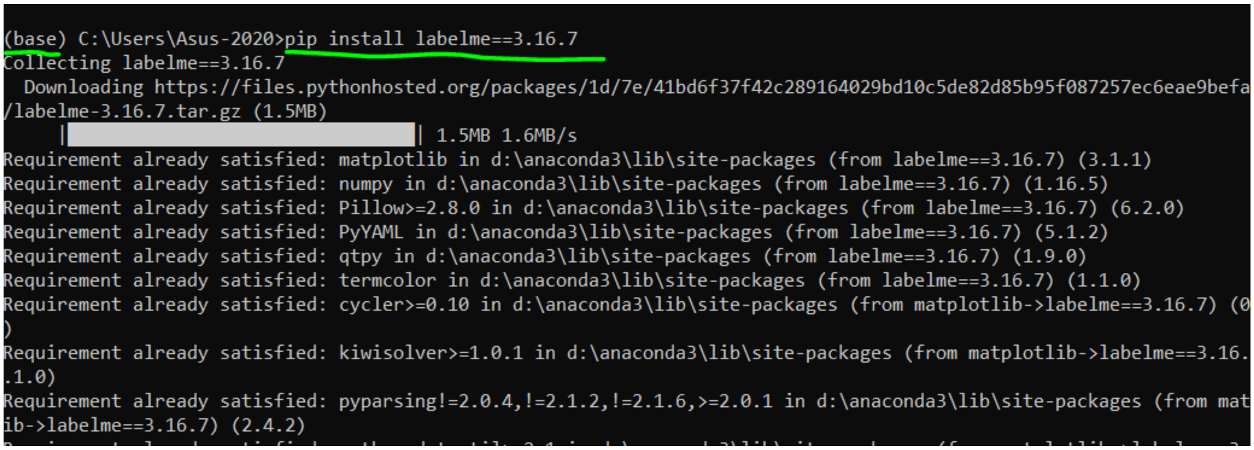
https://github.com/wkentaro/labelme/releases

Don’t download the latest version.

Here I’m going to install the specific version 3.16.7

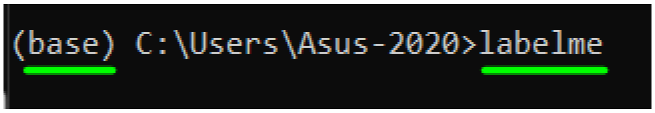
In order to install the package of this LableMe, create a new anaconda environment or you can proceed inside the base environment.

Here I have installed into my base environment

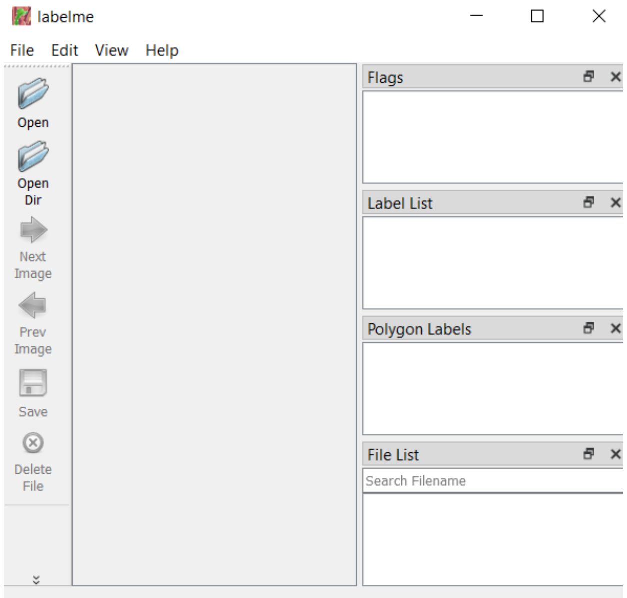


```
(base) C:\Users\Asus-2020>pip install labelme==3.16.7
Collecting labelme==3.16.7
  Downloading https://files.pythonhosted.org/packages/1d/7e/41bd6f37f42c289164029bd10c5de82d85b95f087257ec6eae9befa
/labelme-3.16.7.tar.gz (1.5MB)
     |████████████████████████████████| 1.5MB 1.6MB/s
Requirement already satisfied: matplotlib in d:\anaconda3\lib\site-packages (from labelme==3.16.7) (3.1.1)
Requirement already satisfied: numpy in d:\anaconda3\lib\site-packages (from labelme==3.16.7) (1.16.5)
Requirement already satisfied: Pillow>=2.8.0 in d:\anaconda3\lib\site-packages (from labelme==3.16.7) (6.2.0)
Requirement already satisfied: PyYAML in d:\anaconda3\lib\site-packages (from labelme==3.16.7) (5.1.2)
Requirement already satisfied: qtpy in d:\anaconda3\lib\site-packages (from labelme==3.16.7) (1.9.0)
Requirement already satisfied: termcolor in d:\anaconda3\lib\site-packages (from labelme==3.16.7) (1.1.0)
Requirement already satisfied: cycler>=0.10 in d:\anaconda3\lib\site-packages (from matplotlib->labelme==3.16.7) (0
)
Requirement already satisfied: kiwisolver>=1.0.1 in d:\anaconda3\lib\site-packages (from matplotlib->labelme==3.16.
.1.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in d:\anaconda3\lib\site-packages (from mat
ib->labelme==3.16.7) (2.4.2)
```

Open the LabelMe tool

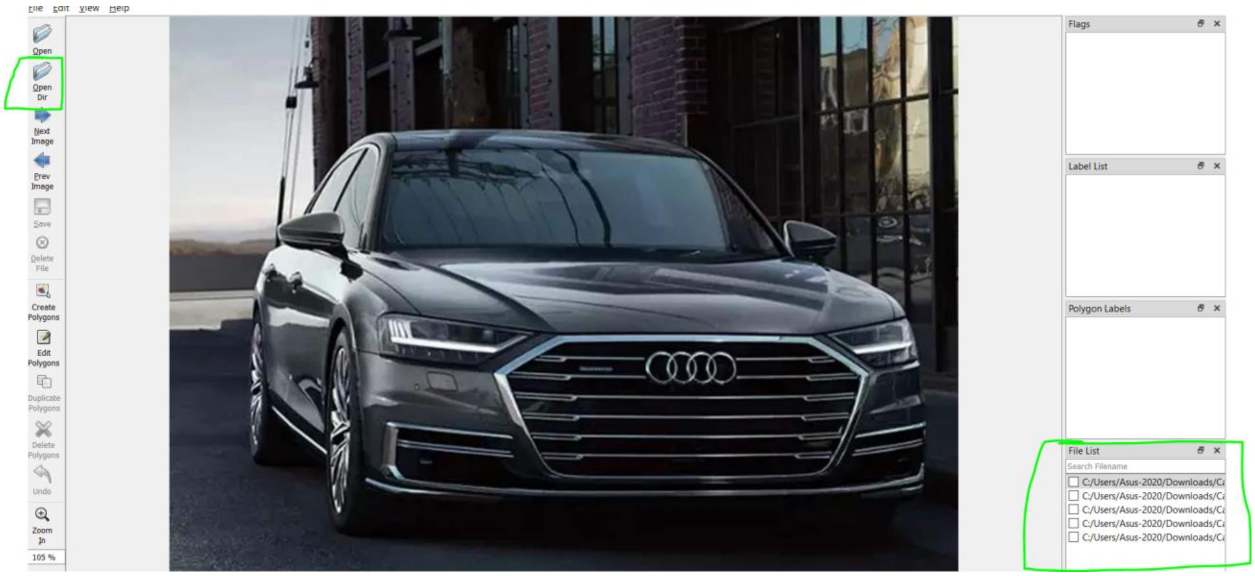


```
(base) C:\Users\Asus-2020>labelme
```

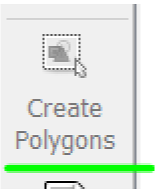


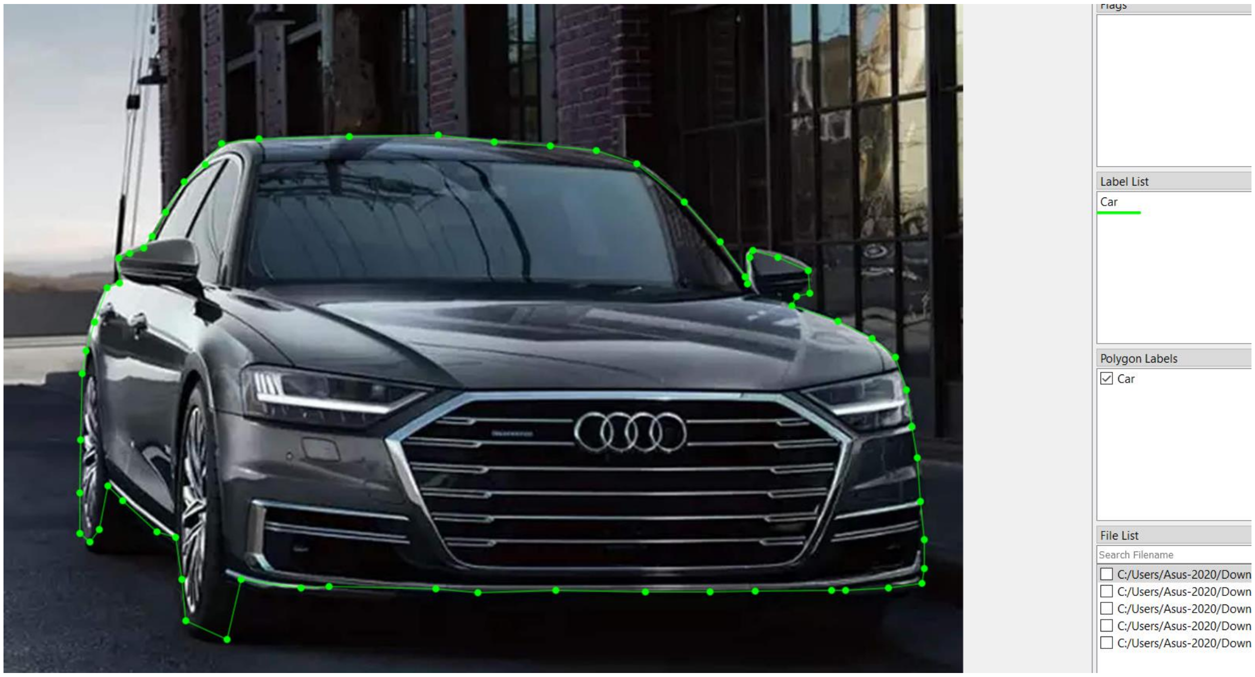
Open dir -- > select the folder to annotate

Sample annotations:

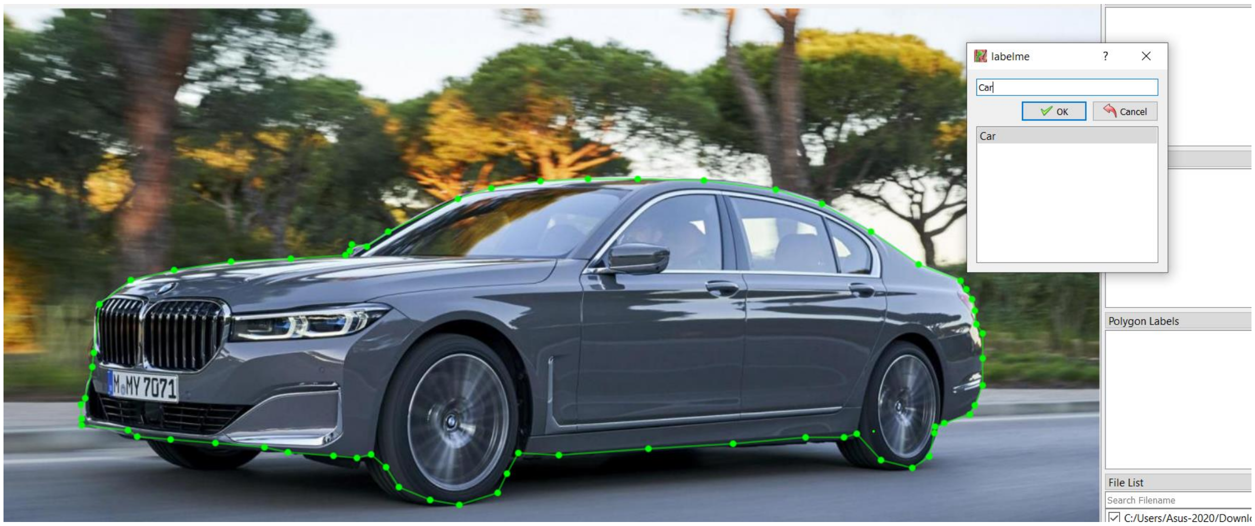


To do the annotation click on create polygons

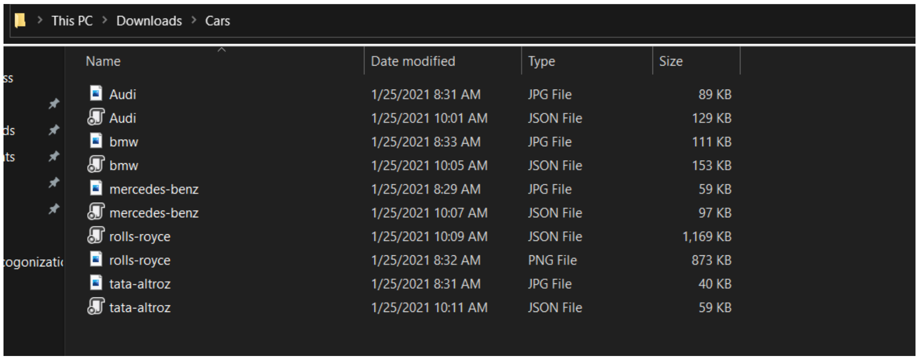



Flags
Label List
Car
Polygon Labels
☑ Car
File List
Search Filename
□ C:/Users/Asus-2020/Down
□ C:/Users/Asus-2020/Down
□ C:/Users/Asus-2020/Down
□ C:/Users/Asus-2020/Down
□ C:/Users/Asus-2020/Down

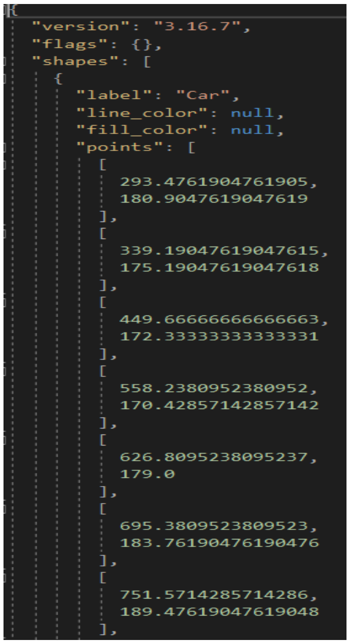
Save the file



Once you annotated for all the images in that folder then corresponding .json files are created for each and every image.



Name	Date modified	Type	Size
Audi	1/25/2021 8:31 AM	JPG File	89 KB
Audi	1/25/2021 10:01 AM	JSON File	129 KB
bmw	1/25/2021 8:33 AM	JPG File	111 KB
bmw	1/25/2021 10:05 AM	JSON File	153 KB
mercedes-benz	1/25/2021 8:29 AM	JPG File	59 KB
mercedes-benz	1/25/2021 10:07 AM	JSON File	97 KB
rolls-royce	1/25/2021 10:09 AM	JSON File	1,169 KB
rolls-royce	1/25/2021 8:32 AM	PNG File	873 KB
tata-altroz	1/25/2021 8:31 AM	JPG File	40 KB
tata-altroz	1/25/2021 10:11 AM	JSON File	59 KB



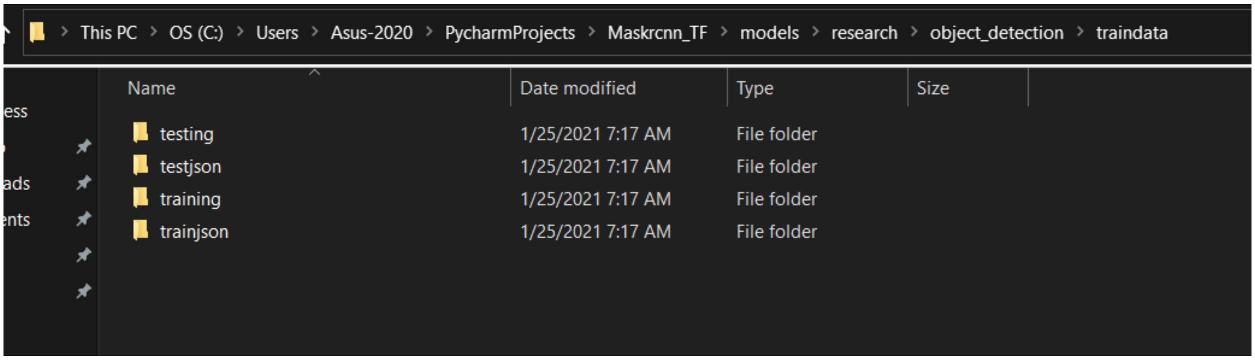
```
{
  "version": "3.16.7",
  "flags": {},
  "shapes": [
    {
      "label": "Car",
      "line_color": null,
      "fill_color": null,
      "points": [
        [
          293.4761904761905,
          180.9047619047619
        ],
        [
          339.19047619047615,
          175.19047619047618
        ],
        [
          449.66666666666663,
          172.33333333333331
        ],
        [
          558.2380952380952,
          170.42857142857142
        ],
        [
          626.8095238095237,
          179.0
        ],
        [
          695.3809523809523,
          183.76190476190476
        ],
        [
          751.5714285714286,
          189.47619047619048
        ],
```

Three files are important – 1. TF record file, 2. Labelmap file, 3. dataset

The data's inside the JSON file looks like above.

Actually our dataset should be in the form of training and test, which is located or created inside the

pycharmProjects -- > Projectname (Maskrcnn_TF) -- > models -- > research -- > object_detection -- > traindata



This PC > OS (C:) > Users > Asus-2020 > PycharmProjects > Maskrcnn_TF > models > research > object_detection > traindata

Name	Date modified	Type	Size
testing	1/25/2021 7:17 AM	File folder	
testjson	1/25/2021 7:17 AM	File folder	
training	1/25/2021 7:17 AM	File folder	
trainjson	1/25/2021 7:17 AM	File folder	

Here we have 4 folders 2 for test and 2 for train (dog class)

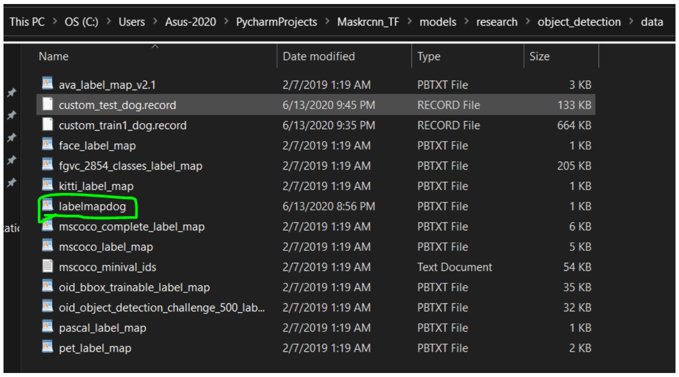
Testing and training folder contains only images

Testjson and trainjson contains the json files

According to the community of tensorflow all the lablemap.pbtxt are available inside the data folder.

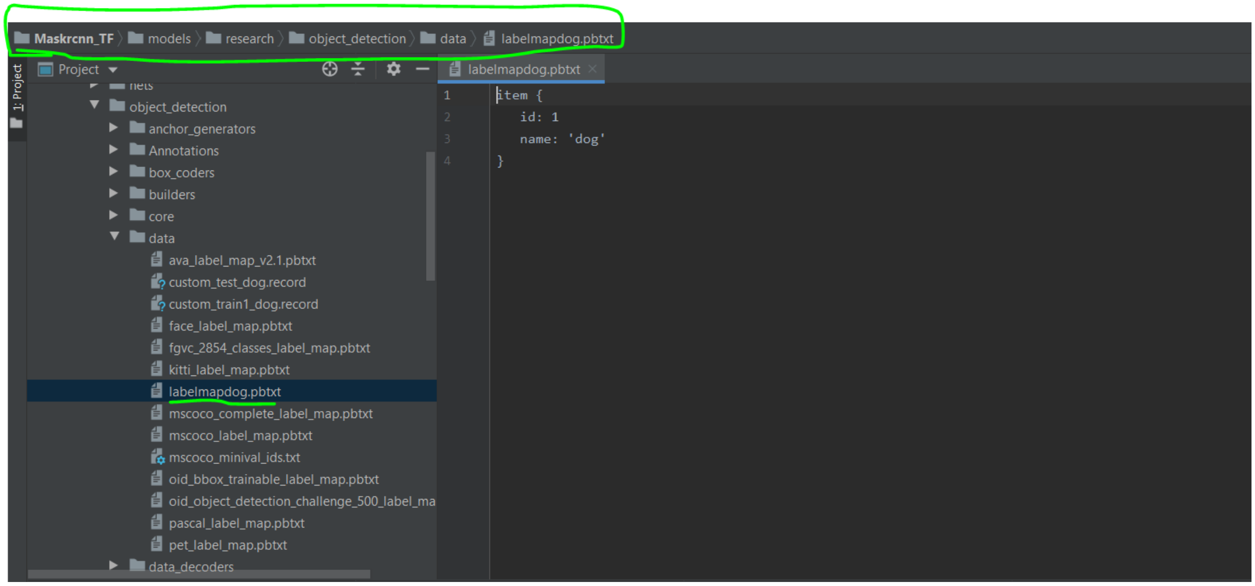
pycharmProjects -- > Projectname (Maskrcnn_TF) -- > models -- > research -- > object_detection -- > data -- > lablemapdog.pbtxt

In that data folder we can able to see many .pbtxt files and I have used only the lablemapdog.pbtxt file.



This PC > OS (C:) > Users > Asus-2020 > PycharmProjects > Maskrcnn_TF > models > research > object_detection > data

Name	Date modified	Type	Size
ava_label_map_v2.1	2/7/2019 1:19 AM	PBTXT File	3 KB
custom_test_dog.record	6/13/2020 9:45 PM	RECORD File	133 KB
custom_train1_dog.record	6/13/2020 9:35 PM	RECORD File	664 KB
face_label_map	2/7/2019 1:19 AM	PBTXT File	1 KB
fgvc_2854_classes_label_map	2/7/2019 1:19 AM	PBTXT File	205 KB
kitti_label_map	2/7/2019 1:19 AM	PBTXT File	1 KB
labelmapdog	6/13/2020 8:56 PM	PBTXT File	1 KB
mscoco_complete_label_map	2/7/2019 1:19 AM	PBTXT File	6 KB
mscoco_label_map	2/7/2019 1:19 AM	PBTXT File	5 KB
mscoco_minival_ids	2/7/2019 1:19 AM	Text Document	54 KB
oid_bbox_trainable_label_map	2/7/2019 1:19 AM	PBTXT File	35 KB
oid_object_detection_challenge_500_lab...	2/7/2019 1:19 AM	PBTXT File	32 KB
pascal_label_map	2/7/2019 1:19 AM	PBTXT File	1 KB
pet_label_map	2/7/2019 1:19 AM	PBTXT File	2 KB

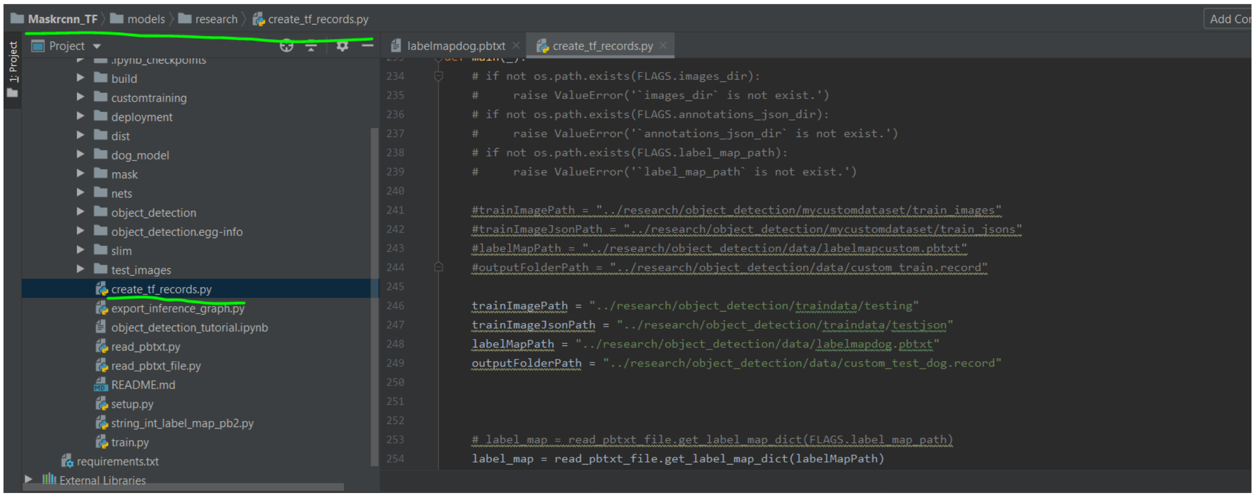


```
item {
  id: 1
  name: 'dog'
}
```

In this file we need to mention the number of classes.

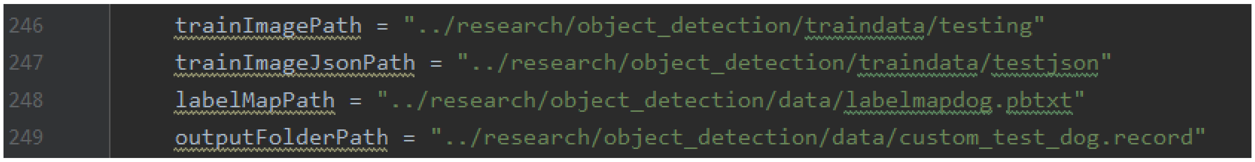
Generate TF record:

pycharmProjects -- > Projectname (Maskrcnn_TF) -- > models -- > research -- > create_tf_records.py



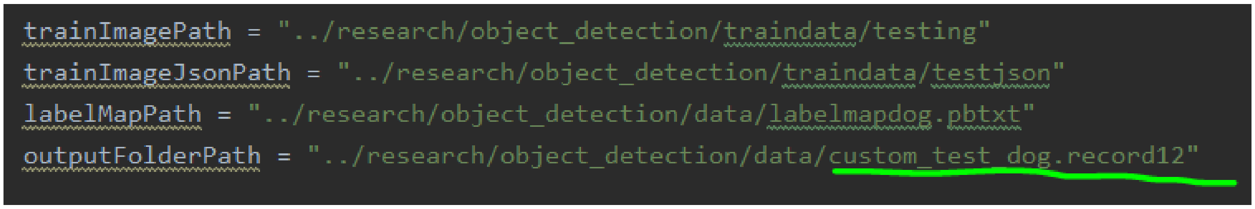
```
234     # if not os.path.exists(FLAGS.images_dir):
235     #     raise ValueError('`images_dir` is not exist.')
236     # if not os.path.exists(FLAGS.annotations_json_dir):
237     #     raise ValueError('`annotations_json_dir` is not exist.')
238     # if not os.path.exists(FLAGS.label_map_path):
239     #     raise ValueError('`label_map_path` is not exist.')
240
241     #trainImagePath = "../research/object_detection/mycustomdataset/train_images"
242     #trainImageJsonPath = "../research/object_detection/mycustomdataset/train_jsons"
243     #labelMapPath = "../research/object_detection/data/labelmapcustom.pbtxt"
244     #outputFolderPath = "../research/object_detection/data/custom_train.record"
245
246     trainImagePath = "../research/object_detection/traindata/testing"
247     trainImageJsonPath = "../research/object_detection/traindata/testjson"
248     labelMapPath = "../research/object_detection/data/labelmapdog.pbtxt"
249     outputFolderPath = "../research/object_detection/data/custom_test_dog.record"
250
251
252
253     # label_map = read_pbtxt_file.get_label_map_dict(FLAGS.label_map_path)
254     label_map = read_pbtxt_file.get_label_map_dict(labelMapPath)
```

In line number 246 to 249 this are the utmost important in the create_tf_records.py file



```
246     trainImagePath = "../research/object_detection/traindata/testing"
247     trainImageJsonPath = "../research/object_detection/traindata/testjson"
248     labelMapPath = "../research/object_detection/data/labelmapdog.pbtxt"
249     outputFolderPath = "../research/object_detection/data/custom_test_dog.record"
```

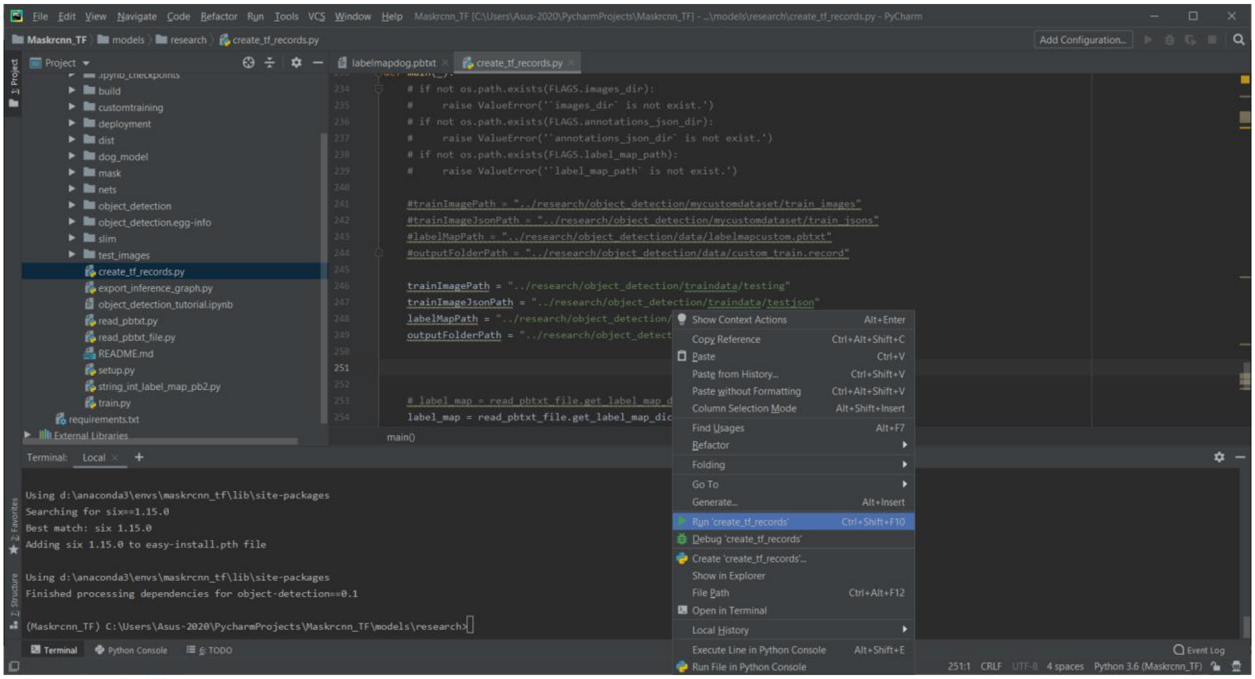
Here the output file (tf record) will be created or generated for the test data on the path which we have specified above



```
trainImagePath = "../research/object_detection/traindata/testing"
trainImageJsonPath = "../research/object_detection/traindata/testjson"
labelMapPath = "../research/object_detection/data/labelmapdog.pbtxt"
outputFolderPath = "../research/object_detection/data/custom_test_dog.record12"
```

Here I have changed the name of the tf record like "custom_test_dog.record12"

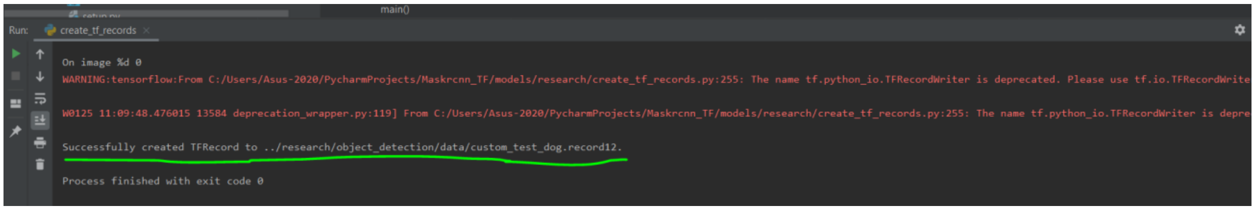
Now run the create_tf_records.py file



```
    # if not os.path.exists(FLAGS.images_dir):
    #     raise ValueError('`images_dir` is not exist.')
    # if not os.path.exists(FLAGS.annotations_json_dir):
    #     raise ValueError('`annotations_json_dir` is not exist.')
    # if not os.path.exists(FLAGS.label_map_path):
    #     raise ValueError('`label_map_path` is not exist.')

    #trainImagePath = "../research/object_detection/mycustomdataset/train_images"
    #trainImageJsonPath = "../research/object_detection/mycustomdataset/train_jsons"
    #labelMapPath = "../research/object_detection/data/labelmapcustom.pbtxt"
    #outputFolderPath = "../research/object_detection/data/custom_train.record"

    trainImagePath = "../research/object_detection/traindata/testing"
    trainImageJsonPath = "../research/object_detection/traindata/testjson"
    labelMapPath = "../research/object_detection/
    outputFolderPath = "../research/object_detect


    # label_map = read_pbtxt_file.get_label_map_d
    label_map = read_pbtxt_file.get_label_map_dic
```

```
Using d:\anaconda3\envs\maskrcnn_tf\lib\site-packages
Searching for six==1.15.0
Best match: six 1.15.0
Adding six 1.15.0 to easy-install.pth file

Using d:\anaconda3\envs\maskrcnn_tf\lib\site-packages
Finished processing dependencies for object-detection==0.1

(Maskrcnn_TF) C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF\models\research>
```

File Edit View Navigate Code Refactor Run Tools VCS Window Help Maskrcnn_TF [C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF] - ...\models\research\create_tf_records.py - PyCharm
Maskrcnn_TF
models
research
create_tf_records.py
Add Configuration...
Project
labelmapdog.pbtxt
create_tf_records.py
build
customtraining
deployment
dist
dog_model
mask
nets
object_detection
object_detection.egg-info
slim
test_images
create_tf_records.py
export_inference_graph.py
object_detection_tutorial.ipynb
read_pbtxt.py
read_pbtxt_file.py
README.md
setup.py
string_int_label_map_pb2.py
train.py
requirements.txt
External Libraries
main()
Terminal: Local
Show Context Actions Alt+Enter
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Paste from History... Ctrl+Shift+V
Paste without Formatting Ctrl+Alt+Shift+V
Column Selection Mode Alt+Shift+Insert
Find Usages Alt+F7
Refactor
Folding
Go To
Generate... Alt+Insert
Run 'create_tf_records' Ctrl+Shift+F10
Debug 'create_tf_records'
Create 'create_tf_records'...
Show in Explorer
File Path Ctrl+Alt+F12
Open in Terminal
Local History
Execute Line in Python Console Alt+Shift+E
Run File in Python Console
Terminal
Python Console
6: TODO
Event Log
251:1 CRLF UTF-8 4 spaces Python 3.6 (Maskrcnn_TF)



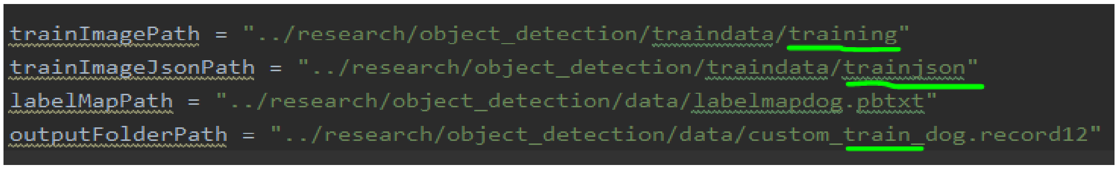
main()
Run: create_tf_records

```
On image %d 0
WARNING:tensorflow:From C:/Users/Asus-2020/PycharmProjects/Maskrcnn_TF/models/research/create_tf_records.py:255: The name tf.python_io.TFRecordWriter is deprecated. Please use tf.io.TFRecordWrite

W0125 11:09:48.476015 13584 deprecation_wrapper.py:119] From C:/Users/Asus-2020/PycharmProjects/Maskrcnn_TF/models/research/create_tf_records.py:255: The name tf.python_io.TFRecordWriter is depre

Successfully created TFRecord to ../research/object_detection/data/custom_test_dog.record12.

Process finished with exit code 0
```

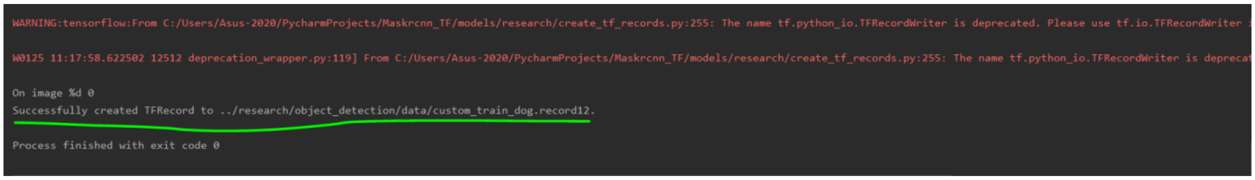
Tf record is successfully created.

I have created for test and now I have to create tf record for training and do some changes like below.

Save and run the file.

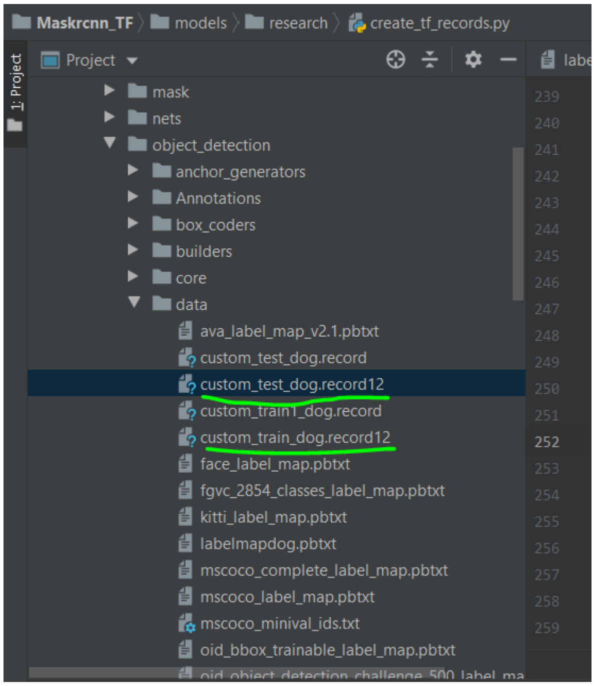


```
trainImagePath = "../research/object_detection/traindata/training"
trainImageJsonPath = "../research/object_detection/traindata/trainjson"
labelMapPath = "../research/object_detection/data/labelmapdog.pbtxt"
outputFolderPath = "../research/object_detection/data/custom_train_dog.record12"
```



```
WARNING:tensorflow:From C:/Users/Asus-2020/PycharmProjects/Maskrcnn_TF/models/research/create_tf_records.py:255: The name tf.python_io.TFRecordWriter is deprecated. Please use tf.io.TFRecordWriter i

W0125 11:17:58.622502 12512 deprecation_wrapper.py:119] From C:/Users/Asus-2020/PycharmProjects/Maskrcnn_TF/models/research/create_tf_records.py:255: The name tf.python_io.TFRecordWriter is deprecat

On image %d 0
Successfully created TFRecord to ../research/object_detection/data/custom_train_dog.record12.

Process finished with exit code 0
```

Two files are created.

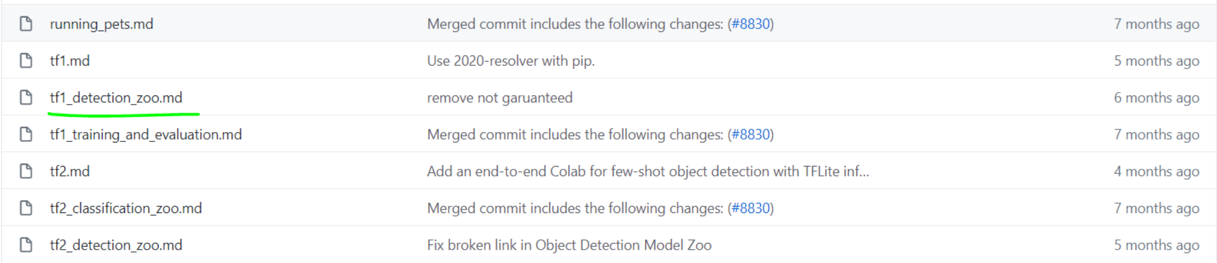


```
Maskrcnn_TF 〉 models 〉 research 〉 create_tf_records.py
Project ▼
    ▶ mask
    ▶ nets
    ▼ object_detection
        ▶ anchor_generators
        ▶ Annotations
        ▶ box_coders
        ▶ builders
        ▶ core
        ▼ data
            ava_label_map_v2.1.pbtxt
            custom_test_dog.record
            custom_test_dog.record12
            custom_train1_dog.record
            custom_train_dog.record12
            face_label_map.pbtxt
            fgvc_2854_classes_label_map.pbtxt
            kitti_label_map.pbtxt
            labelmapdog.pbtxt
            mscoco_complete_label_map.pbtxt
            mscoco_label_map.pbtxt
            mscoco_minival_ids.txt
            oid_bbox_trainable_label_map.pbtxt
            oid_object_detection_challenge_500_label_ma
```

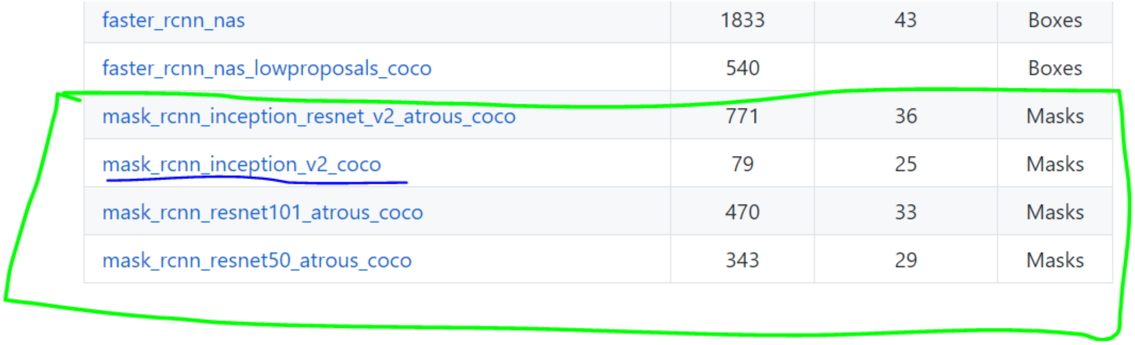
Files needed for Training:

Go to the model zoo

https://github.com/tensorflow/models/tree/master/research/object_detection/g3doc



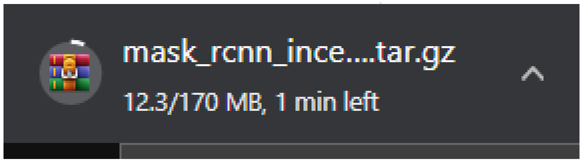
running_pets.md	Merged commit includes the following changes: (#8830)	7 months ago
tf1.md	Use 2020-resolver with pip.	5 months ago
tf1_detection_zoo.md	remove not garuanteed	6 months ago
tf1_training_and_evaluation.md	Merged commit includes the following changes: (#8830)	7 months ago
tf2.md	Add an end-to-end Colab for few-shot object detection with TFLite inf...	4 months ago
tf2_classification_zoo.md	Merged commit includes the following changes: (#8830)	7 months ago
tf2_detection_zoo.md	Fix broken link in Object Detection Model Zoo	5 months ago



faster_rcnn_nas	1833	43	Boxes
faster_rcnn_nas_lowproposals_coco	540		Boxes
mask_rcnn_inception_resnet_v2_atrous_coco	771	36	Masks
mask_rcnn_inception_v2_coco	79	25	Masks
mask_rcnn_resnet101_atrous_coco	470	33	Masks
mask_rcnn_resnet50_atrous_coco	343	29	Masks

Here I’m going to use mask_rcnn_inception_v2_coco

Click the second architecture, then it start the downloading process.

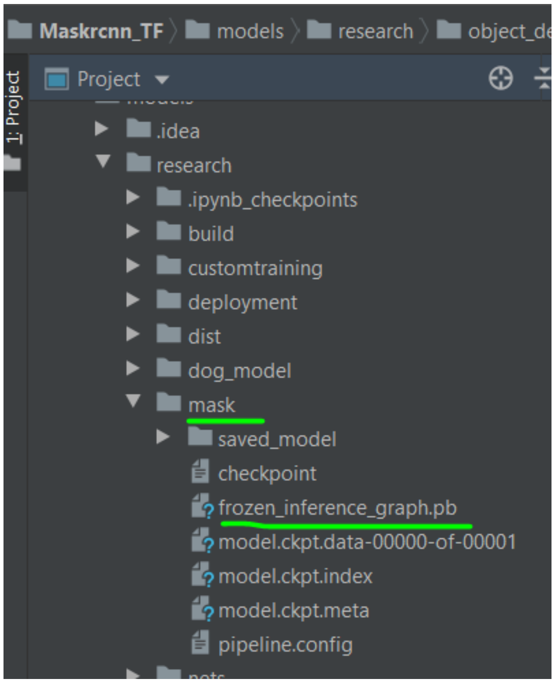


mask_rcnn_ince....tar.gz

12.3/170 MB, 1 min left

Once the file is downloaded then extract the file and if u want to rename the folder as well.

Here I have changed the name as mask and move the folder into the research folder.

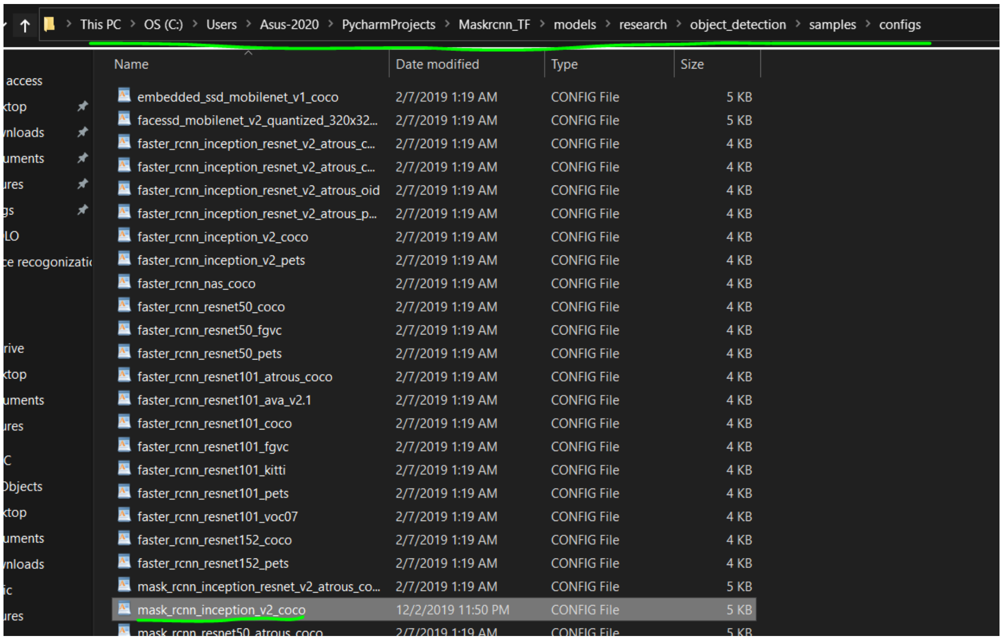


```
models
  .idea
  research
    .ipynb_checkpoints
    build
    customtraining
    deployment
    dist
    dog_model
    mask
      saved_model
      checkpoint
      frozen_inference_graph.pb
      model.ckpt.data-00000-of-00001
      model.ckpt.index
      model.ckpt.meta
      pipeline.config
    nets
```

Config file:

For corresponding each and every architecture we have dedicated config files are available.

pycharmProjects -- > Projectname (Maskrcnn_TF) -- > models -- > research -- > object_detection -- > samples -- > configs

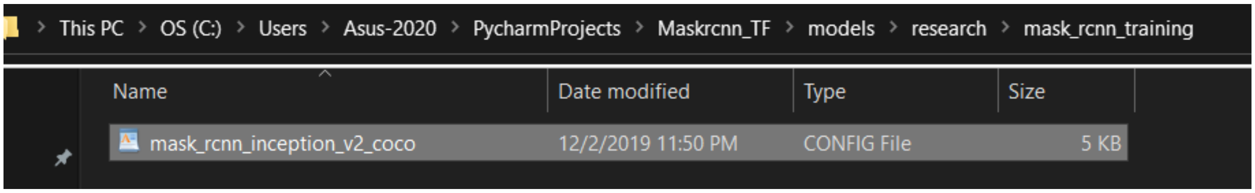

This PC › OS (C:) › Users › Asus-2020 › PycharmProjects › Maskrcnn_TF › models › research › object_detection › samples › configs
access
ktop
vnloads
uments
ures
gs
LO
ce recognizati
rive
ktop
uments
ures
C
Objects
ktop
uments
vnloads
ic
ures

Name	Date modified	Type	Size
embedded_ssd_mobilenet_v1_coco	2/7/2019 1:19 AM	CONFIG File	5 KB
facessd_mobilenet_v2_quantized_320x32...	2/7/2019 1:19 AM	CONFIG File	5 KB
faster_rcnn_inception_resnet_v2_atrous_c...	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_inception_resnet_v2_atrous_c...	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_inception_resnet_v2_atrous_oid	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_inception_resnet_v2_atrous_p...	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_inception_v2_coco	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_inception_v2_pets	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_nas_coco	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet50_coco	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet50_fgvc	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet50_pets	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet101_atrous_coco	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet101_ava_v2.1	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet101_coco	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet101_fgvc	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet101_kitti	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet101_pets	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet101_voc07	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet152_coco	2/7/2019 1:19 AM	CONFIG File	4 KB
faster_rcnn_resnet152_pets	2/7/2019 1:19 AM	CONFIG File	4 KB
mask_rcnn_inception_resnet_v2_atrous_co...	2/7/2019 1:19 AM	CONFIG File	5 KB
mask_rcnn_inception_v2_coco	12/2/2019 11:50 PM	CONFIG File	5 KB
mask_rcnn_resnet50_atrous_coco	2/7/2019 1:19 AM	CONFIG File	5 KB

The above marked one is the config file iam going to use.

Now we need to create a new folder called **“mask_rcnn_training”**

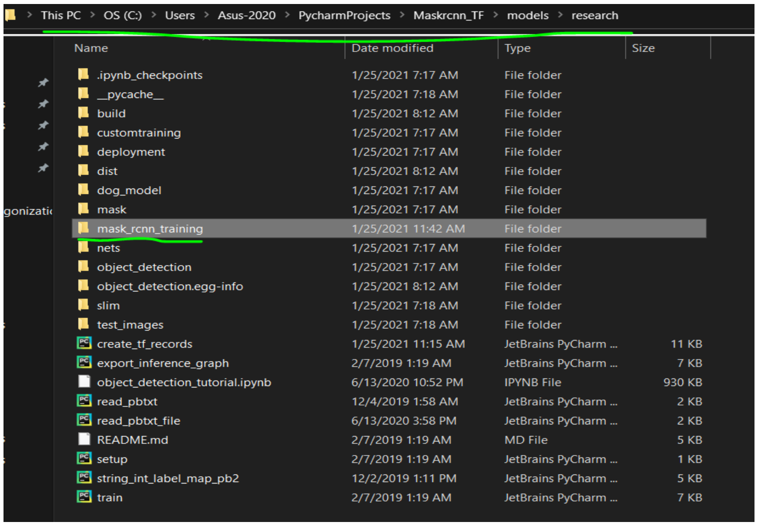
pycharmProjects -- > Projectname (Maskrcnn_TF) -- > models -- > research -- > **mask_rcnn_training (folder)**



This PC > OS (C:) > Users > Asus-2020 > PycharmProjects > Maskrcnn_TF > models > research > mask_rcnn_training

Name	Date modified	Type	Size
mask_rcnn_inception_v2_coco	12/2/2019 11:50 PM	CONFIG File	5 KB

Copy the labelmap.pbtxt and paste into the mask_rcnn_training folder

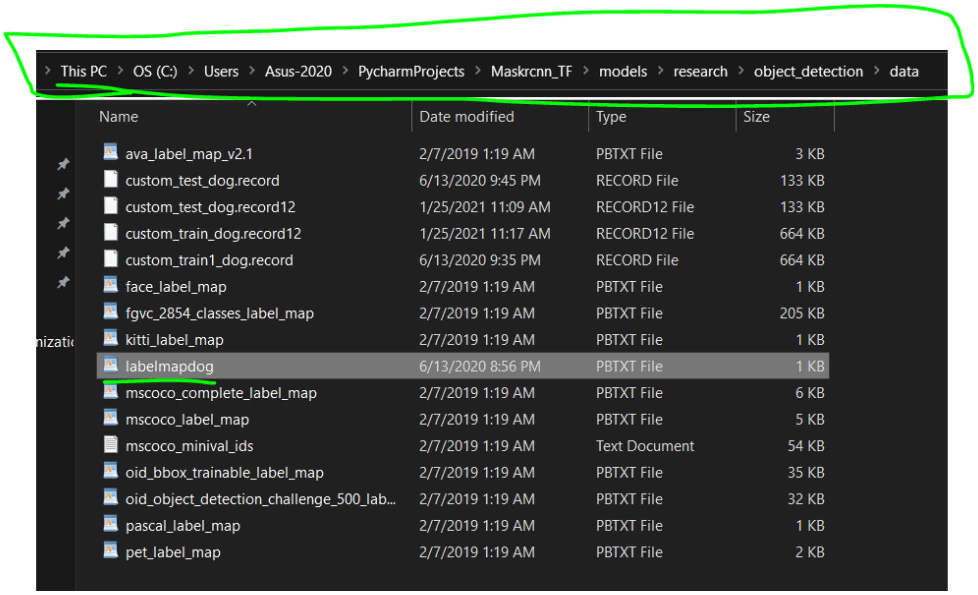


This PC > OS (C:) > Users > Asus-2020 > PycharmProjects > Maskrcnn_TF > models > research

Name	Date modified	Type	Size
.ipynb_checkpoints	1/25/2021 7:17 AM	File folder	
__pycache__	1/25/2021 7:18 AM	File folder	
build	1/25/2021 8:12 AM	File folder	
customtraining	1/25/2021 7:17 AM	File folder	
deployment	1/25/2021 7:17 AM	File folder	
dist	1/25/2021 8:12 AM	File folder	
dog_model	1/25/2021 7:17 AM	File folder	
mask	1/25/2021 7:17 AM	File folder	
mask_rcnn_training	1/25/2021 11:42 AM	File folder	
nets	1/25/2021 7:17 AM	File folder	
object_detection	1/25/2021 7:17 AM	File folder	
object_detection.egg-info	1/25/2021 8:12 AM	File folder	
slim	1/25/2021 7:18 AM	File folder	
test_images	1/25/2021 7:18 AM	File folder	
create_tf_records	1/25/2021 11:15 AM	JetBrains PyCharm ...	11 KB
export_inference_graph	2/7/2019 1:19 AM	JetBrains PyCharm ...	7 KB
object_detection_tutorial.ipynb	6/13/2020 10:52 PM	IPYNB File	930 KB
read_pbtxt	12/4/2019 1:58 AM	JetBrains PyCharm ...	2 KB
read_pbtxt_file	6/13/2020 3:58 PM	JetBrains PyCharm ...	2 KB
README.md	2/7/2019 1:19 AM	MD File	5 KB
setup	2/7/2019 1:19 AM	JetBrains PyCharm ...	1 KB
string_int_label_map_pb2	12/2/2019 1:11 PM	JetBrains PyCharm ...	5 KB
train	2/7/2019 1:19 AM	JetBrains PyCharm ...	7 KB

Copy the config file and paste into the mask_rcnn_training folder

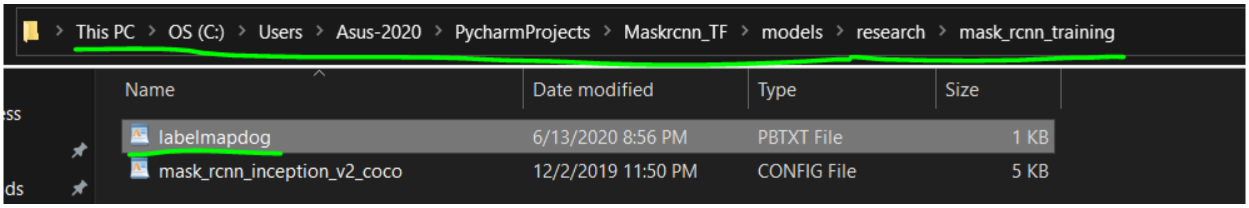
Copy the file from below location



This PC › OS (C:) › Users › Asus-2020 › PycharmProjects › Maskrcnn_TF › models › research › object_detection › data

Name	Date modified	Type	Size
ava_label_map_v2.1	2/7/2019 1:19 AM	PBTXT File	3 KB
custom_test_dog.record	6/13/2020 9:45 PM	RECORD File	133 KB
custom_test_dog.record12	1/25/2021 11:09 AM	RECORD12 File	133 KB
custom_train_dog.record12	1/25/2021 11:17 AM	RECORD12 File	664 KB
custom_train1_dog.record	6/13/2020 9:35 PM	RECORD File	664 KB
face_label_map	2/7/2019 1:19 AM	PBTXT File	1 KB
fgvc_2854_classes_label_map	2/7/2019 1:19 AM	PBTXT File	205 KB
kitti_label_map	2/7/2019 1:19 AM	PBTXT File	1 KB
labelmapdog	6/13/2020 8:56 PM	PBTXT File	1 KB
mscoco_complete_label_map	2/7/2019 1:19 AM	PBTXT File	6 KB
mscoco_label_map	2/7/2019 1:19 AM	PBTXT File	5 KB
mscoco_minival_ids	2/7/2019 1:19 AM	Text Document	54 KB
oid_bbox_trainable_label_map	2/7/2019 1:19 AM	PBTXT File	35 KB
oid_object_detection_challenge_500_lab...	2/7/2019 1:19 AM	PBTXT File	32 KB
pascal_label_map	2/7/2019 1:19 AM	PBTXT File	1 KB
pet_label_map	2/7/2019 1:19 AM	PBTXT File	2 KB

Paste the file in to the mask_rcnn_training

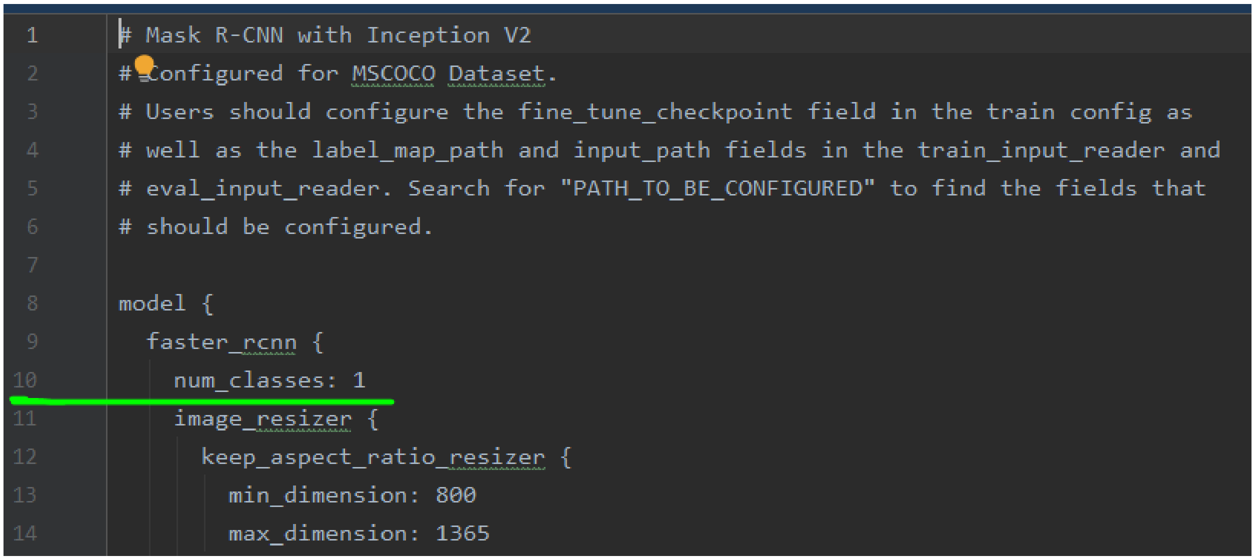


This PC › OS (C:) › Users › Asus-2020 › PycharmProjects › Maskrcnn_TF › models › research › mask_rcnn_training

Name	Date modified	Type	Size
labelmapdog	6/13/2020 8:56 PM	PBTXT File	1 KB
mask_rcnn_inception_v2_coco	12/2/2019 11:50 PM	CONFIG File	5 KB

Now open the config file and we need to do some changes on that file

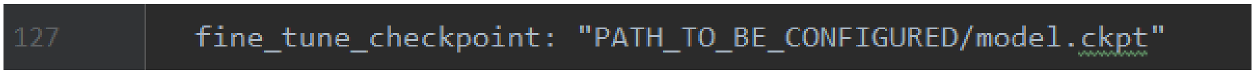
Changes 1:



```
1   # Mask R-CNN with Inception V2
2   # Configured for MSCOCO Dataset.
3   # Users should configure the fine_tune_checkpoint field in the train config as
4   # well as the label_map_path and input_path fields in the train_input_reader and
5   # eval_input_reader. Search for "PATH_TO_BE_CONFIGURED" to find the fields that
6   # should be configured.
7
8   model {
9     faster_rcnn {
10      num_classes: 1
11      image_resizer {
12        keep_aspect_ratio_resizer {
13          min_dimension: 800
14          max_dimension: 1365
```

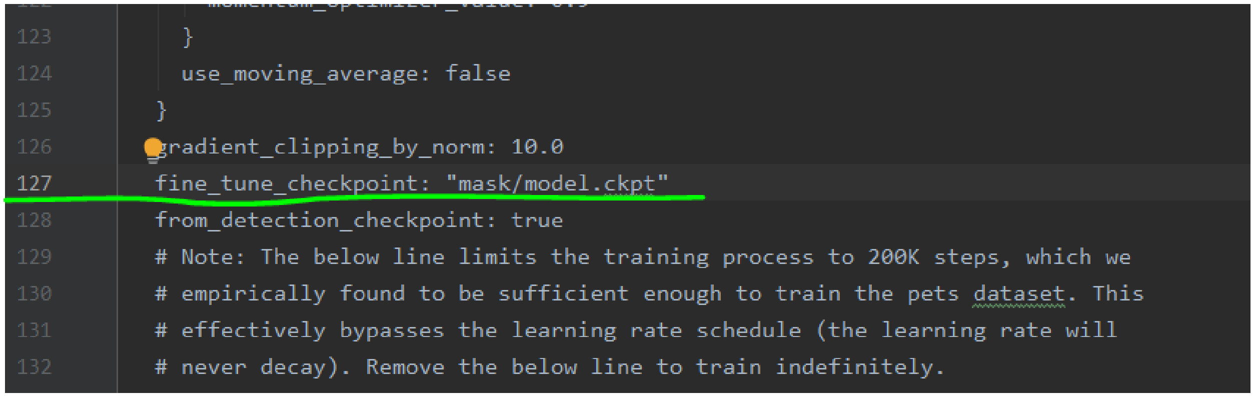
Changes 2:

In line no 127



```
127     fine_tune_checkpoint: "PATH_TO_BE_CONFIGURED/model.ckpt"
```

Change the path



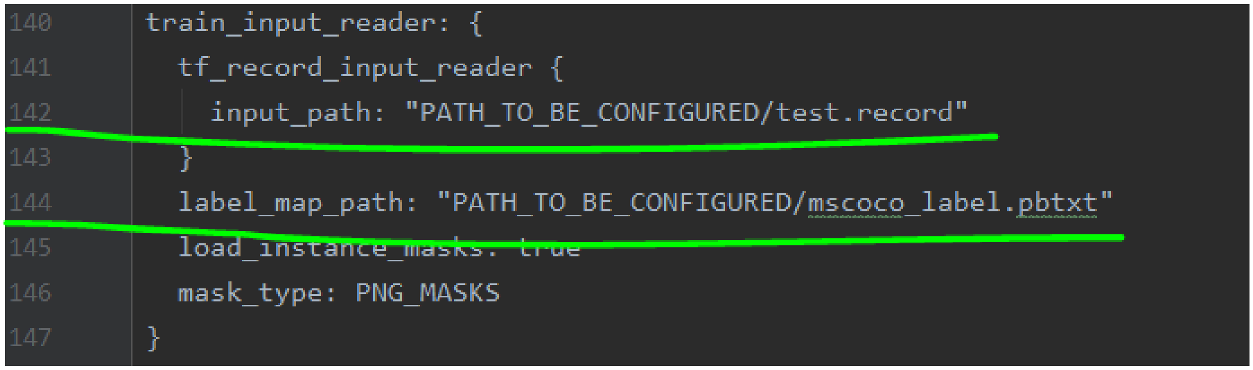
```
122         momentum_optimizer_value: 0.9
123       }
124       use_moving_average: false
125     }
126     gradient_clipping_by_norm: 10.0
127     fine_tune_checkpoint: "mask/model.ckpt"
128     from_detection_checkpoint: true
129     # Note: The below line limits the training process to 200K steps, which we
130     # empirically found to be sufficient enough to train the pets dataset. This
131     # effectively bypasses the learning rate schedule (the learning rate will
132     # never decay). Remove the below line to train indefinitely.
```

Changes 3:

Train and test record

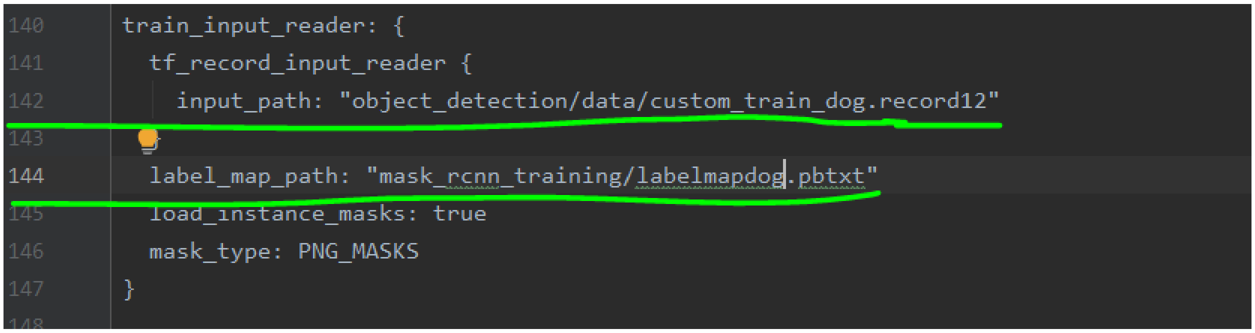
Line no : 142, 144, 158, 160

Train:



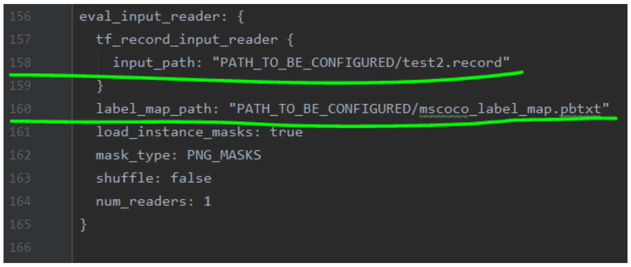
```
train_input_reader: {
  tf_record_input_reader {
    input_path: "PATH_TO_BE_CONFIGURED/test.record"
  }
  label_map_path: "PATH_TO_BE_CONFIGURED/mscoco_label.pbtxt"
  load_instance_masks: true
  mask_type: PNG_MASKS
}
```

Change the location of the train record and labelmap path



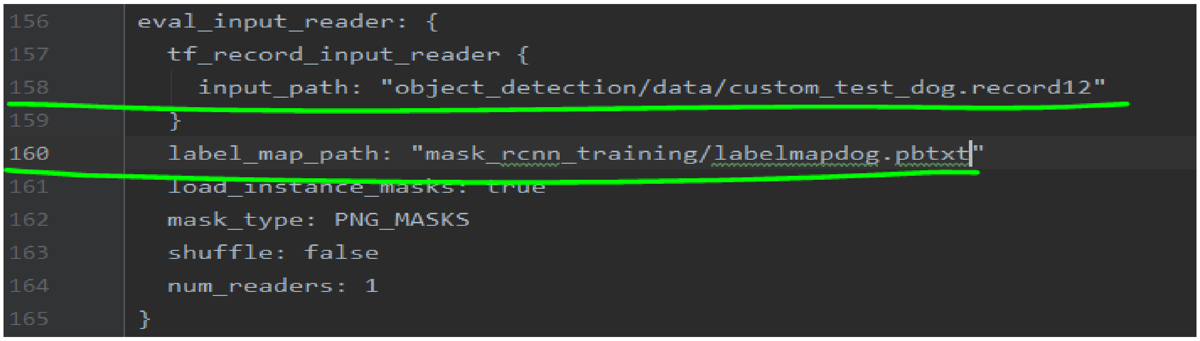
```
train_input_reader: {
  tf_record_input_reader {
    input_path: "object_detection/data/custom_train_dog.record12"
  }
  label_map_path: "mask_rcnn_training/labelmapdog.pbtxt"
  load_instance_masks: true
  mask_type: PNG_MASKS
}
```

Test:



```
eval_input_reader: {
  tf_record_input_reader {
    input_path: "PATH_TO_BE_CONFIGURED/test2.record"
  }
  label_map_path: "PATH_TO_BE_CONFIGURED/mscoco_label_map.pbtxt"
  load_instance_masks: true
  mask_type: PNG_MASKS
  shuffle: false
  num_readers: 1
}
```

Change the test and label map path

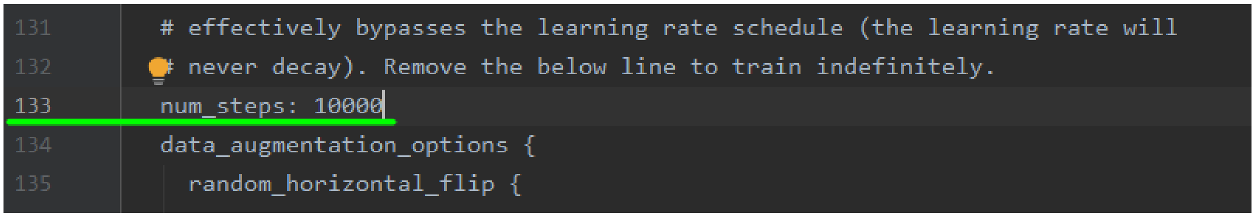


```
eval_input_reader: {
  tf_record_input_reader {
    input_path: "object_detection/data/custom_test_dog.record12"
  }
  label_map_path: "mask_rcnn_training/labelmapdog.pbtxt"
  load_instance_masks: true
  mask_type: PNG_MASKS
  shuffle: false
  num_readers: 1
}
```

Changes 4:

Line no : 133

Number of steps : I have changed the value to 10000 from 100



```
131     # effectively bypasses the learning rate schedule (the learning rate will
132     # never decay). Remove the below line to train indefinitely.
133     num_steps: 10000
134     data_augmentation_options {
135       random_horizontal_flip {
```

Start the training:

Training command:

open the terminal window and navigate to the research folder.

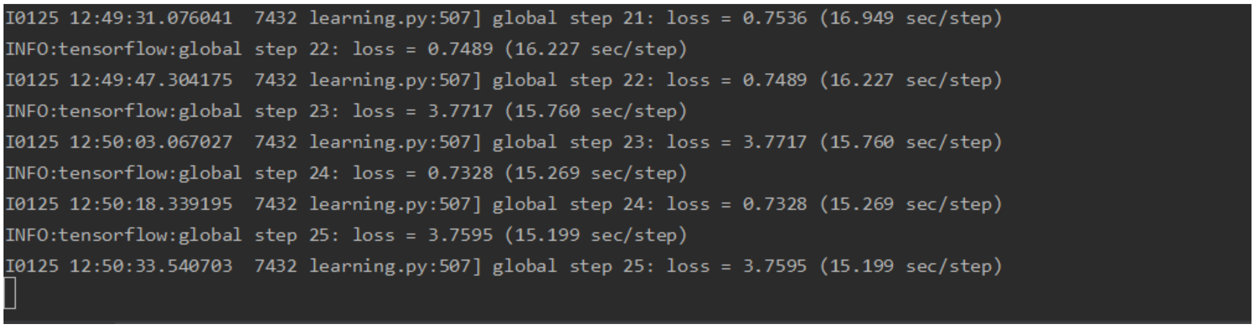
Default command:

python train.py --logtostderr --train_dir=training/ --pipeline_config_path=training/YOUR_MODEL.config

I have done some changes on the above command:

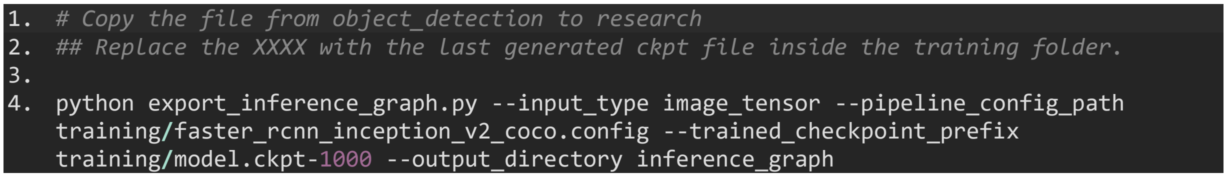
python train.py --logtostderr --train_dir=mask_rcnn_training/ --pipeline_config_path=mask_rcnn_training/mask_rcnn_inception_v2_coco.config

Now the training is going on



```
I0125 12:49:31.076041  7432 learning.py:507] global step 21: loss = 0.7536 (16.949 sec/step)
INFO:tensorflow:global step 22: loss = 0.7489 (16.227 sec/step)
I0125 12:49:47.304175  7432 learning.py:507] global step 22: loss = 0.7489 (16.227 sec/step)
INFO:tensorflow:global step 23: loss = 3.7717 (15.760 sec/step)
I0125 12:50:03.067027  7432 learning.py:507] global step 23: loss = 3.7717 (15.760 sec/step)
INFO:tensorflow:global step 24: loss = 0.7328 (15.269 sec/step)
I0125 12:50:18.339195  7432 learning.py:507] global step 24: loss = 0.7328 (15.269 sec/step)
INFO:tensorflow:global step 25: loss = 3.7595 (15.199 sec/step)
I0125 12:50:33.540703  7432 learning.py:507] global step 25: loss = 3.7595 (15.199 sec/step)
```

Consider now the training has completed,



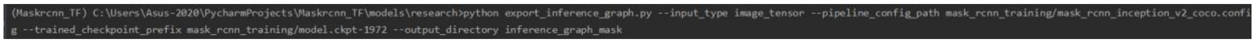
```
# Copy the file from object_detection to research
## Replace the XXXX with the last generated ckpt file inside the training folder.

python export_inference_graph.py --input_type image_tensor --pipeline_config_path
training/faster_rcnn_inception_v2_coco.config --trained_checkpoint_prefix
training/model.ckpt-1000 --output_directory inference_graph
```

Command:

python export_inference_graph.py --input_type image_tensor --pipeline_config_path mask_rcnn_training/mask_rcnn_inception_v2_coco.config --trained_checkpoint_prefix mask_rcnn_training/model.ckpt-1000 --output_directory inference_graph

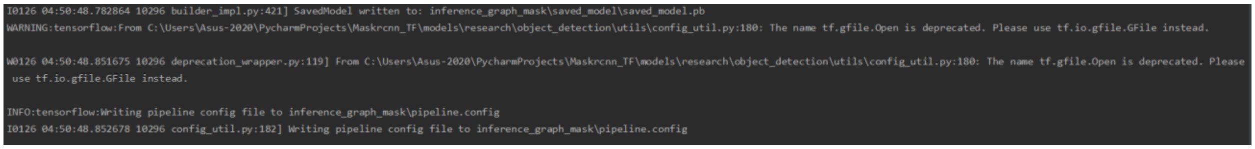
python export_inference_graph.py --input_type image_tensor --pipeline_config_path mask_rcnn_training/mask_rcnn_inception_v2_coco.confi

g --trained_checkpoint_prefix mask_rcnn_training/model.ckpt-1972 --output_directory inference_graph_mask

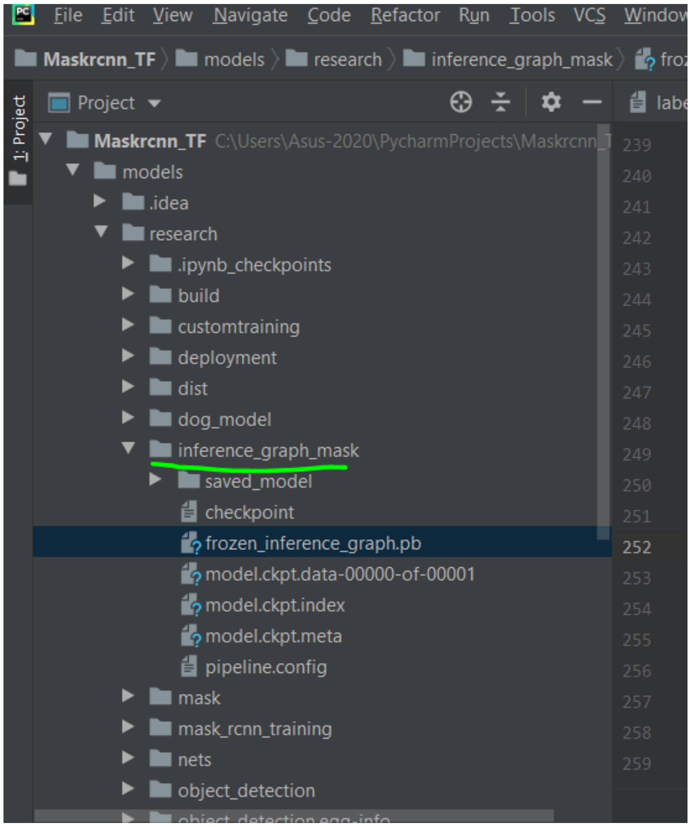


```
(Maskrcnn_TF) C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF\models\research>python export_inference_graph.py --input_type image_tensor --pipeline_config_path mask_rcnn_training/mask_rcnn_inception_v2_coco.confi
g --trained_checkpoint_prefix mask_rcnn_training/model.ckpt-1972 --output_directory inference_graph_mask
```

Completed successfully



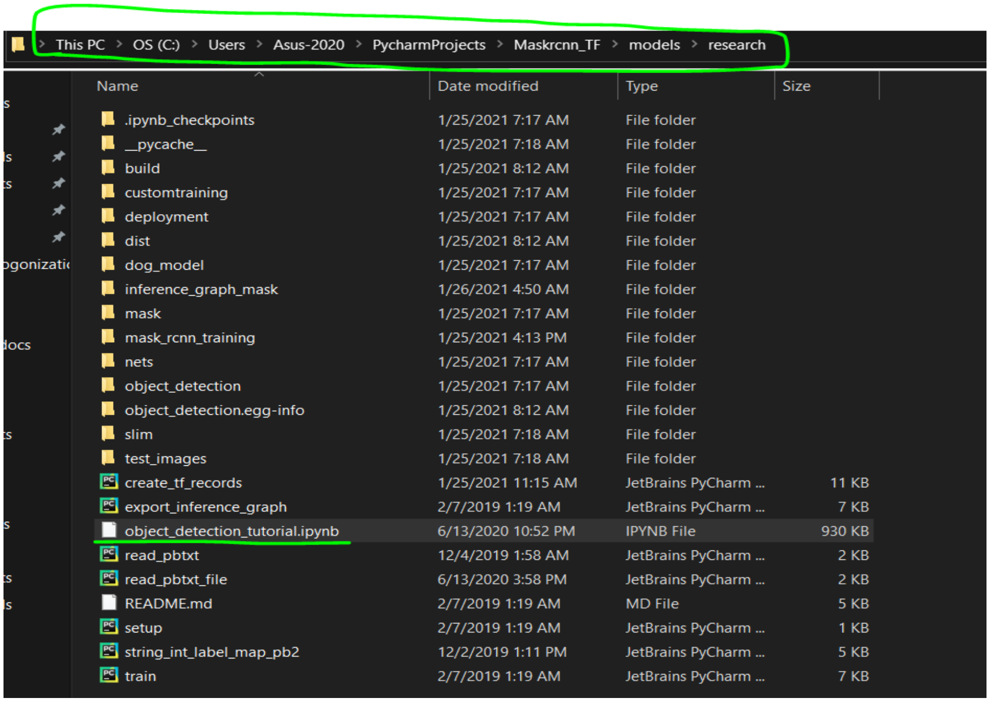
```
I0126 04:50:48.782864 10296 builder_impl.py:421] SavedModel written to: inference_graph_mask\saved_model\saved_model.pb
WARNING:tensorflow:From C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF\models\research\object_detection\utils\config_util.py:180: The name tf.gfile.Open is deprecated. Please use tf.io.gfile.GFile instead.

W0126 04:50:48.851675 10296 deprecation_wrapper.py:119] From C:\Users\Asus-2020\PycharmProjects\Maskrcnn_TF\models\research\object_detection\utils\config_util.py:180: The name tf.gfile.Open is deprecated. Please
 use tf.io.gfile.GFile instead.

INFO:tensorflow:Writing pipeline config file to inference_graph_mask\pipeline.config
I0126 04:50:48.852678 10296 config_util.py:182] Writing pipeline config file to inference_graph_mask\pipeline.config
```


File Edit View Navigate Code Refactor Run Tools VCS Windo
Maskrcnn_TF 〉 models 〉 research 〉 inference_graph_mask 〉 froz
Project
1: Project
labe

```
▼ Maskrcnn_TF  C:\Users\Asus-2020\PycharmProjects\Maskrcnn_T
  ▼ models
    ▶ .idea
    ▼ research
      ▶ .ipynb_checkpoints
      ▶ build
      ▶ customtraining
      ▶ deployment
      ▶ dist
      ▶ dog_model
      ▼ inference_graph_mask
        ▶ saved_model
          checkpoint
          frozen_inference_graph.pb
          model.ckpt.data-00000-of-00001
          model.ckpt.index
          model.ckpt.meta
          pipeline.config
      ▶ mask
      ▶ mask_rcnn_training
      ▶ nets
      ▶ object_detection
      ▶ object_detection.egg-info
```

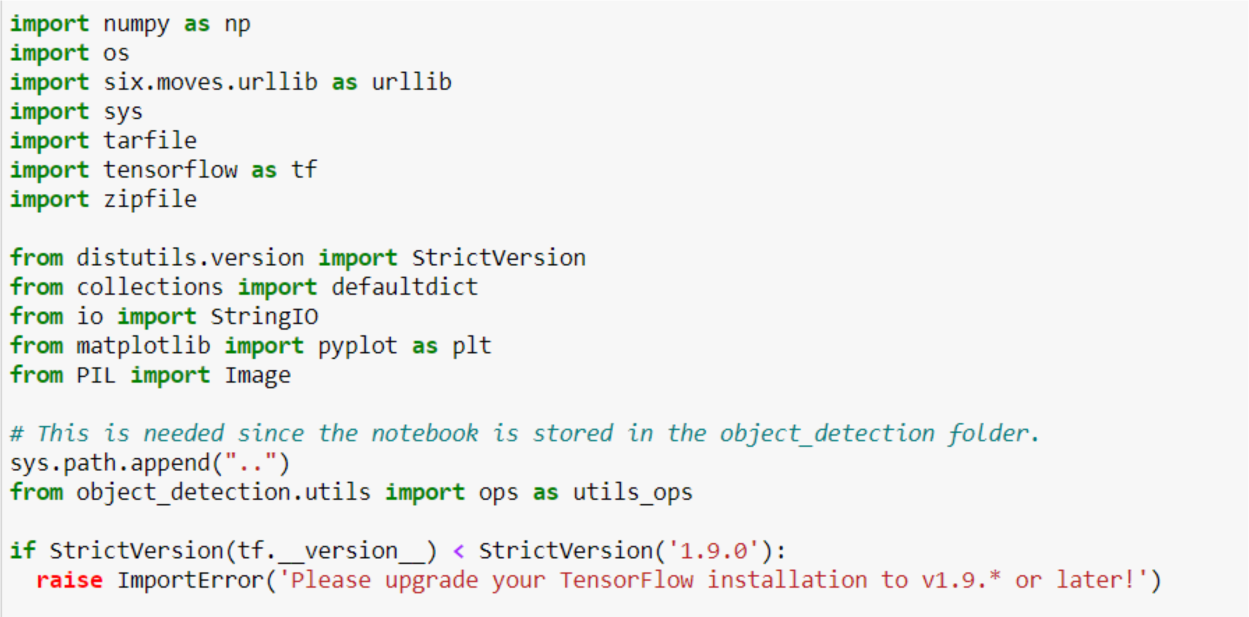
Now the .pb model is ready and we do the prediction.

Prediction:

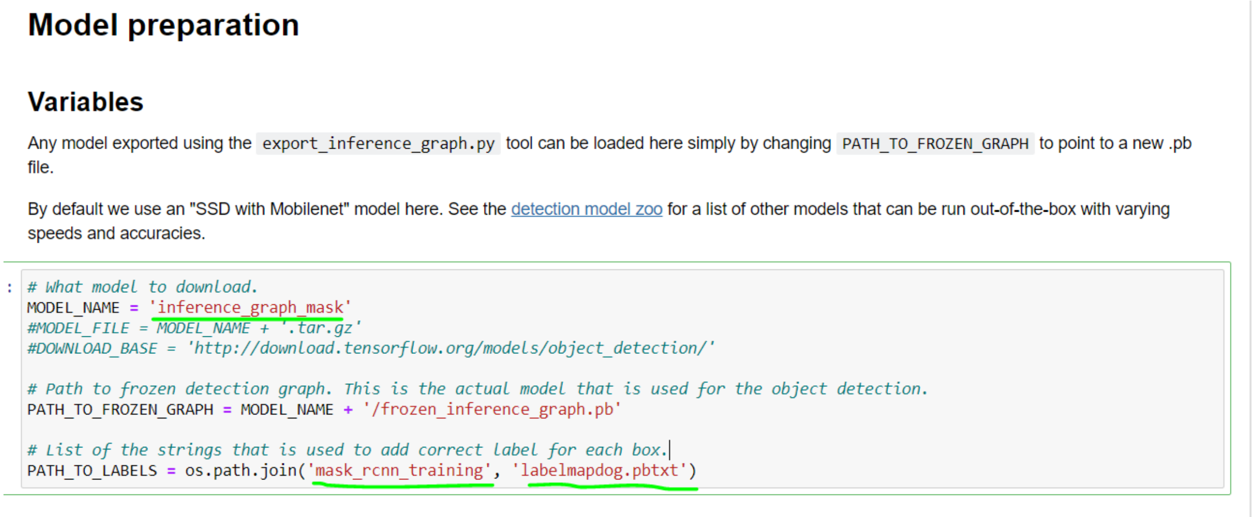

This PC › OS (C:) › Users › Asus-2020 › PycharmProjects › Maskrcnn_TF › models › research

Name	Date modified	Type	Size
.ipynb_checkpoints	1/25/2021 7:17 AM	File folder	
__pycache__	1/25/2021 7:18 AM	File folder	
build	1/25/2021 8:12 AM	File folder	
customtraining	1/25/2021 7:17 AM	File folder	
deployment	1/25/2021 7:17 AM	File folder	
dist	1/25/2021 8:12 AM	File folder	
dog_model	1/25/2021 7:17 AM	File folder	
inference_graph_mask	1/26/2021 4:50 AM	File folder	
mask	1/25/2021 7:17 AM	File folder	
mask_rcnn_training	1/25/2021 4:13 PM	File folder	
nets	1/25/2021 7:17 AM	File folder	
object_detection	1/25/2021 7:17 AM	File folder	
object_detection.egg-info	1/25/2021 8:12 AM	File folder	
slim	1/25/2021 7:18 AM	File folder	
test_images	1/25/2021 7:18 AM	File folder	
create_tf_records	1/25/2021 11:15 AM	JetBrains PyCharm ...	11 KB
export_inference_graph	2/7/2019 1:19 AM	JetBrains PyCharm ...	7 KB
object_detection_tutorial.ipynb	6/13/2020 10:52 PM	IPYNB File	930 KB
read_pbtxt	12/4/2019 1:58 AM	JetBrains PyCharm ...	2 KB
read_pbtxt_file	6/13/2020 3:58 PM	JetBrains PyCharm ...	2 KB
README.md	2/7/2019 1:19 AM	MD File	5 KB
setup	2/7/2019 1:19 AM	JetBrains PyCharm ...	1 KB
string_int_label_map_pb2	12/2/2019 1:11 PM	JetBrains PyCharm ...	5 KB
train	2/7/2019 1:19 AM	JetBrains PyCharm ...	7 KB

Open the file



```python
import numpy as np
import os
import six.moves.urllib as urllib
import sys
import tarfile
import tensorflow as tf
import zipfile

from distutils.version import StrictVersion
from collections import defaultdict
from io import StringIO
from matplotlib import pyplot as plt
from PIL import Image

# This is needed since the notebook is stored in the object_detection folder.
sys.path.append("..")
from object_detection.utils import ops as utils_ops

if StrictVersion(tf.__version__) < StrictVersion('1.9.0'):
  raise ImportError('Please upgrade your TensorFlow installation to v1.9.* or later!')
```



Model preparation

Variables

Any model exported using the `export_inference_graph.py` tool can be loaded here simply by changing `PATH_TO_FROZEN_GRAPH` to point to a new .pb file.

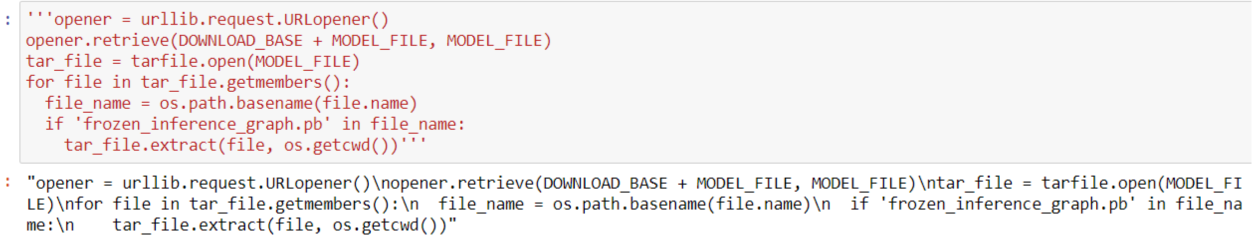
By default we use an "SSD with Mobilenet" model here. See the detection model zoo for a list of other models that can be run out-of-the-box with varying speeds and accuracies.

```python
# What model to download.
MODEL_NAME = 'inference_graph_mask'
#MODEL_FILE = MODEL_NAME + '.tar.gz'
#DOWNLOAD_BASE = 'http://download.tensorflow.org/models/object_detection/'

# Path to frozen detection graph. This is the actual model that is used for the object detection.
PATH_TO_FROZEN_GRAPH = MODEL_NAME + '/frozen_inference_graph.pb'

# List of the strings that is used to add correct label for each box.
PATH_TO_LABELS = os.path.join('mask_rcnn_training', 'labelmapdog.pbtxt')
```

In this above section do some changes accordingly to your folder and file location.

Comment this section

Download Model

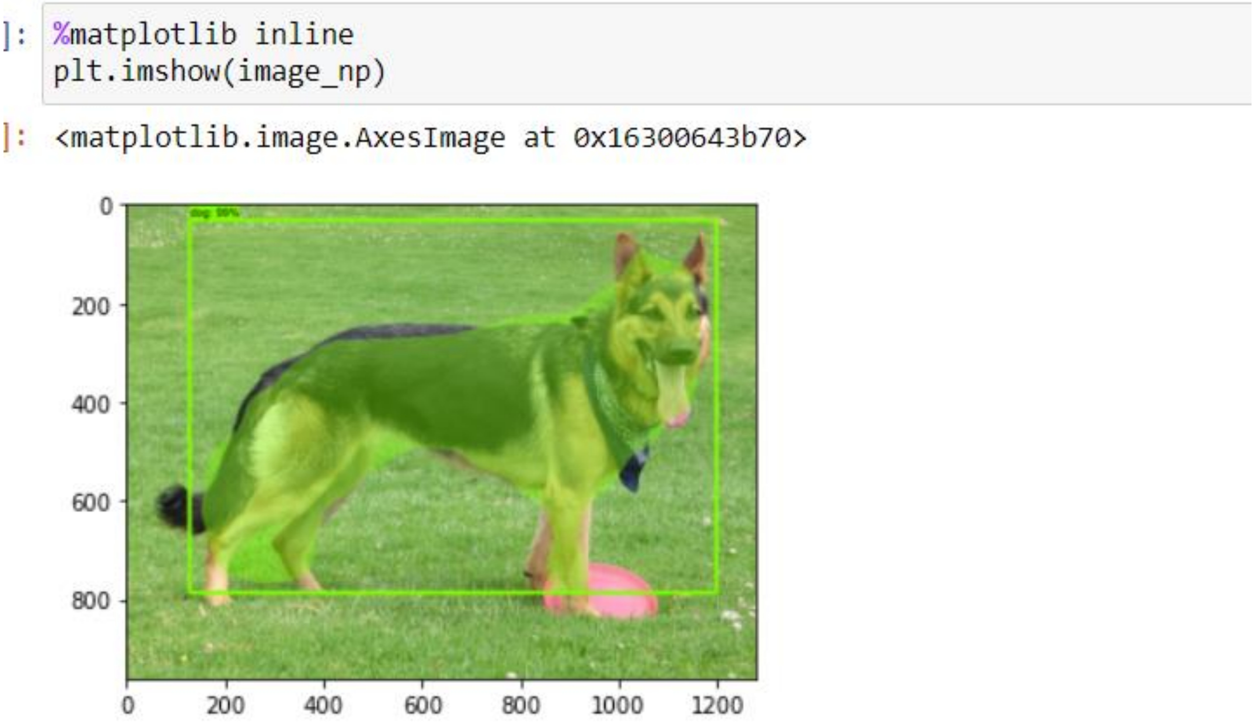


```
'''opener = urllib.request.URLopener()
opener.retrieve(DOWNLOAD_BASE + MODEL_FILE, MODEL_FILE)
tar_file = tarfile.open(MODEL_FILE)
for file in tar_file.getmembers():
  file_name = os.path.basename(file.name)
  if 'frozen_inference_graph.pb' in file_name:
    tar_file.extract(file, os.getcwd())'''
```

```
"opener = urllib.request.URLopener()\nopener.retrieve(DOWNLOAD_BASE + MODEL_FILE, MODEL_FILE)\ntar_file = tarfile.open(MODEL_FI
LE)\nfor file in tar_file.getmembers():\n  file_name = os.path.basename(file.name)\n  if 'frozen_inference_graph.pb' in file_na
me:\n    tar_file.extract(file, os.getcwd())"
```

Run all the rest of cells inside the notebook file

Final prediction looks like below.



```
%matplotlib inline
plt.imshow(image_np)
```

```
<matplotlib.image.AxesImage at 0x16300643b70>
```