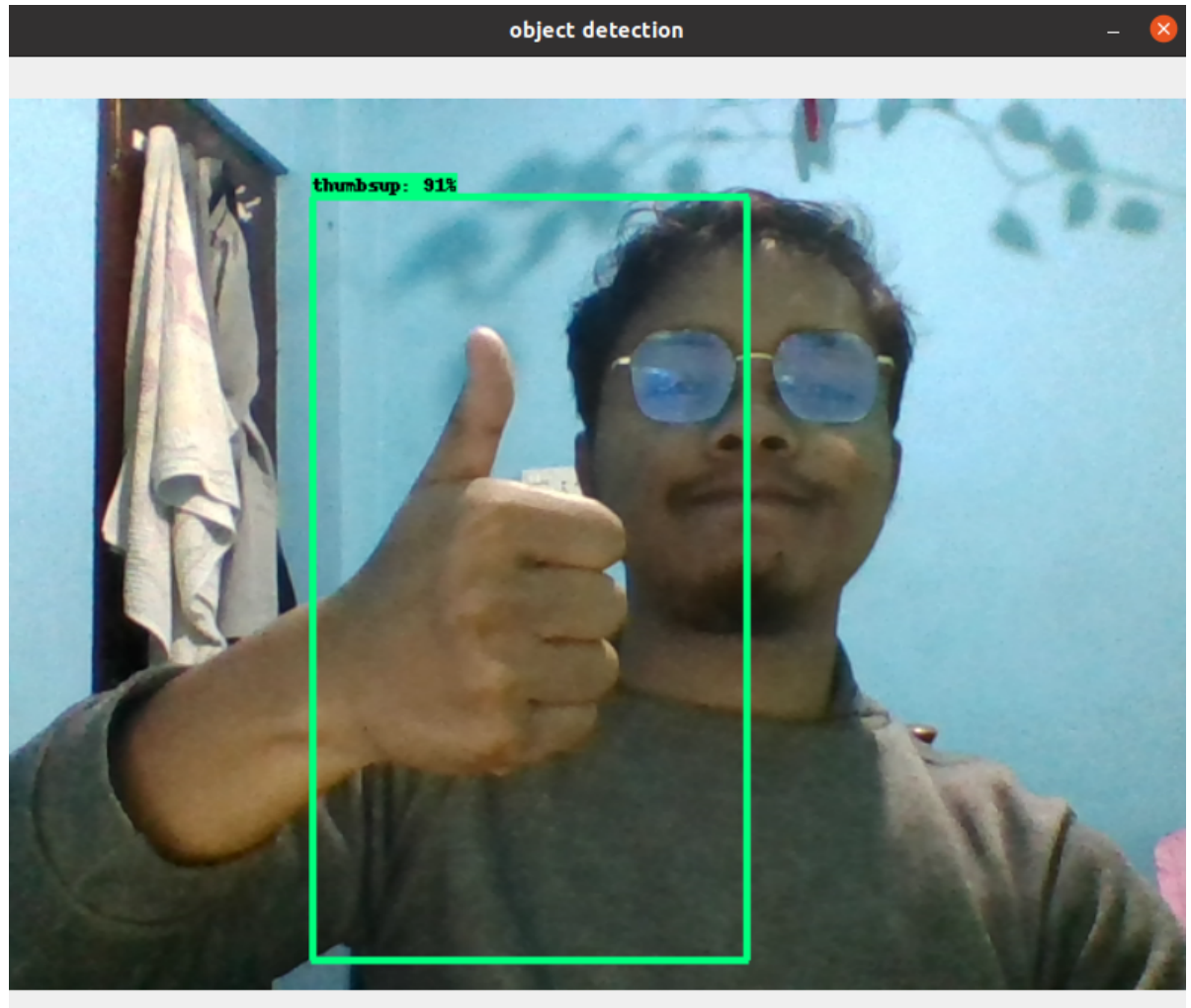


This is the 2nd ipynb file. This file has done all the necessary installation needed for tensorflow object detection, and trained & evaluated the model.

Kindly go to the first file: **imagesdata_collection.ipynb** file that is in same repo to collect the real-time imagedata to train the model.



```
In [1]: import os
```

```
In [2]: custom_model_name='ssd_mobnet'  
pretrained_model_name = 'ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8'  
pretrained_model_url = 'http://download.tensorflow.org/models/object_detection'  
tfrecord_script_name = 'generate_tfrecord.py'  
label_map_name = 'label_map.pbtxt'
```

```
In [3]: paths = {
    'workspace_path': os.path.join('Tensorflow', 'workspace'),
    'api_model_path': os.path.join('Tensorflow', 'models'),
    'protoc_path': os.path.join('Tensorflow', 'protoc'),
    'script_path': os.path.join('Tensorflow', 'script'),
    'image_path': os.path.join('Tensorflow', 'workspace', 'images'),
    'pretrained_model_path': os.path.join('Tensorflow', 'workspace', 'pretrained_model'),
    'model_path': os.path.join('Tensorflow', 'workspace', 'models'),
    'annotations_path': os.path.join('Tensorflow', 'workspace', 'annotations'),
    'checkpoint_path': os.path.join('Tensorflow', 'workspace', 'models', 'checkpoint'),
    'output_path': os.path.join('Tensorflow', 'workspace', 'models', 'custom_ops'),
    'tfjs_path': os.path.join('Tensorflow', 'workspace', 'models', 'custom_ops', 'tfjs'),
    'tflite_path': os.path.join('Tensorflow', 'workspace', 'models', 'custom_ops', 'tflite')
}
```

```
In [4]: for key, values in paths.items():
    print(f'{key}:{values}\n')
```

workspace_path:Tensorflow/workspace

api_model_path:Tensorflow/models

protoc_path:Tensorflow/protoc

script_path:Tensorflow/script

image_path:Tensorflow/workspace/images

pretrained_model_path:Tensorflow/workspace/pretrained_model

model_path:Tensorflow/workspace/models

annotations_path:Tensorflow/workspace/annotations

checkpoint_path:Tensorflow/workspace/models/ssd_mobnet

output_path:Tensorflow/workspace/models/ssd_mobnet/export

tfjs_path:Tensorflow/workspace/models/ssd_mobnet/tfjsexport

tflite_path:Tensorflow/workspace/models/ssd_mobnet/tfliteexport

```
In [5]: files = {
    'labelmap_file': os.path.join(paths['annotations_path'], 'label_map.pbtxt'),
    'tfrecord_script': os.path.join(paths['script_path'], 'tfrecord_script.py'),
    'pipeline_config': os.path.join(paths['checkpoint_path'], 'pipeline.config')
}
```

```
In [7]: for key, values in files.items():
        print(f'{key}:{values}\n')
```

labelmap_file:Tensorflow/workspace/annotations/label_map.pbtxt

tfrecord_script:Tensorflow/script/generate_tfrecord.py

pipeline_config:Tensorflow/workspace/models/ssd_mobnet/pipeline.config

```
In [19]: for value in paths.values():
        if not os.path.exists(value):
            if os.name=='posix':
                !mkdir -p {value}
            if os.name == 'nt':
                !mkdir {value}
```



labelimg



models



protoc



script



workspace

TF object detection model from tf model garden

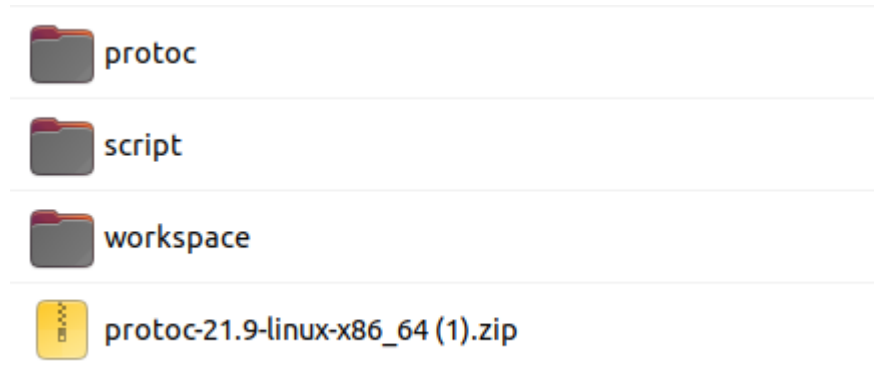
<https://github.com/tensorflow/models> (<https://github.com/tensorflow/models>)

```
In [15]: if not os.path.exists(os.path.join(paths['api_model_path'], 'research', 'c
        !git clone git@github.com:tensorflow/models.git {paths['api_model_pa
```

```
Cloning into 'Tensorflow/models'...
remote: Enumerating objects: 78956, done.
remote: Counting objects: 100% (297/297), done.
remote: Compressing objects: 100% (161/161), done.
remote: Total 78956 (delta 143), reused 276 (delta 135), pack-reused 7
8659
Receiving objects: 100% (78956/78956), 593.86 MiB | 2.92 MiB/s, done.
Resolving deltas: 100% (56124/56124), done.
Updating files: 100% (3243/3243), done.
```

Downloading and compiling the Protocol Buffer, and installing all the necessary dependencies for tf object detection.

<https://github.com/protocolbuffers/protobuf/releases>
<https://github.com/protocolbuffers/protobuf/releases>



Run all the belows' command in sequence. After running **python3 -m pip install .** all the dependencies will start to download. If any error occurs, rerun the command again.

```
(hgd) shreejan@shreejan-inspiron-5559:~/Documents/hand_gesture_detection/Tensorflow$ ls
labelImg  models  protoc  'protoc-21.9-linux-x86_64 (1).zip'  script  workspace
(hgd) shreejan@shreejan-inspiron-5559:~/Documents/hand_gesture_detection/Tensorflow$ cd models/research/
(hgd) shreejan@shreejan-inspiron-5559:~/Documents/hand_gesture_detection/Tensorflow/models/research$ protoc object_detection/protos/*.proto --python_out=.
(hgd) shreejan@shreejan-inspiron-5559:~/Documents/hand_gesture_detection/Tensorflow/models/research$ cp object_detection/packages/tf2/setup.py .
(hgd) shreejan@shreejan-inspiron-5559:~/Documents/hand_gesture_detection/Tensorflow/models/research$ python3 -m pip3 install .
/home/shreejan/Documents/hand_gesture_detection/hgd/bin/python3: No module named pip3
(hgd) shreejan@shreejan-inspiron-5559:~/Documents/hand_gesture_detection/Tensorflow/models/research$ python3 -m pip install .
Processing /home/shreejan/Documents/hand_gesture_detection/Tensorflow/models/research
  Preparing metadata (setup.py) ... done
Collecting Cython
  Using cached Cython-0.29.32-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.manylinux_2_24_x86_64.whl (1.9 MB)
Collecting apache-beam
  Using cached apache_beam-2.43.0-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (14.5 MB)
Collecting avro-python3
  Using cached avro_python3-1.10.2-py3-none-any.whl
Collecting contextlib2
  Using cached contextlib2-21.6.0-py2.py3-none-any.whl (13 kB)
Requirement already satisfied: keras in /home/shreejan/Documents/hand_gesture_detection/hgd/lib/python3.8/site-packages (from object-detection==0.1) (2.11.0)
Collecting lvis
  Using cached lvis-0.5.3-py3-none-any.whl (14 kB)
Requirement already satisfied: lxml in /home/shreejan/Documents/hand_gesture_detection/hgd/lib/python3.8/site-packages (from object-detection==0.1) (4.9.1)
Collecting matplotlib
```

After this, run below command. This command's output will say whether our all of the installations are done perfectly or not.

```
(hgd) shreejan@shreejan-inspiron-5559:~/Documents/hand_gesture_detection/Tensorflow/models/research$ python3 object_detection/builders/model_builder_tf2_test.py
2022-11-19 14:51:19.612853: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-11-19 14:51:19.786439: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libcudart.so.11.0'; dlopen: libcudart.so.11.0: cannot open shared object file: No such file or directory
```

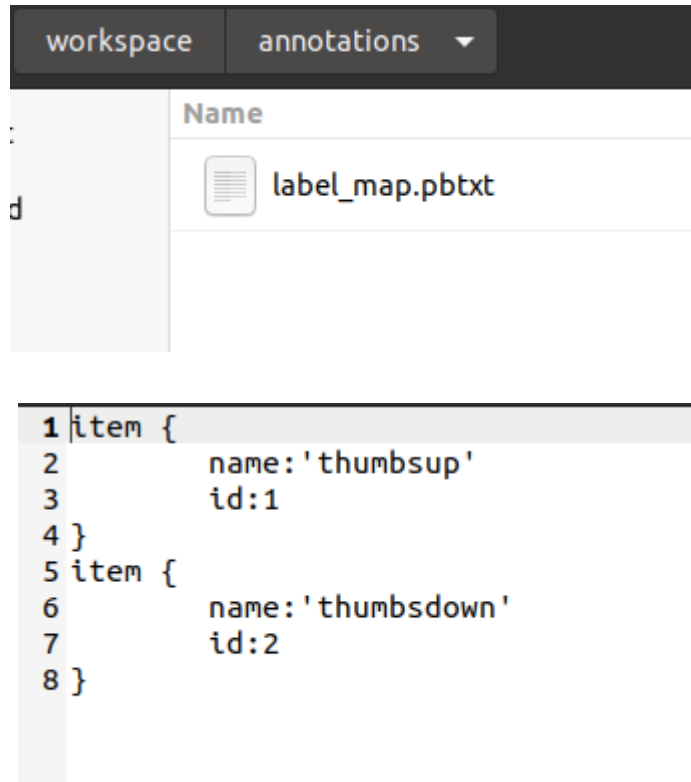
If you see OK on your terminal like below then you are go to good ahead.

```
[ OK ] ModelBuilderTF2Test.test_unknown_faster_rcnn_feature_extractor
[ RUN ] ModelBuilderTF2Test.test_unknown_meta_architecture
INFO:tensorflow:time(__main__.ModelBuilderTF2Test.test_unknown_meta_architecture): 0.0s
I1119 14:52:09.361696 I40710182348608 test_util.py:2460] time(__main__.ModelBuilderTF2Test.test_unknown_meta_architecture): 0.0s
[ OK ] ModelBuilderTF2Test.test_unknown_meta_architecture
[ RUN ] ModelBuilderTF2Test.test_unknown_ssd_feature_extractor
INFO:tensorflow:time(__main__.ModelBuilderTF2Test.test_unknown_ssd_feature_extractor): 0.0s
I1119 14:52:09.363991 I40710182348608 test_util.py:2460] time(__main__.ModelBuilderTF2Test.test_unknown_ssd_feature_extractor): 0.0s
[ OK ] ModelBuilderTF2Test.test_unknown_ssd_feature_extractor
-----
Ran 24 tests in 41.050s
OK (skipped=1)
```

Creating the label map

```
In [8]: labels = [{'name':'thumbsup', 'id':1}, {'name':'thumbsdown', 'id':2}]

with open(files['labelmap_file'], 'w') as f:
    for label in labels:
        f.write('item { \n')
        f.write('\tname:\'{}\'\n'.format(label['name']))
        f.write('\tid:{}\n'.format(label['id']))
        f.write('}\n')
```






Creating the TFrecords of our imagedata.

```
In [22]: if not os.path.exists(files['tfrecord_script']):
         !git clone https://github.com/nicknochnack/GenerateTFRecord {paths['
```




```
Cloning into 'Tensorflow/script'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 1 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 2.67 KiB | 2.67 MiB/s, done.
```





```
In [23]: !python {files['tfrecord_script']} -x {os.path.join(paths['image_path'],
!python {files['tfrecord_script']} -x {os.path.join(paths['image_path'],
```

```
2022-11-19 15:15:31.562563: E tensorflow/stream_executor/cuda/cuda_bla
s.cc:2981] Unable to register cuBLAS factory: Attempting to register f
actory for plugin cuBLAS when one has already been registered
Successfully created the TFRecord file: Tensorflow/workspace/annotatio
ns/train.record
2022-11-19 15:15:34.875321: E tensorflow/stream_executor/cuda/cuda_bla
s.cc:2981] Unable to register cuBLAS factory: Attempting to register f
actory for plugin cuBLAS when one has already been registered
Successfully created the TFRecord file: Tensorflow/workspace/annotatio
ns/validation.record
```

workspace annotations	
Name	
	label_map.pbtxt
	train.record
	validation.record

Copy the pipeline.config file from pretrained_model to models

hand_gesture_detection Tensorflow workspace pretrained_model ssd_mobil...co17_tpu-8					
Name					
	checkpoint				
	saved_model				
	pipeline.config				

Home Documents hand_gesture_detection Tensorflow workspace models ssd_mobnet					
Name					
	export				
	tfjsexport				
	tfliexport				
	pipeline.config				

IMPORTANT STEP!

Below code will download the selected pretrained object detection model and uncompress it using linux's tar command. Now, we are using `ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8`.

```
In [20]: if os.name == 'posix': #for linux
          !wget {pretrained_model_url}
          !mv {pretrained_model_name+'.tar.gz'} {paths['pretrained_model_path']}
          !cd {paths['pretrained_model_path']} && tar -zxvf {pretrained_model_name+'.tar.gz'}

          if os.name == 'nt': #for windows
            wget.download(pretrained_model_url)
            !move {pretrained_model_name+'.tar.gz'} {paths['pretrained_model_path']}
            !cd {paths['pretrained_model_path']} && tar -zxvf {pretrained_model_name+'.tar.gz'}
```

--2022-11-19 15:06:53-- http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8.tar.gz (http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8.tar.gz)
Resolving download.tensorflow.org (download.tensorflow.org)... 2404:6800:4009:824::2010, 142.250.183.144
Connecting to download.tensorflow.org (download.tensorflow.org)|2404:6800:4009:824::2010|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 20515344 (20M) [application/x-tar]
Saving to: 'ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8.tar.gz'

ssd_mobilenet_v2_fp 100%[=====>] 19.56M 2.75MB/s in 6.9s

2022-11-19 15:07:00 (2.82 MB/s) - 'ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8.tar.gz' saved [20515344/20515344]

ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/checkpoint/
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/checkpoint/ckpt-0.data-0000-of-00001
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/checkpoint/checkpoint
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/checkpoint/ckpt-0.index
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/pipeline.config
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/saved_model/
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/saved_model/saved_model.pb
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/saved_model/variables/
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/saved_model/variables/variables.data-00000-of-00001
ssd_mobilenet_v2_fpn-lite_320x320_coco17_tpu-8/saved_model/variables/variables.index

NOW,

Below, I have printed the model's original configuration. We have to adjust it according to our need. Kindly go to `Tensorflow/workspace/models/ssd_mobnet/pipeline_config` and open the file.


```
In [9]: from object_detection.utils import config_util
config = config_util.get_configs_from_pipeline_file(files['pipeline_conf'])
print(config)
```

```
2022-11-19 19:17:08.097231: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-11-19 19:17:08.605941: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libcudart.so.11.0'; dlerror: libcudart.so.11.0: cannot open shared object file: No such file or directory
2022-11-19 19:17:08.605977: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
2022-11-19 19:17:08.669693: E tensorflow/stream_executor/cuda/cuda_blas.cc:2981] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
2022-11-19 19:17:28.712058: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libnvinfer.so.7'; dlerror: libnvinfer.so.7: cannot open shared object file: No such file or directory
2022-11-19 19:17:28.712301: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libnvinfer_plugin.so.7'; dlerror: libnvinfer_plugin.so.7: cannot open shared object file: No such file or directory
2022-11-19 19:17:28.712331: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Cannot dlopen some TensorRT libraries. If you would like to use Nvidia GPU with TensorRT, please make sure the missing libraries mentioned above are installed properly.
```

```
{'model': ssd {
  num_classes: 2
  image_resizer {
    fixed_shape_resizer {
      height: 320
      width: 320
    }
  }
  feature_extractor {
    type: "ssd_mobilenet_v2_fpn_keras"
    depth_multiplier: 1.0
    min_depth: 16
    conv_hyperparams {
      regularizer {
        l2_regularizer {
          weight: 3.9999998989515007e-05
        }
      }
    }
    initializer {
      random_normal_initializer {
        mean: 0.0
        stddev: 0.009999999776482582
      }
    }
  }
}
```



```

    }
    activation: RELU_6
    batch_norm {
      decay: 0.996999979019165
      scale: true
      epsilon: 0.0010000000474974513
    }
  }
  use_depthwise: true
  override_base_feature_extractor_hyperparams: true
  fpn {
    min_level: 3
    max_level: 7
    additional_layer_depth: 128
  }
}
box_coder {
  faster_rcnn_box_coder {
    y_scale: 10.0
    x_scale: 10.0
    height_scale: 5.0
    width_scale: 5.0
  }
}
matcher {
  argmax_matcher {
    matched_threshold: 0.5
    unmatched_threshold: 0.5
    ignore_thresholds: false
    negatives_lower_than_unmatched: true
    force_match_for_each_row: true
    use_matmul_gather: true
  }
}
similarity_calculator {
  iou_similarity {
  }
}
}
box_predictor {
  weight_shared_convolutional_box_predictor {
    conv_hyperparams {
      regularizer {
        l2_regularizer {
          weight: 3.9999998989515007e-05
        }
      }
      initializer {
        random_normal_initializer {
          mean: 0.0
          stddev: 0.009999999776482582
        }
      }
    }
    activation: RELU_6
    batch_norm {
      decay: 0.996999979019165
      scale: true
      epsilon: 0.0010000000474974513
    }
  }
}

```

```

    }
  }
  depth: 128
  num_layers_before_predictor: 4
  kernel_size: 3
  class_prediction_bias_init: -4.599999904632568
  share_prediction_tower: true
  use_depthwise: true
}
}
anchor_generator {
  multiscale_anchor_generator {
    min_level: 3
    max_level: 7
    anchor_scale: 4.0
    aspect_ratios: 1.0
    aspect_ratios: 2.0
    aspect_ratios: 0.5
    scales_per_octave: 2
  }
}
post_processing {
  batch_non_max_suppression {
    score_threshold: 9.99999993922529e-09
    iou_threshold: 0.60000000238418579
    max_detections_per_class: 100
    max_total_detections: 100
    use_static_shapes: false
  }
  score_converter: SIGMOID
}
normalize_loss_by_num_matches: true
loss {
  localization_loss {
    weighted_smooth_l1 {
    }
  }
  classification_loss {
    weighted_sigmoid_focal {
      gamma: 2.0
      alpha: 0.25
    }
  }
  classification_weight: 1.0
  localization_weight: 1.0
}
encode_background_as_zeros: true
normalize_loc_loss_by_codesize: true
inplace_batchnorm_update: true
freeze_batchnorm: false
}
, 'train_config': batch_size: 4
data_augmentation_options {
  random_horizontal_flip {
  }
}
}
data_augmentation_options {

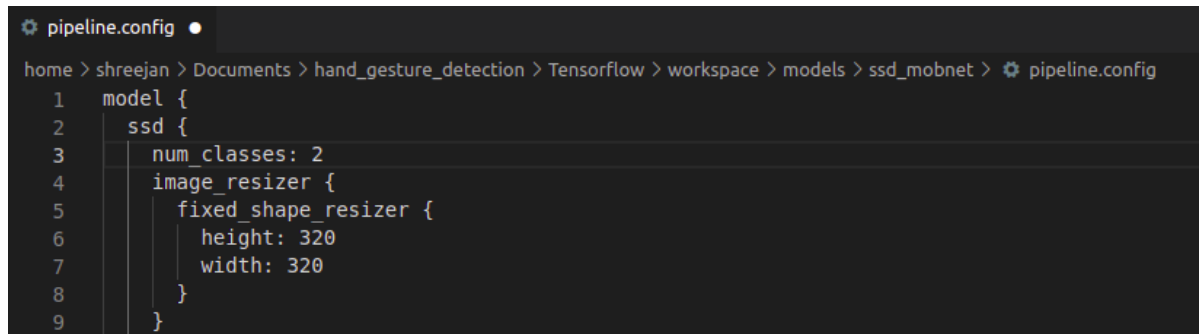
```

```

random_crop_image {
  min_object_covered: 0.0
  min_aspect_ratio: 0.75
  max_aspect_ratio: 3.0
  min_area: 0.75
  max_area: 1.0
  overlap_thresh: 0.0
}
}
sync_replicas: true
optimizer {
  momentum_optimizer {
    learning_rate {
      cosine_decay_learning_rate {
        learning_rate_base: 0.07999999821186066
        total_steps: 50000
        warmup_learning_rate: 0.026666000485420227
        warmup_steps: 1000
      }
    }
    momentum_optimizer_value: 0.8999999761581421
  }
  use_moving_average: false
}
fine_tune_checkpoint: "Tensorflow/workspace/pretrained_model/ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8/checkpoint/ckpt-0"
num_steps: 50000
startup_delay_steps: 0.0
replicas_to_aggregate: 8
max_number_of_boxes: 100
unpad_groundtruth_tensors: false
fine_tune_checkpoint_type: "detection"
fine_tune_checkpoint_version: V2
, 'train_input_config': label_map_path: "Tensorflow/workspace/annotations/label_map.pbtxt"
tf_record_input_reader {
  input_path: "Tensorflow/workspace/annotations/train.record"
}
, 'eval_config': metrics_set: "coco_detection_metrics"
use_moving_averages: false
, 'eval_input_configs': [label_map_path: "Tensorflow/workspace/annotations/label_map.pbtxt"
shuffle: false
num_epochs: 1
tf_record_input_reader {
  input_path: "Tensorflow/workspace/annotations/validation.record"
}
], 'eval_input_config': label_map_path: "Tensorflow/workspace/annotations/label_map.pbtxt"
shuffle: false
num_epochs: 1
tf_record_input_reader {
  input_path: "Tensorflow/workspace/annotations/validation.record"
}
}
}

```

After opening the pipeline.config file from where ever you want (in my case, I use Vscode), modify it according your custom model need. Since, I have two classes (thumbsup and thumbsdown), I have set the num_classes:2

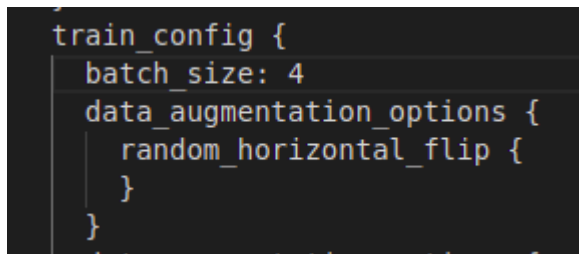


```

1  model {
2    ssd {
3      num_classes: 2
4      image_resizer {
5        fixed_shape_resizer {
6          height: 320
7          width: 320
8        }
9      }

```

For now, I set the batch_size:4 I



```

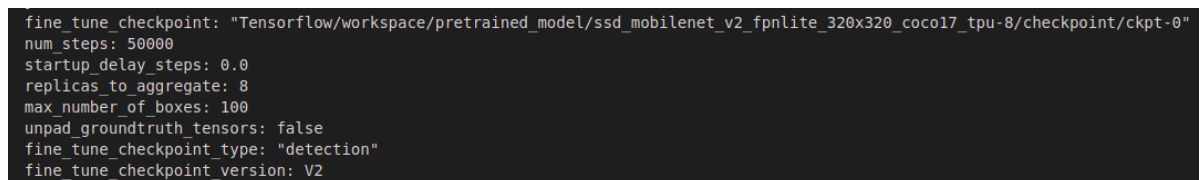
train_config {
  batch_size: 4
  data_augmentation_options {
    random_horizontal_flip {

```

Below's two screenshots have a couple of things to edit.

- 1) fine_tune_checkpoint
- 2) fine_tune_checkpoint_type (set this to detection since we are doing detection)
- 3) label_map_path
- 4) input_path

For 1,3,4 give the exact same path.



```

fine_tune_checkpoint: "Tensorflow/workspace/pretrained_model/ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8/checkpoint/ckpt-0"
num_steps: 50000
startup_delay_steps: 0.0
replicas_to_aggregate: 8
max_number_of_boxes: 100
unpad_groundtruth_tensors: false
fine_tune_checkpoint_type: "detection"
fine_tune_checkpoint_version: V2

```

```

train_input_reader {
  label_map_path: "Tensorflow/workspace/annotations/label_map.pbtxt"
  tf_record_input_reader {
    input_path: "Tensorflow/workspace/annotations/train.record"
  }
}
eval_config {
  metrics_set: "coco_detection_metrics"
  use_moving_averages: false
}
eval_input_reader {
  label_map_path: "Tensorflow/workspace/annotations/label_map.pbtxt"
  shuffle: false
  num_epochs: 1
  tf_record_input_reader {
    input_path: "Tensorflow/workspace/annotations/validation.record"
  }
}
}

```

Save the file and close it.

Training our model

```
In [10]: training_script = os.path.join(paths['api_model_path'], 'research', 'obj
```

```
In [11]: training_script
```

```
Out[11]: 'Tensorflow/models/research/object_detection/model_main_tf2.py'
```

```
In [32]: command = f"python3 {training_script} --model_dir={paths['checkpoint_pa
```

```
In [34]: print(command)
#copy and run it in your terminal.
```

```
python3 Tensorflow/models/research/object_detection/model_main_tf2.py
--model_dir=Tensorflow/workspace/models/ssd_mobnet --pipeline_config_p
ath=Tensorflow/workspace/models/ssd_mobnet/pipeline.config --num_train
_steps=2000
```

```

(hgd) shreejangshreejan-Inspiron-5559:~/Documents/hand_gesture_detection$ python3 Tensorflow/models/research/object_detection/model_main_tf2.py --model_dir=Tensorflow/
workspace/models/ssd_mobnet --pipeline_config_path=Tensorflow/workspace/models/ssd_mobnet/pipeline.config --num_train_steps=2000
2022-11-19 15:44:49.024178: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) t
o use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-11-19 15:44:49.259806: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libcudart.so.11.0'; dlopen: libcudart.so.1

```

```

I1119 16:27:29.059579 140139170592576 model_lib_v2.py:705] Step 1900 per-step time 1.219s
INFO:tensorflow:{'Loss/classification_loss': 0.092233285,
'Loss/localization_loss': 0.0135296,
'Loss/regularization_loss': 0.14477651,
'Loss/total_loss': 0.2505394,
'learning_rate': 0.07993342}
I1119 16:27:29.059859 140139170592576 model_lib_v2.py:708] {'Loss/classification_loss': 0.092233285,
'Loss/localization_loss': 0.0135296,
'Loss/regularization_loss': 0.14477651,
'Loss/total_loss': 0.2505394,
'learning_rate': 0.07993342}
INFO:tensorflow:Step 2000 per-step time 1.266s
I1119 16:29:35.640299 140139170592576 model_lib_v2.py:705] Step 2000 per-step time 1.266s
INFO:tensorflow:{'Loss/classification_loss': 0.062785655,
'Loss/localization_loss': 0.03801403,
'Loss/regularization_loss': 0.14399973,
'Loss/total_loss': 0.2447994,
'learning_rate': 0.07991781}
I1119 16:29:35.640596 140139170592576 model_lib_v2.py:708] {'Loss/classification_loss': 0.062785655,
'Loss/localization_loss': 0.03801403,
'Loss/regularization_loss': 0.14399973,
'Loss/total_loss': 0.2447994,
'learning_rate': 0.07991781}

```

Evaluating the model

In [14]: `command = "python3 {} --model_dir={} --pipeline_config_path={} --checkpoint_dir={}"`

In [15]: `print(command)`

```

python3 Tensorflow/models/research/object_detection/model_main_tf2.py
--model_dir=Tensorflow/workspace/models/ssd_mobnet --pipeline_config_path=Tensorflow/workspace/models/ssd_mobnet/pipeline.config --checkpoint_dir=Tensorflow/workspace/models/ssd_mobnet

```

```

Accumulating evaluation results...
DONE (t=0.01s).
Average Precision (AP) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.570
Average Precision (AP) @[ IoU=0.50 | area= all | maxDets=100 ] = 1.000
Average Precision (AP) @[ IoU=0.75 | area= all | maxDets=100 ] = 0.668
Average Precision (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
Average Precision (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = -1.000
Average Precision (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.570
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 1 ] = 0.492
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 10 ] = 0.575
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.575
Average Recall (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
Average Recall (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = -1.000
Average Recall (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.575
INFO:tensorflow:Eval metrics at step 2000
I1119 19:21:34.815694 140508770121536 model_lib_v2.py:1015] Eval metrics at step 2000
INFO:tensorflow: + DetectionBoxes_Precision/mAP: 0.569802
I1119 19:21:34.825533 140508770121536 model_lib_v2.py:1018] + DetectionBoxes_Precision/mAP: 0.569802
INFO:tensorflow: + DetectionBoxes_Precision/mAP@.50IOU: 1.000000
I1119 19:21:34.826544 140508770121536 model_lib_v2.py:1018] + DetectionBoxes_Precision/mAP@.50IOU: 1.000000
INFO:tensorflow: + DetectionBoxes_Precision/mAP@.75IOU: 0.668317
I1119 19:21:34.827601 140508770121536 model_lib_v2.py:1018] + DetectionBoxes_Precision/mAP@.75IOU: 0.668317

```

We can also visu

Using the tensorboard.

To run the tensorboard on training, in terminal, go inside

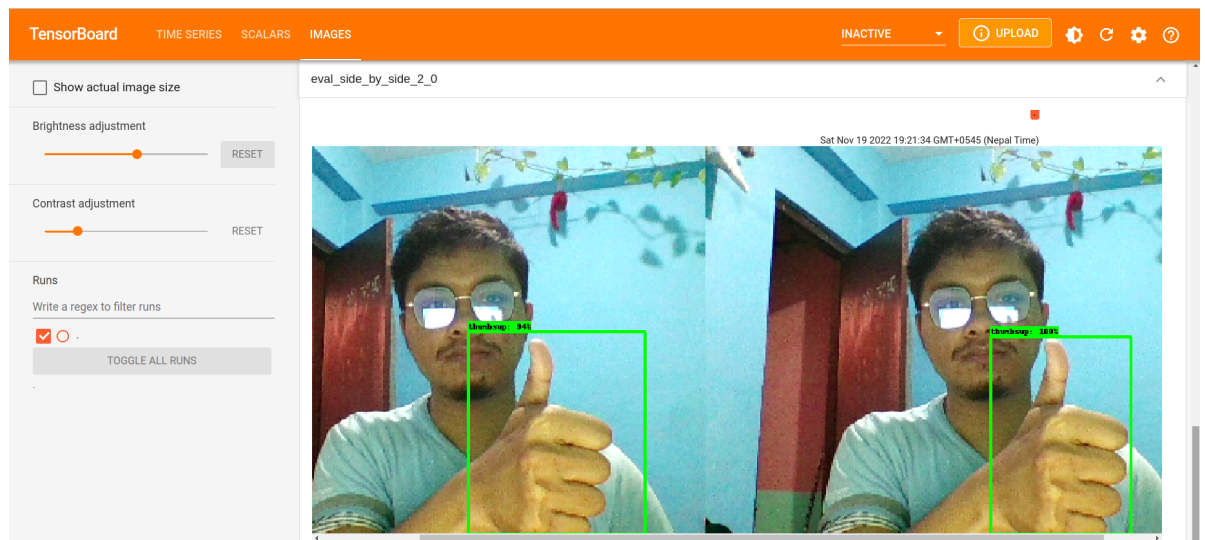
Tensorflow/workspace/models/ssd_mobnet/train and write `tensorboard --logdir=.`

To run the tensorboard on evaluation, in terminal, go inside
Tensorflow/workspace/models/ssd_mobnet/eval and write `tensorboard --logdir=.`

```
NOTE: Using experimental fast data loading logic. To disable, pass
"--load_fast=false" and report issues on GitHub. More details:
https://github.com/tensorflow/tensorboard/issues/4784

Serving TensorBoard on localhost; to expose to the network, use a proxy or pass --bind_all
TensorBoard 2.10.1 at http://localhost:6006/ (Press CTRL+C to quit)
```

After running the above mentioned command, you will get the http link. Open it and you will have your tensorboard.



Loading the trained model from the checkpoint

```
In [7]: import tensorflow as tf
from object_detection.utils import label_map_util
from object_detection.utils import visualization_utils as viz_utils
from object_detection.builders import model_builder
from object_detection.utils import config_util
```



```
In [8]: # Load pipeline config and build a detection model
configs = config_util.get_configs_from_pipeline_file(files['pipeline_config'], is_detection_model = model_builder.build(model_config=configs['model'], is_detection_model = True))

# Restore checkpoint
ckpt = tf.compat.v2.train.Checkpoint(model=detection_model)
ckpt.restore(os.path.join(paths['checkpoint_path'], 'ckpt-3')).expect_partial()

@tf.function
def detect_fn(image):
    image, shapes = detection_model.preprocess(image)
    prediction_dict = detection_model.predict(image, shapes)
    detections = detection_model.postprocess(prediction_dict, shapes)
    return detections
```

```
2022-11-19 20:03:03.860604: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libcuda.so.1'; dlerror: libcuda.so.1: cannot open shared object file: No such file or directory
2022-11-19 20:03:03.860634: W tensorflow/stream_executor/cuda/cuda_driver.cc:263] failed call to cuInit: UNKNOWN ERROR (303)
2022-11-19 20:03:03.860663: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:156] kernel driver does not appear to be running on this host (shreejan-Inspiron-5559): /proc/driver/nvidia/version does not exist
2022-11-19 20:03:03.860949: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
```

Object Detection in an image

```
In [9]: import cv2
import numpy as np
from matplotlib import pyplot as plt
%matplotlib inline
```

```
In [10]: category_index = label_map_util.create_category_index_from_labelmap(file_path)
```

```
In [11]: IMAGE_PATH = os.path.join(paths['image_path'], 'validation', 'thumbsup.1')
```

```

In [12]: img = cv2.imread(IMAGE_PATH)
         image_np = np.array(img)

         input_tensor = tf.convert_to_tensor(np.expand_dims(image_np, 0), dtype=tf.float32)
         detections = detect_fn(input_tensor)

         num_detections = int(detections.pop('num_detections'))
         detections = {key: value[0, :num_detections].numpy()
                        for key, value in detections.items()}
         detections['num_detections'] = num_detections

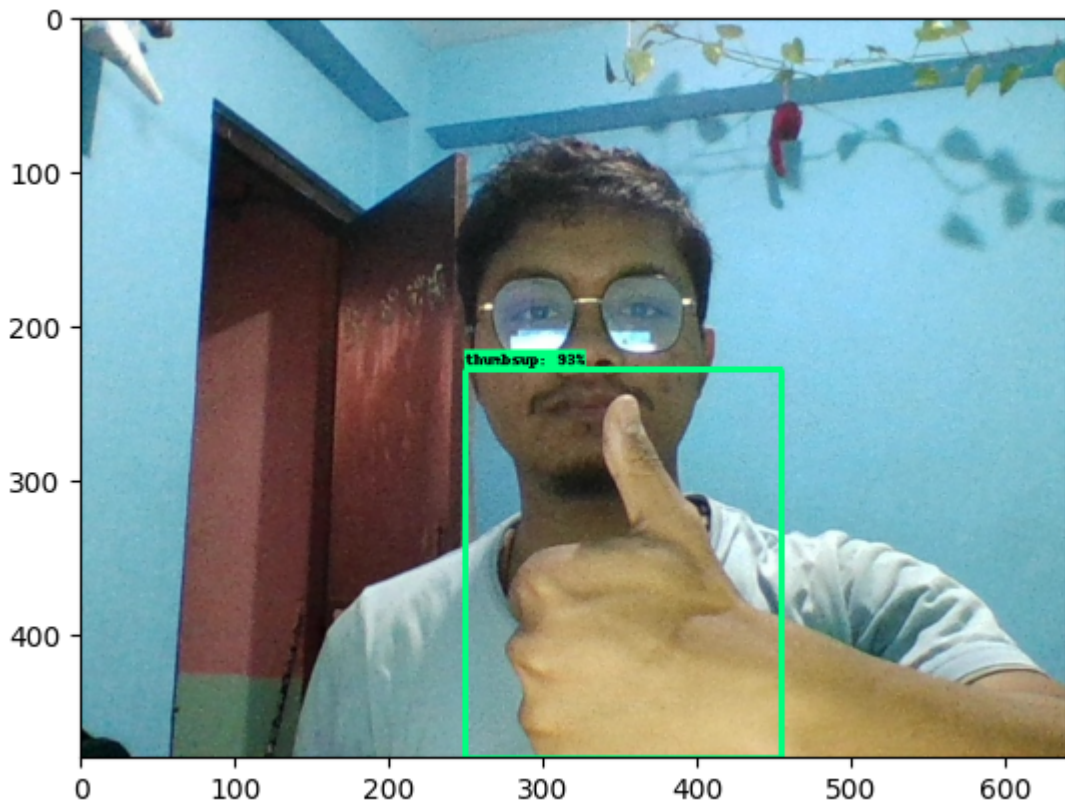
         # detection_classes should be ints.
         detections['detection_classes'] = detections['detection_classes'].astype(np.int64)

         label_id_offset = 1
         image_np_with_detections = image_np.copy()

         viz_utils.visualize_boxes_and_labels_on_image_array(
             image_np_with_detections,
             detections['detection_boxes'],
             detections['detection_classes'] + label_id_offset,
             detections['detection_scores'],
             category_index,
             use_normalized_coordinates=True,
             max_boxes_to_draw=5,
             min_score_thresh=.8,
             agnostic_mode=False)

         plt.imshow(cv2.cvtColor(image_np_with_detections, cv2.COLOR_BGR2RGB))
         plt.show()

```



Real-Time Object Detection from the Webcam

```
In [14]: cap = cv2.VideoCapture(0)
width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))

while cap.isOpened():
    ret, frame = cap.read()
    image_np = np.array(frame)

    input_tensor = tf.convert_to_tensor(np.expand_dims(image_np, 0), dtype=tf.float32)
    detections = detect_fn(input_tensor)

    num_detections = int(detections.pop('num_detections'))
    detections = {key: value[0, :num_detections].numpy()
                  for key, value in detections.items()}
    detections['num_detections'] = num_detections

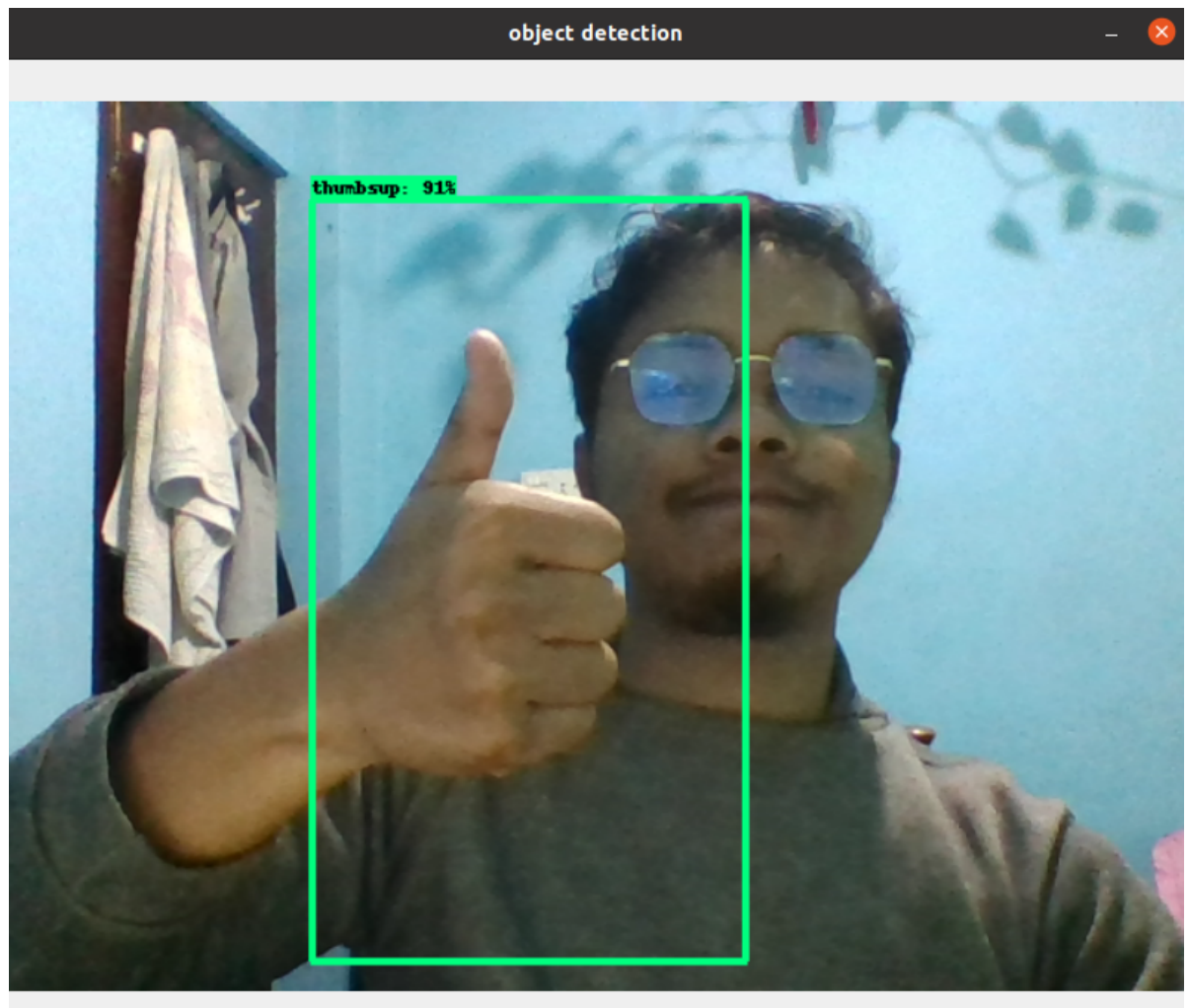
    # detection_classes should be ints.
    detections['detection_classes'] = detections['detection_classes'].astype(np.int64)

    label_id_offset = 1
    image_np_with_detections = image_np.copy()

    viz_utils.visualize_boxes_and_labels_on_image_array(
        image_np_with_detections,
        detections['detection_boxes'],
        detections['detection_classes']+label_id_offset,
        detections['detection_scores'],
        category_index,
        use_normalized_coordinates=True,
        max_boxes_to_draw=5,
        min_score_thresh=.8,
        agnostic_mode=False)

    cv2.imshow('object detection', cv2.resize(image_np_with_detections, (800, 600)))

    if cv2.waitKey(10) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```



P.S. If you encountered

error: OpenCV(4.6.0) /io/opencv/modules/highgui/src/window.cpp:1261: error: (-2:Unspecified error) The function is not implemented. Rebuild the library with Windows, GTK+ 2.x or Cocoa support. If you are on Ubuntu or Debian, install libgtk2.0-dev and pkg-config, then re-run cmake or configure script in function 'cvDestroyAllWindows'

error (which I faced) kindly refer to this stackoverflow

<https://stackoverflow.com/questions/67120450/error-2unspecified-error-the-function-is-not-implemented-rebuild-the-libra> (<https://stackoverflow.com/questions/67120450/error-2unspecified-error-the-function-is-not-implemented-rebuild-the-libra>)

```
In [34]: # !pip uninstall opencv-python-headless -y
```

WARNING: Skipping opencv-python-headless as it is not installed.

```
In [35]: # !pip install opencv-python --upgrade
```

Requirement already satisfied: opencv-python in ./hgd/lib/python3.8/site-packages (4.6.0.66)

Requirement already satisfied: numpy>=1.17.3 in ./hgd/lib/python3.8/site-packages (from opencv-python) (1.22.4)

In [36]: `!pip3 list | grep opencv`

```
opencv-python          4.6.0.66
```

Freezing the model so that we can use it later.

In [16]: `freeze_script = os.path.join(paths['api_model_path'], 'research', 'object_detection', 'exporter_main_v2.py')`

In [17]: `freeze_script`

Out[17]: 'Tensorflow/models/research/object_detection/exporter_main_v2.py '

In [19]: `command = "python3 {} --input_type=image_tensor --pipeline_config_path={}`

In [21]: `!{command}`

```
2022-11-19 20:19:39.504755: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-11-19 20:19:39.802312: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libcudart.so.11.0'; dlerror: libcudart.so.11.0: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /home/shreejan/Documents/hand_gesture_detection/hgd/lib/python3.8/site-packages/cv2/../../lib64:
2022-11-19 20:19:39.802336: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
2022-11-19 20:19:39.833085: E tensorflow/stream_executor/cuda/cuda_blas.cc:2981] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
2022-11-19 20:19:42.179171: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libnvinfer.so.7'; dlerror: libnvinfer.so.7: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /home/shreejan/Documents/hand_gesture_detection/hgd/lib/python3.8/site-packages/cv2/../../lib64:
2022-11-19 20:19:42.179171: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libnvinfer_plugin.so.7'; dlerror: libnvinfer_plugin.so.7: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /home/shreejan/Documents/hand_gesture_detection/hgd/lib/python3.8/site-packages/cv2/../../lib64:
```

In []: