

MEERUT INSTITUTE OF ENGINEERING AND TECHNOLOGY



(SESSION 2021-2022)

MINI PROJECT REPORT

ON

“OBJECT DETECTION”

**BACHELOR OF TECHNOLOGY IN
COMPUTER SCIENCE AND ENGINEERING**

Submitted to:

MR. ABHAY JAIN

Submitted by:

Nidhi Singhal - RN:2000681530033

Shreya Singh - RN: 2000681530047

Prachi Gupta - RN:2000681530037

3rd SEMESTER

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MEERUT INSTITUTE OF ENGINEERING AND TECHNOLOGY, MEERUT

DECLARATION

We hereby declare that the project entitled - “**Object Detection**”, which is being submitted as Mini Project in department of Computer Science and engineering to Meerut Institute of Engineering and Technology, Meerut (U.P.) is an authentic record of our genuine work done under the guidance of Assistant Prof. “**Mr. Abhay Jain**” of Computer Science and Engineering, Meerut Institute of Engineering and Technology, Meerut.

DATE: 17 JANUARY 2022

PLACE: MEERUT

NAME: Nidhi Singhal (2000681530033)

Prachi Gupta (2000681530037)

Shreya Singh (2000681530047)

CERTIFICATE

This is to certify that mini project report entitled – “**Object Detection**” submitted by “**Prachi Gupta, Shreya Singh, Nidhi Singhal**” has been carried out under the guidance of “**Mr. Abhay Jain**” of Computer Science and Engineering, Meerut Institute of Engineering and Technology, Meerut. This project report is approved for Mini Project (KCN 354) in 3rd semester in computer science and engineering department from Meerut Institute of Engineering and Technology, Meerut.

Supervisor : Mr. Abhay Jain

Date: 17 January 2022

ACKNOWLEDGEMENT

We express our sincere indebtedness towards our guide **Mr Abhay Jain** our professor of Computer Science and Engineering, Meerut Institute of Engineering and Technology, Meerut for his valuable suggestion, guidance and supervision throughout the work. Without his kind patronage and guidance the project would not have taken shape. We would also like to extend our gratitude and sincere regards for his kind approval of the project. Time to time counselling and advises.

We would also like to thank our HOD Dr. SWATI SHARMA, Department of Computer Science and engineering, Meerut Institute of Engineering and Technology, Meerut for her expert advice and counselling from time to time. We owe sincere thanks to all the faculty members in the department of Computer Science and engineering for their kind guidance and encouragement .

TABLE OF CONTENTS

	DESCRIPTION	PAGE NO
	COVER	1
	DECLARATION	2
	CERTIFICATE	3
	ACKNOWLEDGEMENT	4
CHAPTER 1	INTRODUCTION	6
	1.1 Object detection	6
	1.2 Background	7
CHAPTER 2	System Design	8
	2.1 Flowchart	8
	2.2 Tech stack	9
	2.3 Programming Tools	11
CHAPTER 3	PROCEDURE/ METHODODLOGIES	13
	3.1 What is Object-detection API ?	13
	3.2 Procedure	13
CHAPTER 4	OUTPUT	16
CHAPTER 5	APPENDIX	18
	5.1 implementation code	18
	5.2 TF record file	22
	5.3 TRAINING STEP	23
	REFERENCES	28

CHAPTER 1

INTRODUCTION

Object detection :

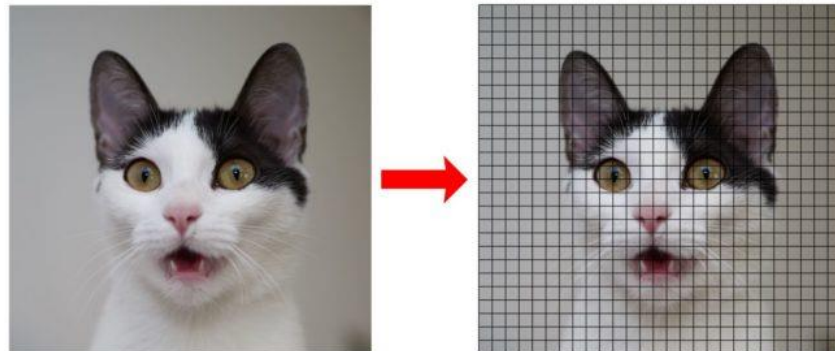
Object detection is a computer vision technique in which a software system can detect, locate, and trace the object from a given image or video. It identifies the class of object (person, table, chair, etc.) and their location-specific coordinates in the image. The location is pointed out by drawing a bounding box around the object. The bounding box may or may not accurately locate the position of the object. The ability to locate the object inside an image defines the performance of the algorithm used for detection .

Here in our project we used pre-trained models for training our system because creating from scratch would require an in-depth knowledge of CNN and DEEP neural network.

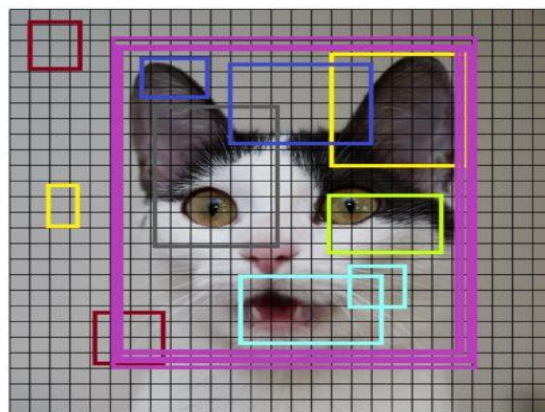
BACKGROUND

Generally, the object detection task is carried out in three steps:

- Generates the small segments in the input as shown in the image below. As you can see the large set of bounding boxes are spanning the full image.



- Feature extraction is carried out for each segmented rectangular area to predict whether the rectangle contains a valid object.

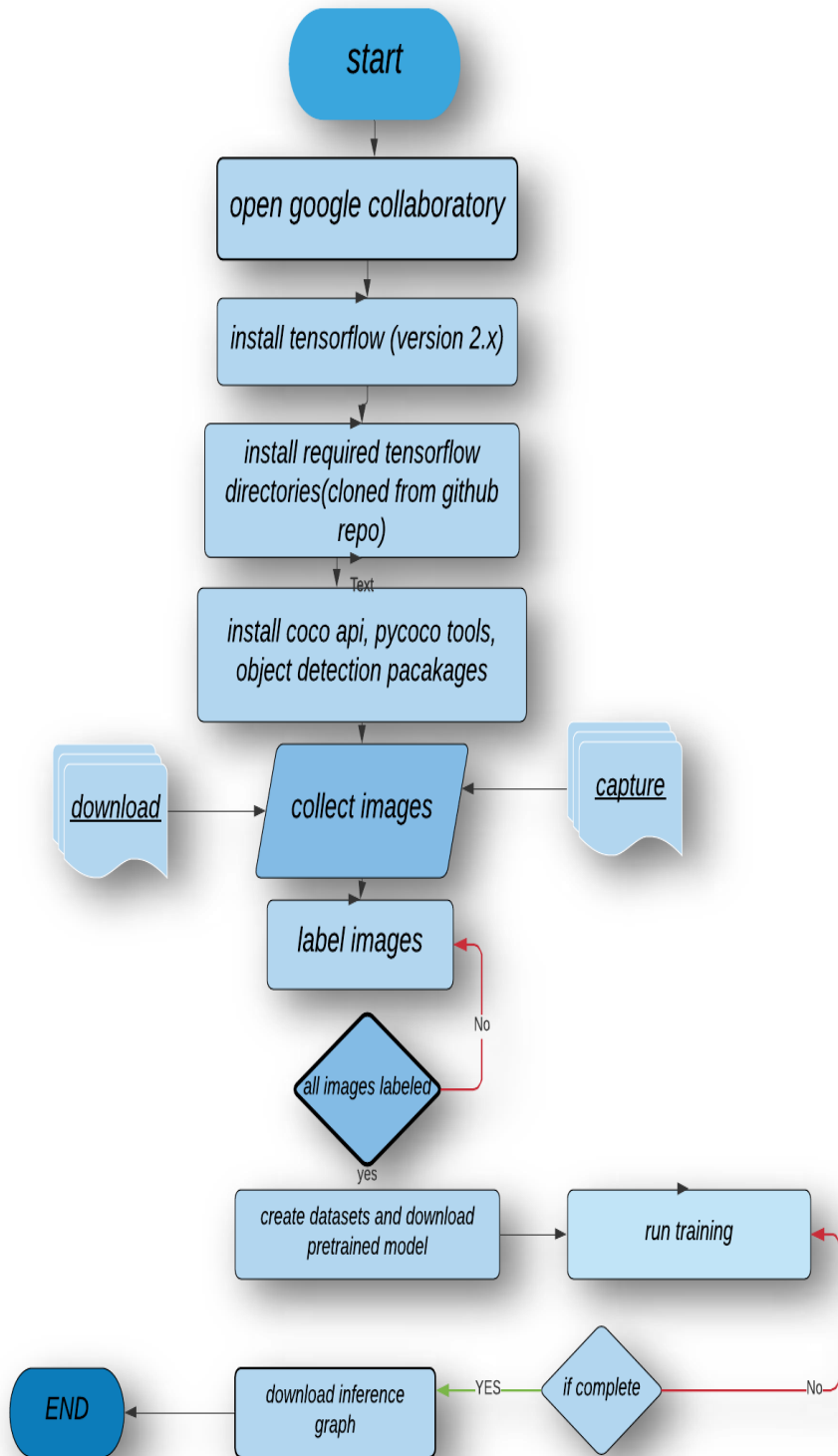


- Overlapping boxes are combined into a single bounding rectangle (Non-Maximum Suppression)



Chapter – 2 **SYSTEM DESIGN**

FLOWCHART



Tech stack

TENSORFLOW: TENSORFLOW is an open-source library for numerical computation and large-scale machine learning that ease Google Brain Tensor Flow, the process of acquiring data, training models, serving predictions, and refining future results.

- Tensorflow bundles through **MACHINE LEARNING** and **DEEP LEARNING** models and algorithms.



GOOGLE COLABORATORY: Colaboratory, or “Colab” for short, is a **product from Google Research**. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. It also provides us GPU or TPU support which is needed in object-detection .



GITHUB : GitHub is a **code hosting platform for version control and collaboration**. It lets you and others work together on projects from anywhere . Provider of Internet hosting for software development and version control using Git. It offers the distributed version control .



Labellmg: Labellmg is a **free, open source tool for graphically labeling images**. It's written in Python and uses QT for its graphical interface. It's an easy, free way to label a few hundred images to try out your next object detection project.



INTERFACE:



PROGRAMMING TOOLS

ANACONDA : **Anaconda** is a distribution of the PYTHON and R Programming Languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analysis etc.), that aims to simplify package management and deployment.

Provides anaconda prompt used in our project.



ATOM : **Atom** is a free and open source text and [source](#) code editor for macOS and LINUX, and Microsoft Windows with support for plug-ins written in Javascript and embedded Git Control .



OTHER TECHNOLOGIES USED:

1. Browser – Internet explorer,chrome
2. Nvidia GPU (GTX 650 or newer)
3. CUDA Toolkit v11.2
4. Flowchart – Visme Flowchart maker
5. CuDNN 8.1.0

Chapter – 3

PROCEDURE / METHAODOLOGY

What is Tensorflow object detection API?

The TensorFlow Object Detection API is an open-source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models.

- There are already pre-trained models in their framework which are referred to as Model Zoo.
- It includes a collection of pre-trained models trained on various datasets such as the
 - COCO (Common Objects in Context) dataset,

PROCEDURE

1. INSTALL ANACONDA PYTHON 3.8 / IF WORKING ON COLLABORATORY CREATE A NEW NOTEBOOK.

Setting up a virtual environment

2. INSTALL TENSORFLOW VERSION 2.X OR ABOVE
If you have gpu in your system install Tensorflow-gpu.
3. CLONE OR DOWNLOAD TESORFLOW GITHUB REPOSITORY.
<https://github.com/tensorflow/models.git>

NOTE : DOWNLOADING TENSORFLOW API CAN BE DONE MANUALLY OR BY GITHUB.

4. NOW CLONE AND DOWNLOAD COCO API,
<https://github.com/cocodataset/cocoapi.git>

COCO API: COCO is a large image dataset designed for object detection, segmentation, person keypoints detection, stuff segmentation, and caption generation. This package provides Matlab, Python, and Lua APIs that assists in loading, parsing, and visualizing the annotations in COCO.

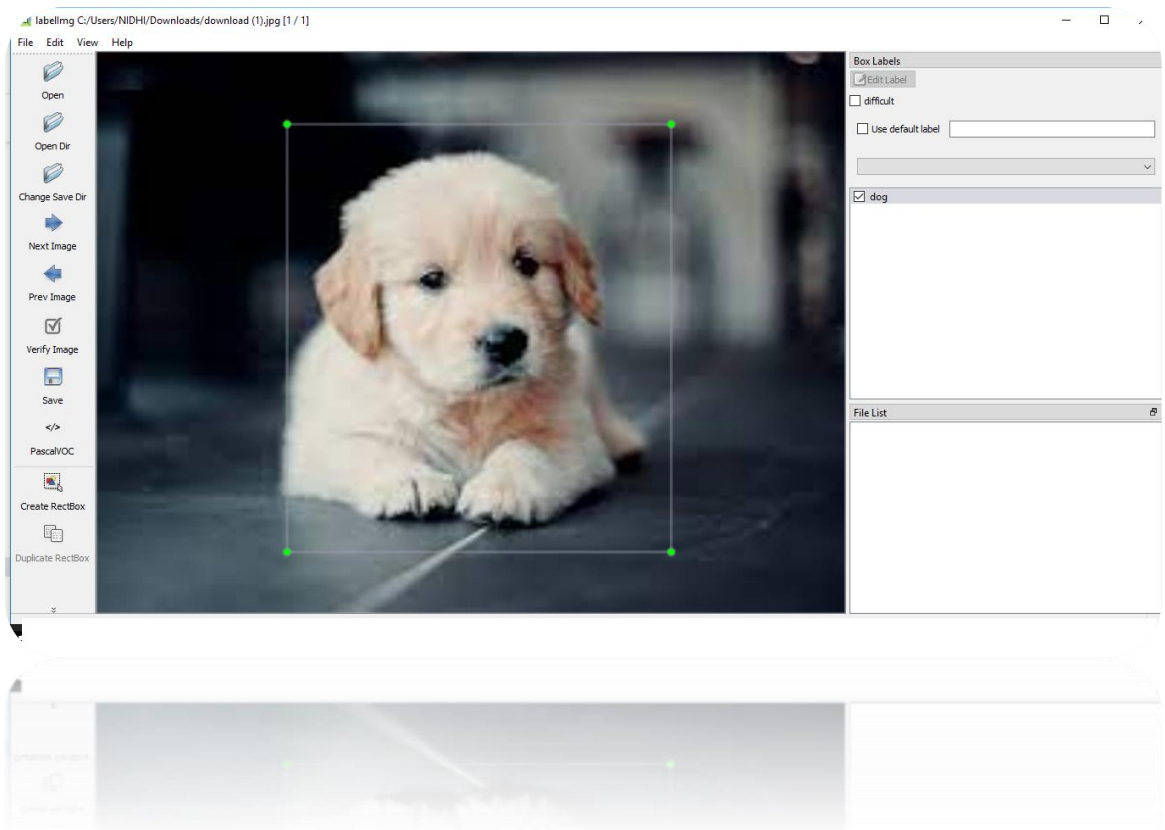
5. `pip install pillow Cython lxml jupyter matplotlib contextlib2 tf_slim`

6. Download object detection packages in research folder of model.

7. create work environment now . Here image data set is created and pre-trained model is downloaded.

Image dataset must have ration of 9:1 where 9 is for train and 1 for test.

Along with ther labeled xml files created using labelImg.



Pre-trained model is downloaded from models zoo here there is trade off between speed and accuracy if for real time we use speed over accuracy hence **faster-rcnn** or any else can be used we used **SSD Resnet** because of average accuracy.

Here **mAP (mean average precision)** is the product of precision and recall on detecting bounding boxes. It's a good combined measure for how sensitive the network is to objects of interest and how well it avoids false alarms. The higher the mAP score, the more accurate the network is but that comes at the cost of execution speed which we want to avoid here.

Model zoo:

Model name	Speed (ms)	COCO mAP[^1]	Outputs
ssd_mobilenet_v1_coco	30	21	Boxes
ssd_mobilenet_v1_0.75_depth_coco ☆	26	18	Boxes
ssd_mobilenet_v1_quantized_coco ☆	29	18	Boxes
ssd_mobilenet_v1_0.75_depth_quantized_coco ☆	29	16	Boxes
ssd_mobilenet_v1_ppn_coco ☆	26	20	Boxes
ssd_mobilenet_v1_fpn_coco ☆	56	32	Boxes
ssd_resnet_50_fpn_coco ☆	76	35	Boxes
ssd_mobilenet_v2_coco	31	22	Boxes
ssd_mobilenet_v2_quantized_coco	29	22	Boxes
ssdlite_mobilenet_v2_coco	27	22	Boxes
ssd_inception_v2_coco	42	24	Boxes
faster_rcnn_inception_v2_coco	58	28	Boxes
faster_rcnn_resnet50_coco	89	30	Boxes

8. Now tfreord file is created.

9. Configuration file is edited then changed as per our needs.

10. Then training process is started depending upon gpu presence or absence it may take time if done on cpu may take more than an hour.

11. After training model is tested by images present.

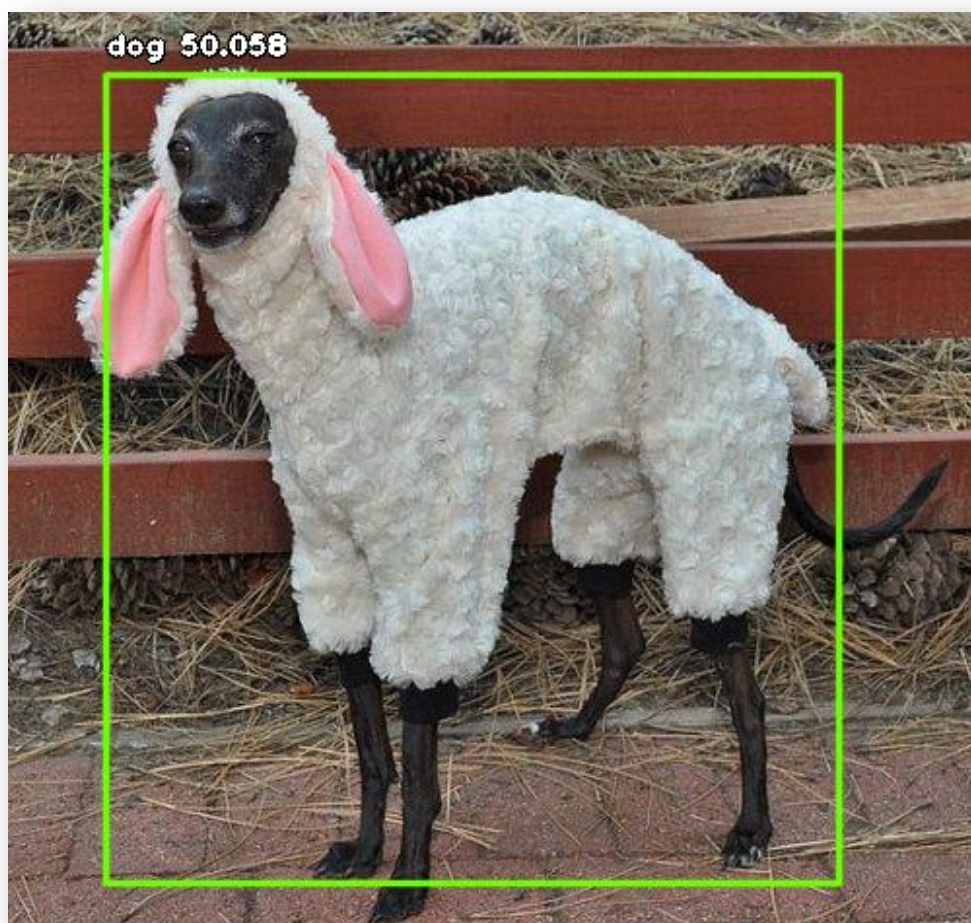
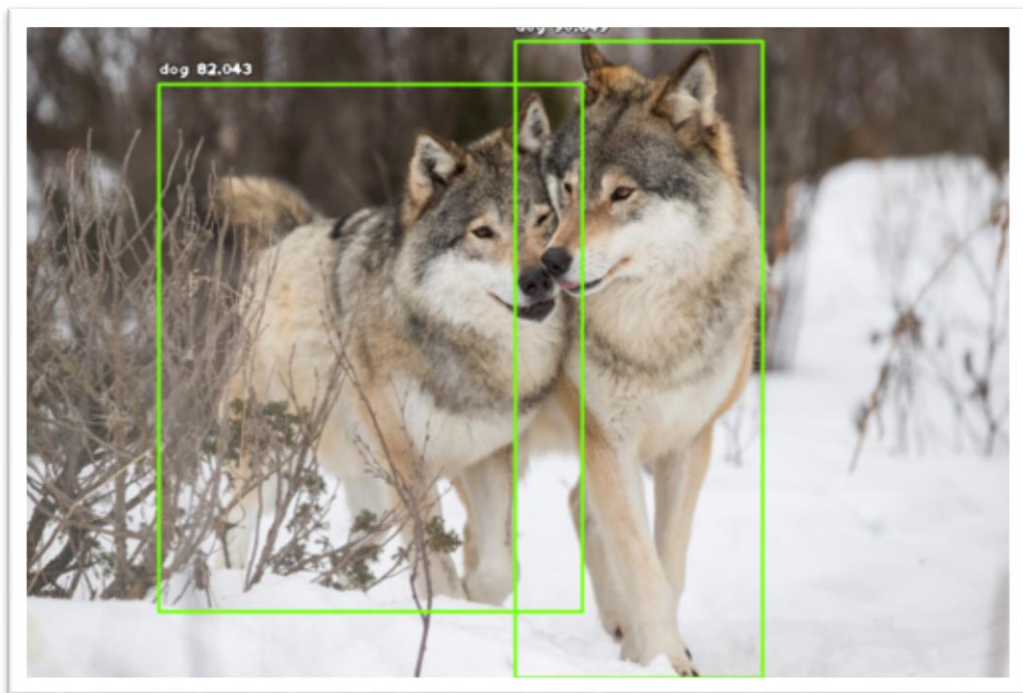
MORE WE TRAIN MORE ACCURATE WILL BE THE OUTPUT.

Note: In our project we only trained it for dog class making it static.

Chapter – 4

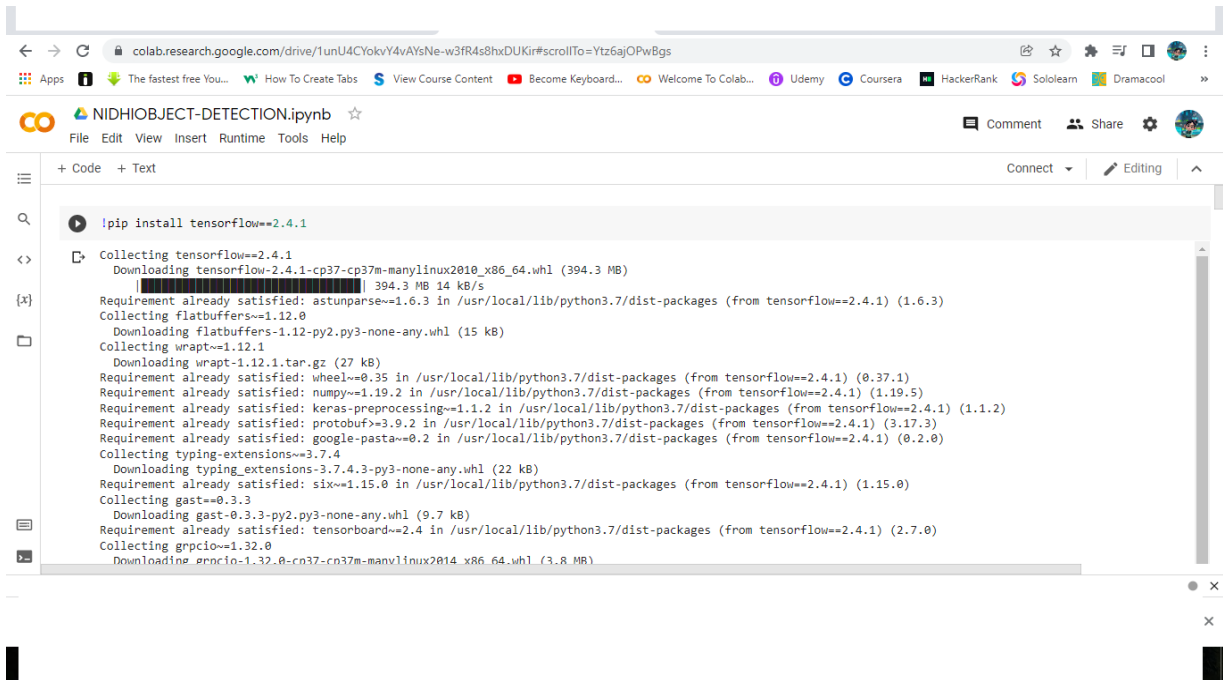
OUTPUT





Chapter – 5 APPENDIX

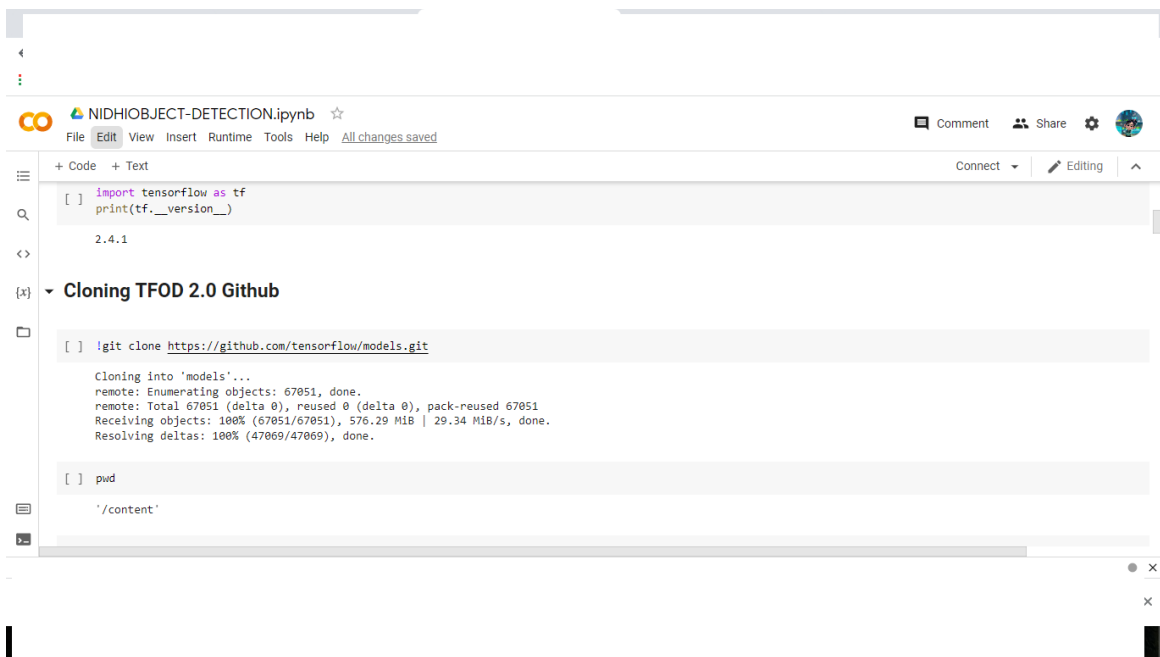
IMPLEMENATATION CODE



The screenshot shows a Jupyter Notebook titled "NIDHIOBJECT-DETECTION.ipynb" in a web browser. The code cell contains the command `!pip install tensorflow==2.4.1`. The output shows the installation progress for TensorFlow 2.4.1 and its dependencies, including flatbuffers, wrapt, typing_extensions, gast, and grpcio. The installation is successful.

```
!pip install tensorflow==2.4.1

Collecting tensorflow==2.4.1
  Downloading tensorflow-2.4.1-cp37-cp37m-manylinux2010_x86_64.whl (394.3 MB)
    394.3 MB 14 kB/s
Requirement already satisfied: astunparse==1.6.3 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (1.6.3)
Collecting flatbuffers==1.12.0
  Downloading flatbuffers-1.12-py2.py3-none-any.whl (15 kB)
Collecting wrapt==1.12.1
  Downloading wrapt-1.12.1.tar.gz (27 kB)
Requirement already satisfied: wheel==0.35 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (0.37.1)
Requirement already satisfied: numpy==1.19.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (1.19.5)
Requirement already satisfied: keras-preprocessing==1.1.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (1.1.2)
Requirement already satisfied: protobuf>=3.9.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (3.17.3)
Requirement already satisfied: google-pasta==0.2 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (0.2.0)
Collecting typing_extensions==3.7.4
  Downloading typing_extensions-3.7.4.3-py3-none-any.whl (22 kB)
Requirement already satisfied: six==1.15.0 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (1.15.0)
Collecting gast==0.3.3
  Downloading gast-0.3.3-py2.py3-none-any.whl (9.7 kB)
Requirement already satisfied: tensorboard==2.4 in /usr/local/lib/python3.7/dist-packages (from tensorflow==2.4.1) (2.7.0)
Collecting grpcio==1.32.0
  Downloading grpcio-1.32.0-cp37-cp37m-manylinux2014_x86_64.whl (3.8 MB)
```



The screenshot shows a Jupyter Notebook titled "NIDHIOBJECT-DETECTION.ipynb" in a web browser. The code cell contains the command `!git clone https://github.com/tensorflow/models.git`. The output shows the cloning process, including enumerating objects, receiving objects, and resolving deltas. The cloning is successful.

```
[ ] import tensorflow as tf
print(tf.__version__)

2.4.1

Cloning TFOD 2.0 Github

[ ] !git clone https://github.com/tensorflow/models.git

Cloning into 'models'...
remote: Enumerating objects: 67051, done.
remote: Total 67051 (delta 0), reused 0 (delta 0), pack-reused 67051
Receiving objects: 100% (67051/67051), 576.29 MiB | 29.34 MiB/s, done.
Resolving deltas: 100% (47069/47069), done.

[ ] pwd

'/content'
```

NIDHIOBJECT-DETECTION.ipynb

☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Connect Editing

[] cd /content/models/research

/content/models/research

[] !protoc object_detection/protos/*.proto --python_out=.

!git clone https://github.com/cocodataset/cocoapi.git

Cloning into 'cocoapi'...
remote: Enumerating objects: 975, done.
remote: Total 975 (delta 0), reused 0 (delta 0), pack-reused 975
Receiving objects: 100% (975/975), 11.72 MiB | 22.44 MiB/s, done.
Resolving deltas: 100% (576/576), done.

[] cd cocoapi/PythonAPI

/content/models/research/cocoapi/PythonAPI

[] !make

NIDHIOBJECT-DETECTION.ipynb

☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Connect Editing

[] !make

python setup.py build_ext --inplace
running build_ext
cythoning pycocotools/_mask.pyx to pycocotools/_mask.c
/usr/local/lib/python3.7/dist-packages/Cython/Compiler/Main.py:369: FutureWarning: Cython directive 'language_level' not set, using 2 for now (Py2). This will
tree = Parsing.p_module(s, pxd, full_module_name)
building 'pycocotools._mask' extension
creating build
creating build/common
creating build/temp.linux-x86_64-3.7
creating build/temp.linux-x86_64-3.7/pycocotools
x86_64-linux-gnu-gcc -pthread -Wno-unused-result -Wsign-compare -DNDEBUG -g -fwrapv -O2 -Wall -g -fdebug-prefix-map=/build/python3.7-Y7dWB/python3.7-3.7.12=. .
../common/maskApi.c: In function 'rleDecode':
../common/maskApi.c:46:7: warning: this 'for' clause does not guard... [-Wmisleading-indentation]
for(k=0; k<R[i].cnts[j]; k++) *(M++)=v; v=!v; }
^~
../common/maskApi.c:46:49: note: ...this statement, but the latter is misleadingly indented as if it were guarded by the 'for'
for(k=0; k<R[i].cnts[j]; k++) *(M++)=v; v=!v; }
^
../common/maskApi.c: In function 'rleFrPoly':
../common/maskApi.c:166:3: warning: this 'for' clause does not guard... [-Wmisleading-indentation]
for(j=0; j<k; j++) x[j]=(int)(scale*xy[j*2+0]+.5); x[k]=x[0];
^~

```
NIDHI OBJECT-DETECTION.ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
Connect Editing

[ ] cp object_detection/packages/tf2/setup.py .

[ ] !python -m pip install .
Collecting apache-beam
  Downloading apache-beam-2.35.0-cp37m-cp37m-manylinux2010_x86_64.whl (9.9 MB)
    9.9 MB 8.9 MB/s
Requirement already satisfied: pillow in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (7.1.2)
Requirement already satisfied: lxml in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (4.2.6)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (3.2.2)
Requirement already satisfied: Cython in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (0.29.26)
Requirement already satisfied: contextlib2 in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (0.5.5)
Collecting tf-slim
  Downloading tf-slim-1.1.0-py2.py3-none-any.whl (352 kB)
    352 kB 37.5 MB/s
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (1.15.0)
Requirement already satisfied: pycocotools in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (2.0.4)
Collecting lvis
  Downloading lvis-0.5.3-py3-none-any.whl (14 kB)
Requirement already satisfied: scipy in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (1.4.1)
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (from object-detection==0.1) (1.1.5)
Collecting tf-models-official>=2.5.1
  Downloading tf-models-official-2.7.0-py2.py3-none-any.whl (1.8 MB)
    1.8 MB 36.1 MB/s
```

```
NIDHI OBJECT-DETECTION.ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
Connect Editing

[ ] # From within TensorFlow/models/research/
!python object_detection/builders/model_builder_tf2_test.py

Running tests under Python 3.7.12: /usr/bin/python3
[ RUN      ] ModelBuilderTF2Test.test_create_center_net_deepmac
2022-01-16 16:45:02.876797: W tensorflow/core/common_runtime/gpu/gpu_bfc_allocator.cc:39] Overriding allow_growth setting because the TF_FORCE_GPU_ALLOW_GROWTH
W0116 16:45:03.286084 140036829165440 model_builder.py:1100] Building experimental DeepMAC meta-arch. Some features may be omitted.
INFO:tensorflow:time(__main__.ModelBuilderTF2Test.test_create_center_net_deepmac): 1.43s
I0116 16:45:03.628643 140036829165440 test_util.py:2309] time(__main__.ModelBuilderTF2Test.test_create_center_net_deepmac): 1.43s
[ OK      ] ModelBuilderTF2Test.test_create_center_net_deepmac
[ RUN      ] ModelBuilderTF2Test.test_create_center_net_model0 (customize_head_params=True)
INFO:tensorflow:time(__main__.ModelBuilderTF2Test.test_create_center_net_model0 (customize_head_params=True)): 0.63s
I0116 16:45:04.260837 140036829165440 test_util.py:2309] time(__main__.ModelBuilderTF2Test.test_create_center_net_model0 (customize_head_params=True)): 0.63s
[ OK      ] ModelBuilderTF2Test.test_create_center_net_model0 (customize_head_params=True)
[ RUN      ] ModelBuilderTF2Test.test_create_center_net_model1 (customize_head_params=False)
INFO:tensorflow:time(__main__.ModelBuilderTF2Test.test_create_center_net_model1 (customize_head_params=False)): 0.35s
I0116 16:45:04.612725 140036829165440 test_util.py:2309] time(__main__.ModelBuilderTF2Test.test_create_center_net_model1 (customize_head_params=False)): 0.35s
[ OK      ] ModelBuilderTF2Test.test_create_center_net_model1 (customize_head_params=False)
[ RUN      ] ModelBuilderTF2Test.test_create_center_net_model_from_keypoints
INFO:tensorflow:time(__main__.ModelBuilderTF2Test.test_create_center_net_model_from_keypoints): 0.33s
I0116 16:45:04.939988 140036829165440 test_util.py:2309] time(__main__.ModelBuilderTF2Test.test_create_center_net_model_from_keypoints): 0.33s
[ OK      ] ModelBuilderTF2Test.test_create_center_net_model_from_keypoints
[ RUN      ] ModelBuilderTF2Test.test_create_center_net_model_mobilenet
```

TF RECORD FILE

```
NIDHIOBJECT-DETECTION.ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
Connect Editing

[ ] cd /content/training_demo/pre-trained-models
/content/training_demo/pre-trained-models

[ ] wget http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_resnet101_v1_fpn_640x640_coco17_tpu-8.tar.gz
--2022-01-16 16:54:45-- http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_resnet101_v1_fpn_640x640_coco17_tpu-8.tar.gz
Resolving download.tensorflow.org (download.tensorflow.org)... 172.217.212.128, 2607:f8b0:4001:c03::80
Connecting to download.tensorflow.org (download.tensorflow.org)|172.217.212.128|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 386527459 (369M) [application/x-tar]
Saving to: 'ssd_resnet101_v1_fpn_640x640_coco17_tpu-8.tar.gz'

ssd_resnet101_v1_fpn_640x640_coco17_tpu-8.tar.gz 100%[=====] 368.62M 243MB/s in 1.5s
2022-01-16 16:54:47 (243 MB/s) - 'ssd_resnet101_v1_fpn_640x640_coco17_tpu-8.tar.gz' saved [386527459/386527459]

[ ] tar -xvf ssd_resnet101_v1_fpn_640x640_coco17_tpu-8.tar.gz
ssd_resnet101_v1_fpn_640x640_coco17_tpu-8/
ssd_resnet101_v1_fpn_640x640_coco17_tpu-8/ckpt/
```

```
NIDHIOBJECT-DETECTION.ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
Connect Editing

[ ] # Create train data:
!python generate_tfrecord.py -x /content/training_demo/images/train -l /content/training_demo/annotations/label_map.pbtxt -o /content/training_demo/annotations/train.record

# Create test data:
!python generate_tfrecord.py -x /content/training_demo/images/test -l /content/training_demo/annotations/label_map.pbtxt -o /content/training_demo/annotations/test.record

Successfully created the TFRecord file: /content/training_demo/annotations/train.record
Successfully created the TFRecord file: /content/training_demo/annotations/test.record

[ ] pwd

'/content/training_demo'

[ ] ls

annotations/      exporte_tfLite_graph_tf2.py  model_main_tf2.py
exported-models/  generate_tfrecord.py         models/
exporter_main_v2.py images/                       pre-trained-models/
```


TRAINING STEP

Custom Object Detection.ipynb

File Edit View Insert Runtime Tools Help Changes will not be saved

+ Code + Text Copy to Drive

Connect Editing

```
[ ] export_tflite_graph_tf2.py models/
exported_models/ images/ pre-trained-models/
exporter_main_v2.py model_main_tf2.py __pycache__/

!python model_main_tf2.py --model_dir=/content/training_demo/models/my_ssd_resnet101_v1_fpn --pipeline_config_path=/content/training_demo/models/my_ssd_resnet101

2021-04-17 13:15:59.823117: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcudart.so.11.0
2021-04-17 13:16:01.994474: I tensorflow/compiler/jit/xla_cpu_device.cc:41] Not creating XLA devices, tf_xla_enable_xla_devices not set
2021-04-17 13:16:01.995335: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcuda.so.1
2021-04-17 13:16:02.009382: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:941] successful NUMA node read from SysFS had negative value (-1), but there
2021-04-17 13:16:02.009964: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1720] Found device 0 with properties:
pciBusID: 0000:00:04.0 name: Tesla P100-PCIE-16GB computeCapability: 6.0
coreClock: 1.3285GHz coreCount: 56 deviceMemorySize: 15.90GiB deviceMemoryBandwidth: 681.88GiB/s
2021-04-17 13:16:02.010017: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcudart.so.11.0
2021-04-17 13:16:02.012498: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcublas.so.11
2021-04-17 13:16:02.012576: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcublasLt.so.11
2021-04-17 13:16:02.014299: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcufft.so.10
2021-04-17 13:16:02.014627: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcusolver.so.10
2021-04-17 13:16:02.016393: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcusolver.so.10
2021-04-17 13:16:02.017147: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcusparsolver.so.11
2021-04-17 13:16:02.017332: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcudnn.so.8
2021-04-17 13:16:02.017437: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:941] successful NUMA node read from SysFS had negative value (-1), but there
2021-04-17 13:16:02.018047: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:941] successful NUMA node read from SysFS had negative value (-1), but there
2021-04-17 13:16:02.018563: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1862] Adding visible gpu devices: 0
```

Custom Object Detection.ipynb

File Edit View Insert Runtime Tools Help Changes will not be saved

+ Code + Text Copy to Drive

Connect Editing

```
[ ] !python exporter_main_v2.py --input_type image_tensor --pipeline_config_path /content/training_demo/models/my_ssd_resnet101_v1_fpn/pipeline.config --trained_checkpoint_dir /content/training_demo/

2021-04-17 13:48:14.171062: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcudart.so.11.0
2021-04-17 13:48:16.317732: I tensorflow/compiler/jit/xla_cpu_device.cc:41] Not creating XLA devices, tf_xla_enable_xla_devices not set
2021-04-17 13:48:16.318525: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcuda.so.1
2021-04-17 13:48:16.339905: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:941] successful NUMA node read from SysFS had negative value (-1), but there
2021-04-17 13:48:16.340490: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1720] Found device 0 with properties:
pciBusID: 0000:00:04.0 name: Tesla P100-PCIE-16GB computeCapability: 6.0
coreClock: 1.3285GHz coreCount: 56 deviceMemorySize: 15.90GiB deviceMemoryBandwidth: 681.88GiB/s
2021-04-17 13:48:16.340520: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcudart.so.11.0
2021-04-17 13:48:16.346286: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcublas.so.11
2021-04-17 13:48:16.346353: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcublasLt.so.11
2021-04-17 13:48:16.348208: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcufft.so.10
2021-04-17 13:48:16.348509: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcusolver.so.10
2021-04-17 13:48:16.350644: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcusolver.so.10
2021-04-17 13:48:16.351240: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcusparsolver.so.11
2021-04-17 13:48:16.351413: I tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully opened dynamic library libcudnn.so.8
2021-04-17 13:48:16.351499: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:941] successful NUMA node read from SysFS had negative value (-1), but there
2021-04-17 13:48:16.352070: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:941] successful NUMA node read from SysFS had negative value (-1), but there
2021-04-17 13:48:16.352560: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1862] Adding visible gpu devices: 0
2021-04-17 13:48:16.352850: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
```

OBJECT-DETECTION CODE FOR TESTING IMAGES ON TEST LEVEL:

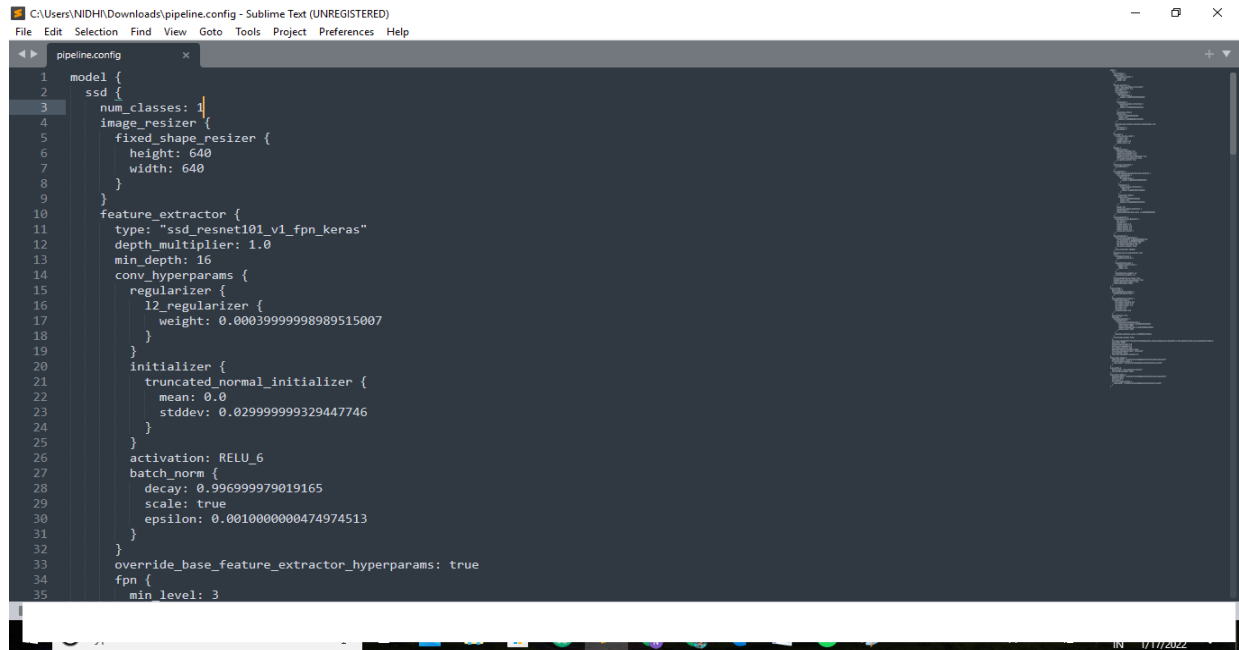
OBJ-DET.PY — C:\Users\NIDHI\Desktop\object-detection — Atom

File Edit View Selection Find Packages Help

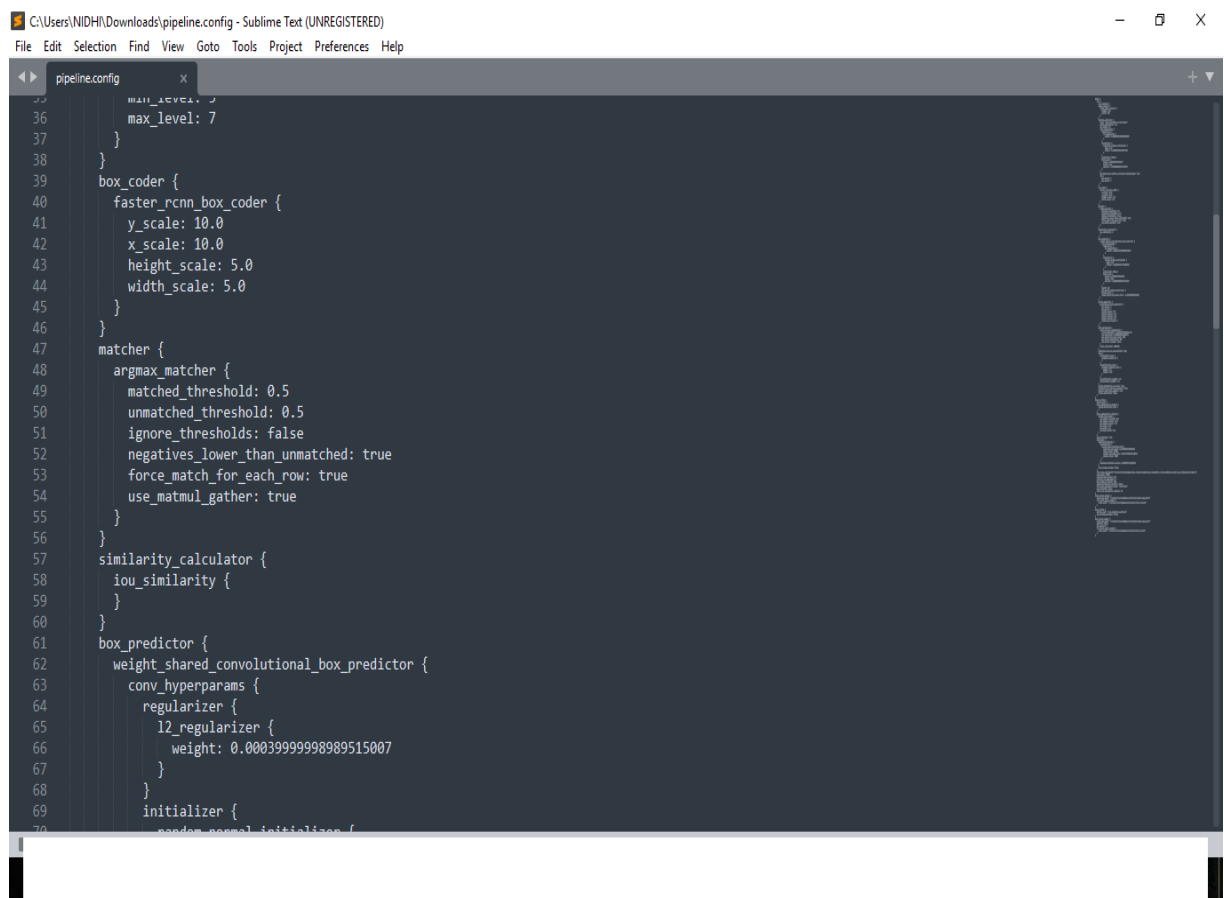
```
1 """
2 Object Detection (On Image) From TF2 Saved Model
3 =====
4 """
5
6 import os
7 os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2' # Suppress TensorFlow logging (1)
8
9 import pathlib
10 import tensorflow as tf
11 import cv2
12 import argparse
13 from google.colab.patches import cv2_imshow
14
15 # Enable GPU dynamic memory allocation
16 gpus = tf.config.experimental.list_physical_devices('GPU')
17 for gpu in gpus:
18     tf.config.experimental.set_memory_growth(gpu, True)
19
20 # PROVIDE PATH TO IMAGE DIRECTORY
21 IMAGE_PATHS = '/content/training_demo/images/train/image1.jpg'
22
23 # PROVIDE PATH TO MODEL DIRECTORY
24 PATH_TO_MODEL_DIR = '/content/training_demo/exported_models/my_model'
25
26 # PROVIDE PATH TO LABEL MAP
27 PATH_TO_LABELS = '/content/training_demo/annotations/label_map.pbtxt'
28
```

OBJ-DET.PY 7:7 CRLF UTF-8 Python GitHub Git (0)

CONFIGURATION FILE



```
1 model {
2   ssd {
3     num_classes: 1
4     image_resizer {
5       fixed_shape_resizer {
6         height: 640
7         width: 640
8       }
9     }
10    feature_extractor {
11      type: "ssd_resnet101_v1_fpn_keras"
12      depth_multiplier: 1.0
13      min_depth: 16
14      conv_hyperparams {
15        regularizer {
16          l2_regularizer {
17            weight: 0.000399999998989515007
18          }
19        }
20        initializer {
21          truncated_normal_initializer {
22            mean: 0.0
23            stddev: 0.0299999999329447746
24          }
25        }
26        activation: RELU_6
27        batch_norm {
28          decay: 0.996999979019165
29          scale: true
30          epsilon: 0.0010000000474974513
31        }
32      }
33      override_base_feature_extractor_hyperparams: true
34    }
35    fpn {
36      min_level: 3
```



```
36      max_level: 7
37    }
38  }
39  box_coder {
40    faster_rcnn_box_coder {
41      y_scale: 10.0
42      x_scale: 10.0
43      height_scale: 5.0
44      width_scale: 5.0
45    }
46  }
47  matcher {
48    argmax_matcher {
49      matched_threshold: 0.5
50      unmatched_threshold: 0.5
51      ignore_thresholds: false
52      negatives_lower_than_unmatched: true
53      force_match_for_each_row: true
54      use_matmul_gather: true
55    }
56  }
57  similarity_calculator {
58    iou_similarity {
59    }
60  }
61  box_predictor {
62    weight_shared_convolutional_box_predictor {
63      conv_hyperparams {
64        regularizer {
65          l2_regularizer {
66            weight: 0.000399999998989515007
67          }
68        }
69        initializer {
70          truncated_normal_initializer {
```



```
C:\Users\NIDHI\Downloads\pipeline.config - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

pipeline.config
69
70     random_normal_initializer {
71         mean: 0.0
72         stddev: 0.009999999776482582
73     }
74 }
75 activation: RELU_6
76 batch_norm {
77     decay: 0.996999979019165
78     scale: true
79     epsilon: 0.0010000000474974513
80 }
81 }
82 depth: 256
83 num_layers_before_predictor: 4
84 kernel_size: 3
85 class_prediction_bias_init: -4.599999904632568
86 }
87 }
88 anchor_generator {
89     multiscale_anchor_generator {
90         min_level: 3
91         max_level: 7
92         anchor_scale: 4.0
93         aspect_ratios: 1.0
94         aspect_ratios: 2.0
95         aspect_ratios: 0.5
96         scales_per_octave: 2
97     }
98 }
99 post_processing {
100     batch_non_max_suppression {
101         score_threshold: 9.9999993922529e-09
102         iou_threshold: 0.6000000238418579
103         max_detections_per_class: 100
104         max_total_detections: 100
```

```
C:\Users\NIDHI\Downloads\pipeline.config - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

pipeline.config
105     use_static_shapes: false
106 }
107 score_converter: SIGMOID
108 }
109 normalize_loss_by_num_matches: true
110 loss {
111     localization_loss {
112         weighted_smooth_l1 {
113         }
114     }
115     classification_loss {
116         weighted_sigmoid_focal {
117             gamma: 2.0
118             alpha: 0.25
119         }
120     }
121     classification_weight: 1.0
122     localization_weight: 1.0
123 }
124 encode_background_as_zeros: true
125 normalize_loc_loss_by_codesize: true
126 inplace_batchnorm_update: true
127 freeze_batchnorm: false
128 }
129 }
130 train_config {
131     batch_size: 8
132     data_augmentation_options {
133         random_horizontal_flip {
134         }
135     }
136     data_augmentation_options {
137         random_crop_image {
138             min_object_covered: 0.0
139             min_aspect_ratio: 0.75
```

```
C:\Users\NIDHI\Downloads\pipeline.config - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

pipeline.config
140     max_aspect_ratio: 3.0
141     min_area: 0.75
142     max_area: 1.0
143     overlap_thresh: 0.0
144 }
145 }
146 sync_replicas: true
147 optimizer {
148     momentum_optimizer {
149         learning_rate {
150             cosine_decay_learning_rate {
151                 learning_rate_base: 0.03999999910593033
152                 total_steps: 25000
153                 warmup_learning_rate: 0.013333000242710114
154                 warmup_steps: 2000
155             }
156         }
157         momentum_optimizer_value: 0.8999999761581421
158     }
159     use_moving_average: false
160 }
161 fine_tune_checkpoint: "/content/trainingdemo/pre-trained-models/ssd_resnet101_v1_fpn_640x640_coco17_tpu-8/checkpoint/ckpt-0"
162 num_steps: 25000
163 startup_delay_steps: 0.0
164 replicas_to_aggregate: 8
165 max_number_of_boxes: 100
166 unpad_groundtruth_tensors: false
167 fine_tune_checkpoint_type: "detection"
168 use_bfloat16: false
169 fine_tune_checkpoint_version: V2
170 }
171 train_input_reader {
172     label_map_path: "/content/trainingdemo/notations/label_map.pbtxt"
173     tf_record_input_reader {
174         input_path: "/content/trainingdemo/notations/train.record"
```

```
C:\Users\NIDHI\Downloads\pipeline.config - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

pipeline.config
164     replicas_to_aggregate: 8
165     max_number_of_boxes: 100
166     unpad_groundtruth_tensors: false
167     fine_tune_checkpoint_type: "detection"
168     use_bfloat16: false
169     fine_tune_checkpoint_version: V2
170 }
171 train_input_reader {
172     label_map_path: "/content/trainingdemo/notations/label_map.pbtxt"
173     tf_record_input_reader {
174         input_path: "/content/trainingdemo/notations/train.record"
175     }
176 }
177 eval_config {
178     metrics_set: "coco_detection_metrics"
179     use_moving_averages: false
180 }
181 eval_input_reader {
182     label_map_path: "/content/trainingdemo/notations/label_map.pbtxt"
183     shuffle: false
184     num_epochs: 1
185     tf_record_input_reader {
186         input_path: "/content/trainingdemo/notations/test.record"
187     }
188 }
189 }
```

REFERENCES

1. TFOD custom object detection step by step

<https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/>

2. GOOGLE COLABORATORY

https://colab.research.google.com/?utm_source=scs-index

3.Tensorflow Github

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

3. Images (GOOGLE): <https://www.google.co.in/>

4. Stack Overflow (for issues)

<https://stackoverflow.com/>

