ADVANCED AUTONOMOUS NAVIGATION AND ITS CONTROL ACTIONS

(MINOR PROJECT REPORT)

Submitted in
Fulfillment of the requirements for the degree of
Master of Technology
by

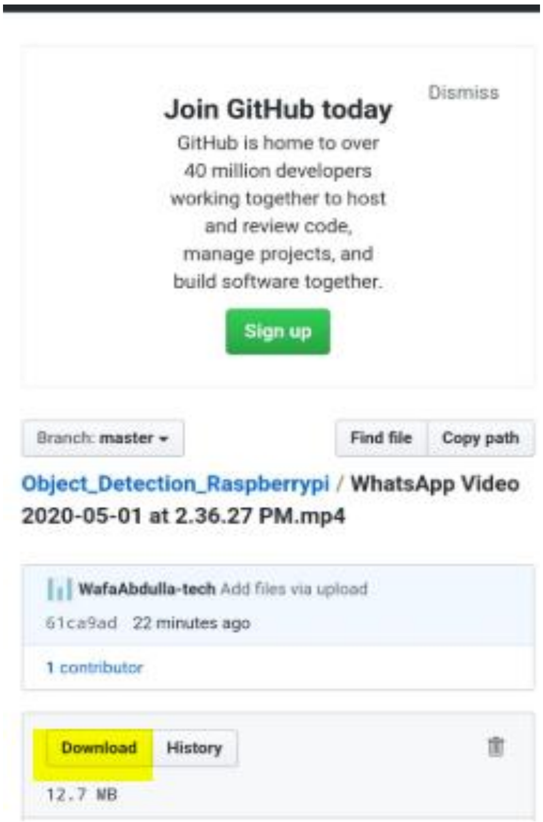
WAFA ABDULLA V.T
ID:2019PEB5458

Under the Supervision of
Mr.Rakesh Bairathi

Department of Electronics and communication Technology

Malviya National Institute Of Technology,Jaipur

INDEX



Join GitHub today

GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together.

Object_Detection_Raspberrypi / WhatsApp Video 2020-05-01 at 2.36.27 PM.mp4

WafaAbdulla-tech Add files via upload

61ca9ad 22 minutes ago

1 contributor

12.7 MB

PART 3. control actions on speed performance using control system

- Future work

INTRODUCTION

Integrating vehicle autonomy requires system to have capability to perceive real world environment. In autonomous driving Neural Networks provides a substantial role in identifying obstacle, traffic signs etc.

This project has three stages of development. First stage involves interfacing camera with raspberry pi to capture real world images and differentiate it. Second stage involves enhancing with deep learning to identify and differentiate objects using trained Algorithm, and third stage performs control actions on speed, performance using control system which makes vital role in Autonomous Navigation.

LIST OF FIGURES:

PART-1

Part-2

HARDWARE COMPONENTS :

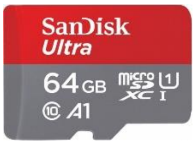
Specifications of Raspberry pi 3B+

The Raspberry Pi 3 Model B is the earliest model of the third-generation Raspberry Pi. It replaced the Raspberry Pi 2 Model B in February 2016. See also the Raspberry Pi 3 Model B+, the latest product in the Raspberry Pi 3 range.

- Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
- 1GB RAM
- BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board
- 100 Base Ethernet
- 40-pin extended GPIO
- 4 Pole stereo output and composite video port
- Full size HDMI
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- Micro SD port for loading your operating system and storing data
- Upgraded switched Micro USB power source up to 2.5A

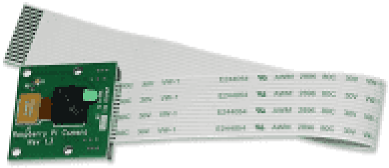
- Sd card

Secure Digital, officially abbreviated as **SD**, is a proprietary non-volatile **memory card** format developed by the **SD** Association (SDA) for use in portable devices



Pi Camera(5Mega Pixels-Version 1)

The **Pi camera** module is a portable light weight camera that supports Raspberry Pi. It communicates with Pi using the MIPI camera serial interface protocol. It is normally used in image processing, machine learning or in surveillance projects.



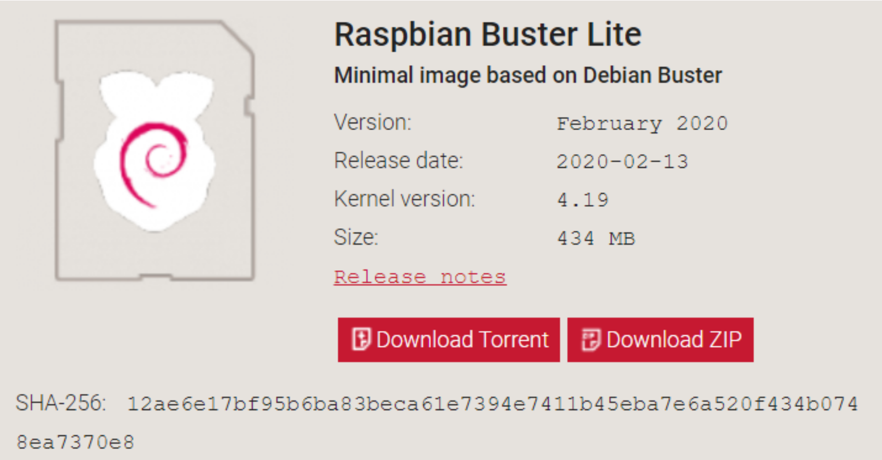
Ethernet Straight Cable

The **straight**-through is the most common type and is used to connect computers to hubs or switches.

PART -1

1.SET UP RASBERRYPI-3 AND INTERFACE CAMERA MODULE

- Download Rasbian buster lite from https://www.raspberrypi.org/downloads/raspbian/



Raspbian Buster Lite

Minimal image based on Debian Buster

Version: February 2020

Release date: 2020-02-13

Kernel version: 4.19

Size: 434 MB

SHA-256: 12ae6e17bf95b6ba83beca61e7394e7411b45eba7e6a520f434b074 8ea7370e8

Fig 1.1 Rasbian Buster Lite

- Rasbian buster lite has pre installed os
- Put SD card in to laptop and format it using sd card formatter

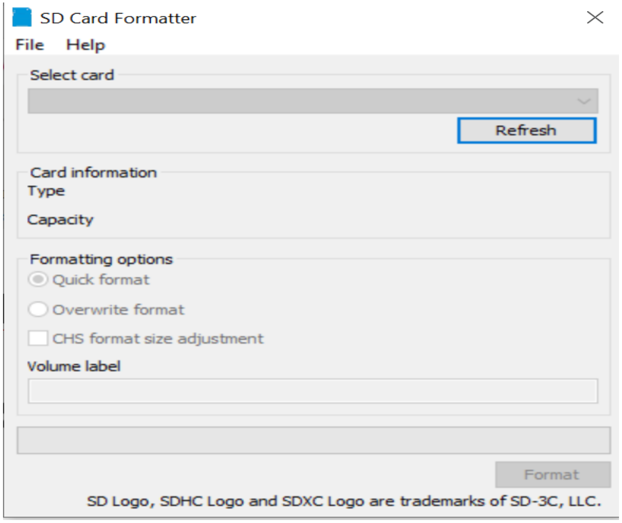


Fig 1.2 :sd card formatter

- Use Etcher balena software to flash the EEProm and to write the Rasbian buster lite .img file in to required sdcard.

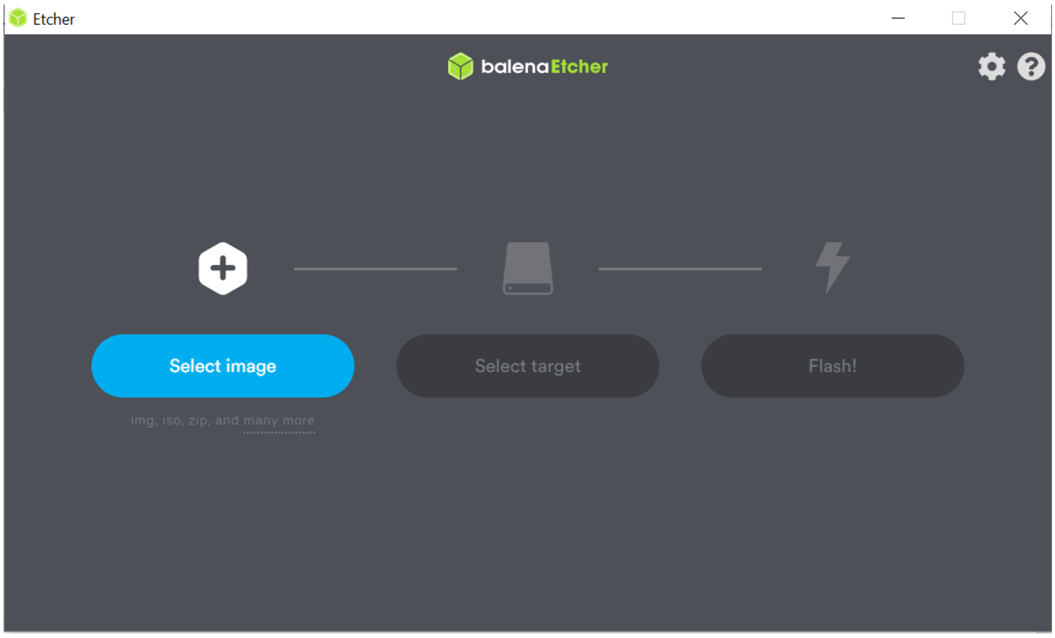
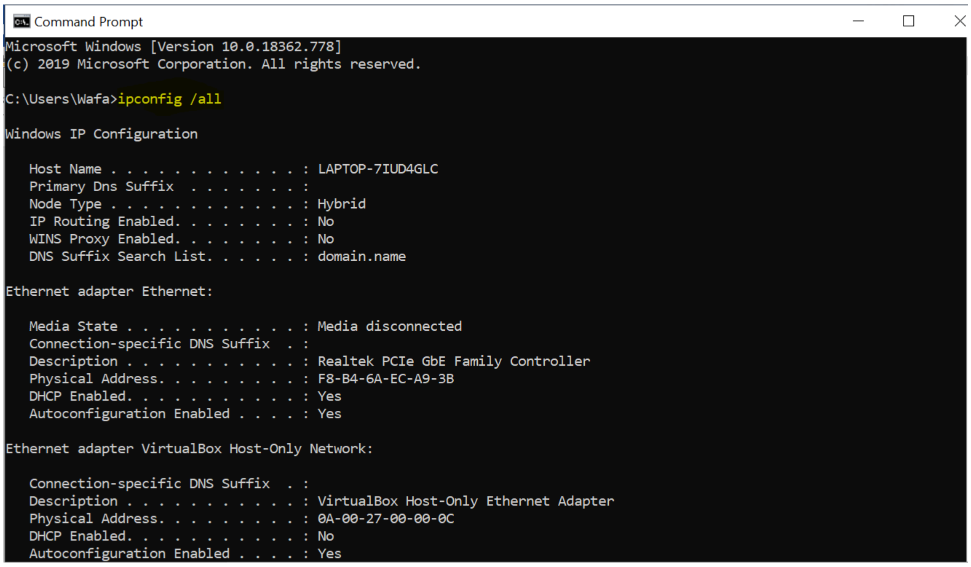


Fig 1.3:Etcher balena for to flash sd card

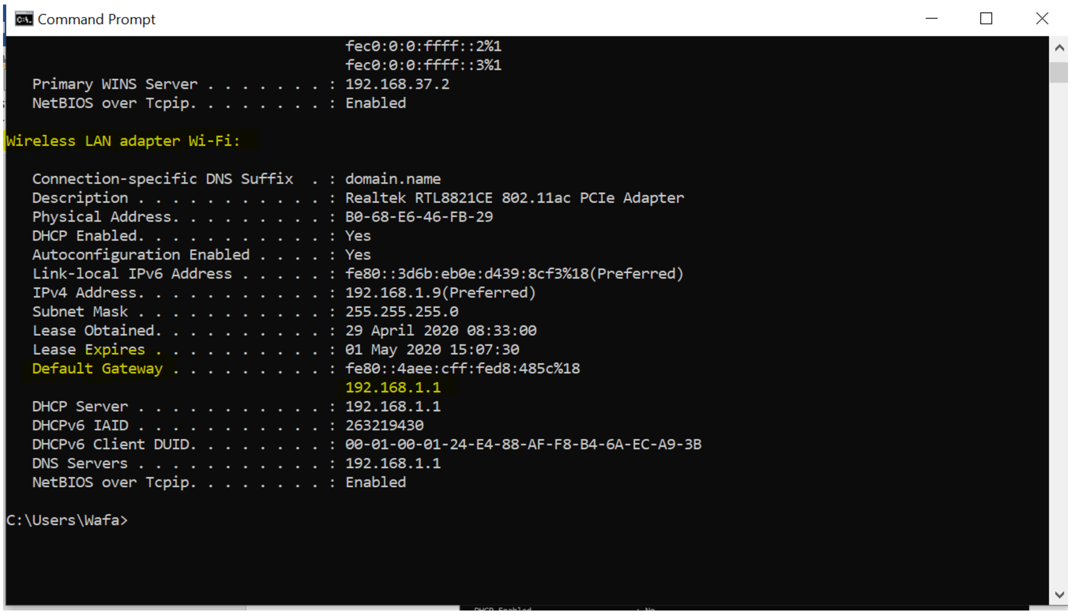
- Unmount the sdcard and put it again into laptop,open boot folder,add a file named ssh(without extension),Eject the sd card.
- Put the sd card in to Rasberrypi and plugin pi,connect an ethernet cable to Router.
- Connect wifi in laptop with the same network.
- Boot the sd card(ssh file will gets deleted from boot folder(which enables ssh in Rasberrypi)
- Check the ip address of Rasberrypi.



```
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Wafa>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : LAPTOP-7IUD4GLC
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : domain.name

Ethernet adapter Ethernet:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Realtek PCIe GbE Family Controller
   Physical Address. . . . . . . . . : F8-B4-6A-EC-A9-3B
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes

Ethernet adapter VirtualBox Host-Only Network:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : VirtualBox Host-Only Ethernet Adapter
   Physical Address. . . . . . . . . : 0A-00-27-00-00-0C
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
```

Fig 1.4:command prompt,ipconfig /all

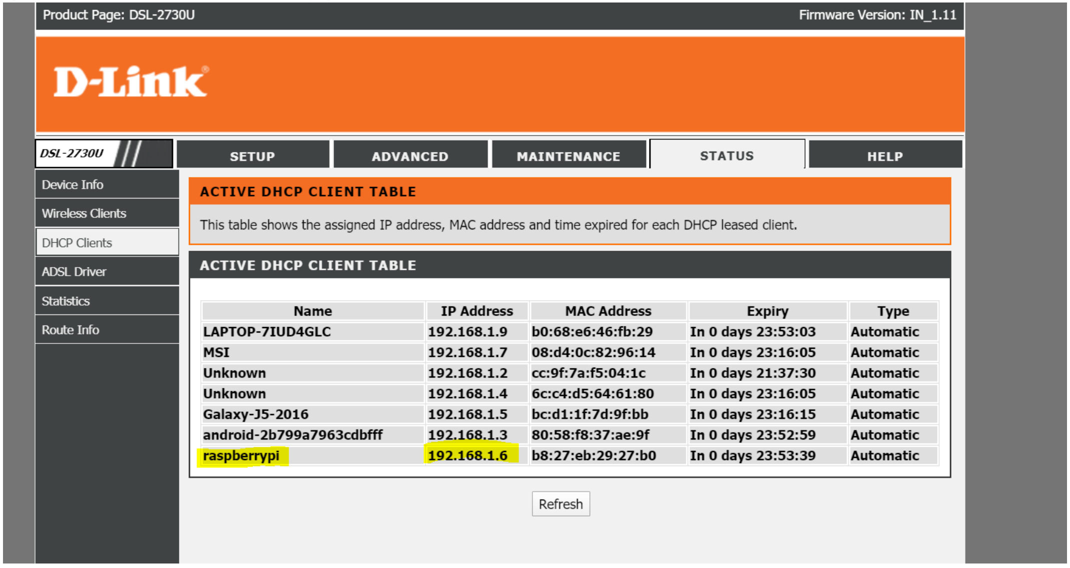
- Get the ip address of router



```
                                       fec0:0:0:ffff::2%1
                                       fec0:0:0:ffff::3%1
   Primary WINS Server . . . . . . . : 192.168.37.2
   NetBIOS over Tcpip. . . . . . . . : Enabled

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : domain.name
   Description . . . . . . . . . . . : Realtek RTL8821CE 802.11ac PCIe Adapter
   Physical Address. . . . . . . . . : B0-68-E6-46-FB-29
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::3d6b:eb0e:d439:8cf3%18(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.1.9(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : 29 April 2020 08:33:00
   Lease Expires . . . . . . . . . . : 01 May 2020 15:07:30
   Default Gateway . . . . . . . . . : fe80::4aee:cff:fed8:485c%18
                                       192.168.1.1
   DHCP Server . . . . . . . . . . . : 192.168.1.1
   DHCPv6 IAID . . . . . . . . . . . : 263219430
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-24-E4-88-AF-F8-B4-6A-EC-A9-3B
   DNS Servers . . . . . . . . . . . : 192.168.1.1
   NetBIOS over Tcpip. . . . . . . . : Enabled

C:\Users\Wafa>
```

Fig 1.5:Get the gateway address of wifi

- Check that ip address in google,In Router page check the DHCP list and get ip address of Rasberrypi(it will be in the range (192.168.1.(1-254)



ACTIVE DHCP CLIENT TABLE

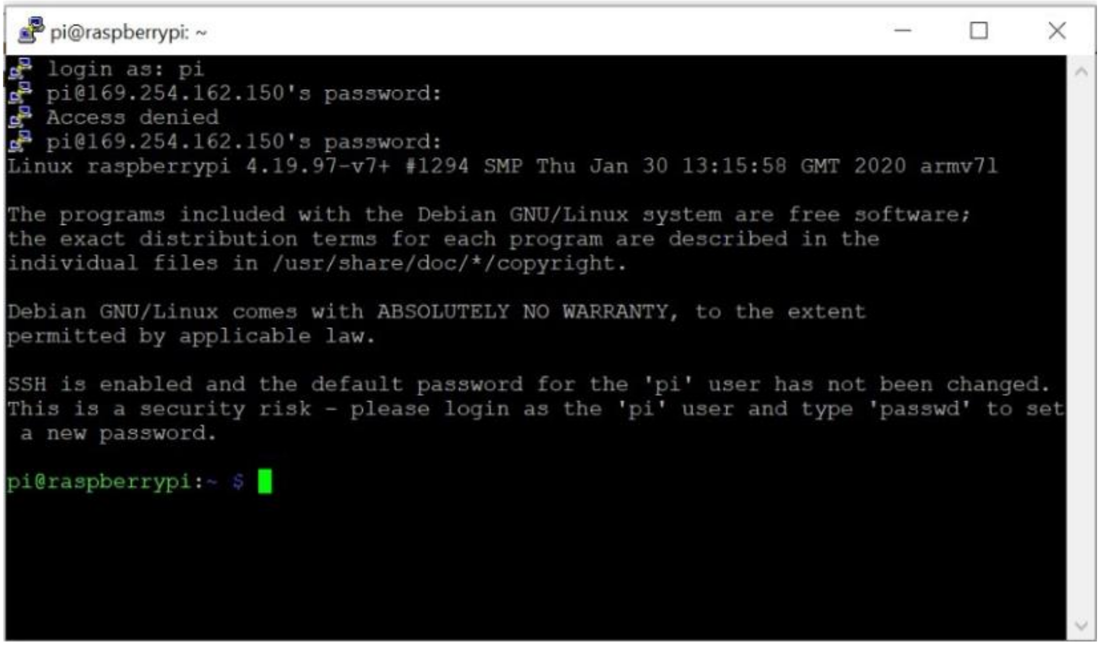
This table shows the assigned IP address, MAC address and time expired for each DHCP leased client.

ACTIVE DHCP CLIENT TABLE

Name	IP Address	MAC Address	Expiry	Type
LAPTOP-7IUD4GLC	192.168.1.9	b0:68:e6:46:fb:29	In 0 days 23:53:03	Automatic
MSI	192.168.1.7	08:d4:0c:82:96:14	In 0 days 23:16:05	Automatic
Unknown	192.168.1.2	cc:9f:7a:f5:04:1c	In 0 days 21:37:30	Automatic
Unknown	192.168.1.4	6c:c4:d5:64:61:80	In 0 days 23:16:05	Automatic
Galaxy-J5-2016	192.168.1.5	bc:d1:1f:7d:9f:bb	In 0 days 23:16:15	Automatic
android-2b799a7963cdbfff	192.168.1.3	80:58:f8:37:ae:9f	In 0 days 23:52:59	Automatic
raspberrypi	192.168.1.6	b8:27:eb:29:27:b0	In 0 days 23:53:39	Automatic

Fig 1.6:Get ip address of Raspberrypi from routers DHCP client list

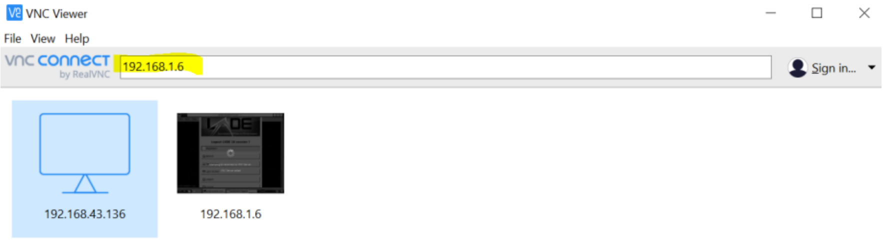
- Assign this ip address as static ip address of Rasberry pi.
- Remove the SD card from Rasberrypi,and connect to laptop using card reader,open boot folder,open cmdline.txt,append ip=192.168.1.6 at the end and save it.
- Remove sd card and put it back to Rasberrypi ,boot it.
- Open ssh(putty),type host address as the static ip,login as username pi ans password raspberry as default.



```
login as: pi
pi@169.254.162.150's password:
Access denied
pi@169.254.162.150's password:
Linux raspberrypi 4.19.97-v7+ #1294 SMP Thu Jan 30 13:15:58 GMT 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
 a new password.

pi@raspberrypi:~ $
```

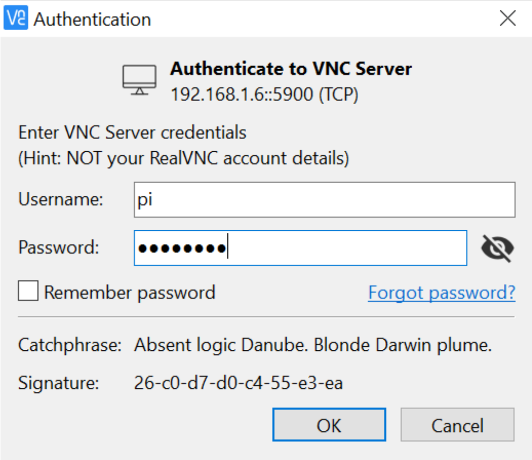
Fig 1.7:Login Raspberrypi through Putty

- Enable vnc server,type sudo raspi-config(go to interfaces,enable vnc server).
- sudo apt-get install lxsession execute the command for Debain GUI Desktop.(In boot option,select Desktop)and Reboot Raspberrypi.
- Install vnc viewer



192.168.1.6

Fig 1.8:Connect to vnc viewer using Raspberrypis ip address



Authenticate to VNC Server
192.168.1.6::5900 (TCP)

Enter VNC Server credentials
(Hint: NOT your RealVNC account details)

Username: pi

Password: ••••••••

□ Remember password Forgot password?

Catchphrase: Absent logic Danube. Blonde Darwin plume.

Signature: 26-c0-d7-d0-c4-55-e3-ea

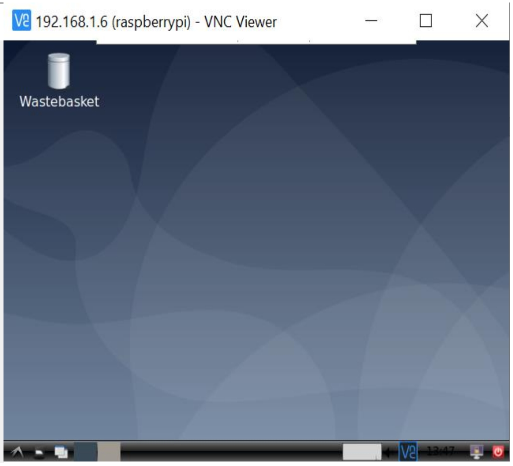
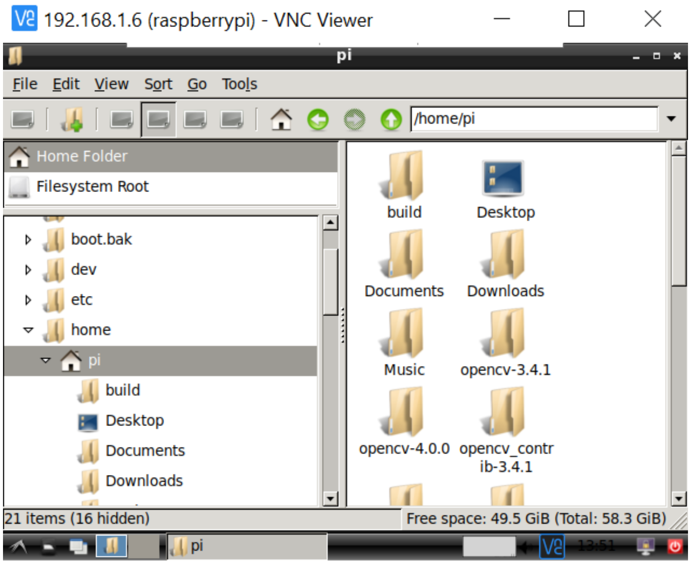


Fig 1.9:Login to VNC viewer

Fig 1.10:Debian desktop,through VNC viewer



21 items (16 hidden)

Free space: 49.5 GiB (Total: 58.3 GiB)

Fig 1.11:pi directory

CAMERA INTERFACING:

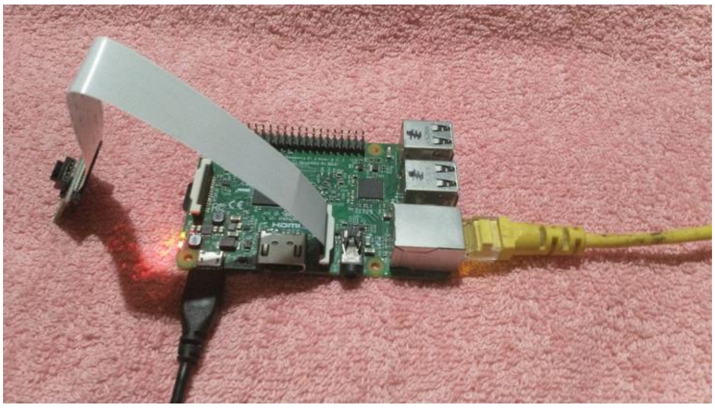


Fig 1.12:Camera interfacing

.Enable camera module in interfaces and Reboot Rasberrypi.

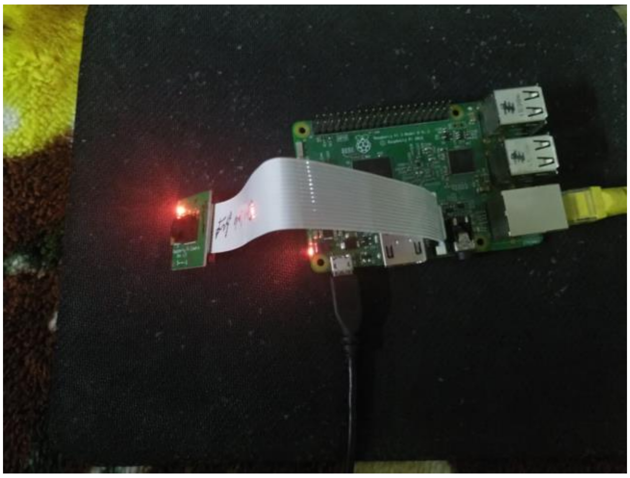
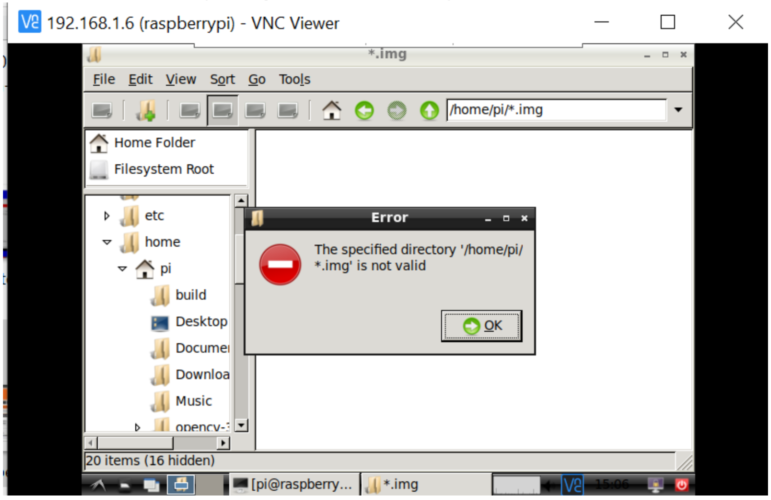


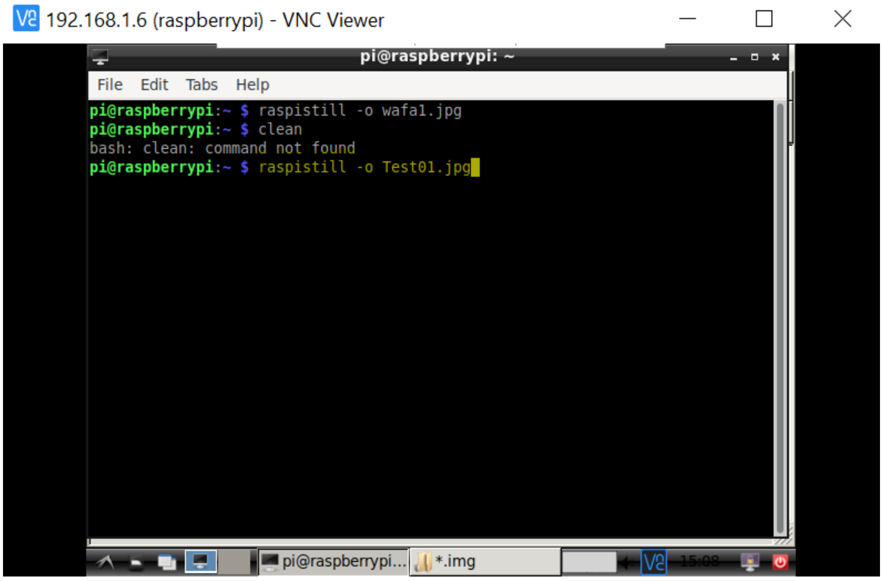
Fig 1.13:Camera Initialization

- Test the camera module by capture an image



The specified directory '/home/pi/*.img' is not valid

Fig 1.14:a.Testing camera module



```
pi@raspberrypi:~ $ raspistill -o wafa1.jpg
pi@raspberrypi:~ $ clean
bash: clean: command not found
pi@raspberrypi:~ $ raspistill -o Test01.jpg
```

Fig 1.14:b.Testing camera module

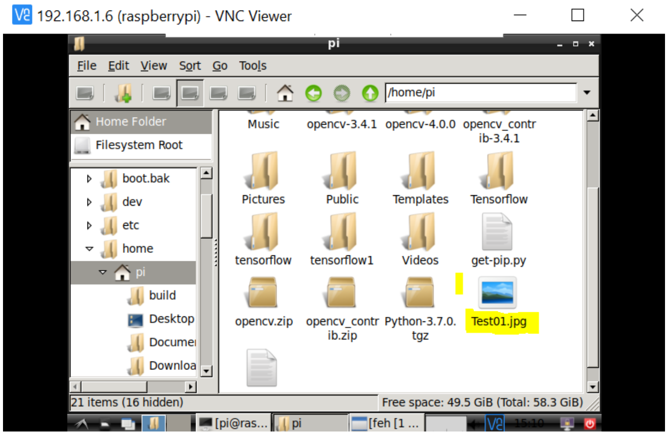


Fig 1.14:c.Testing camera module

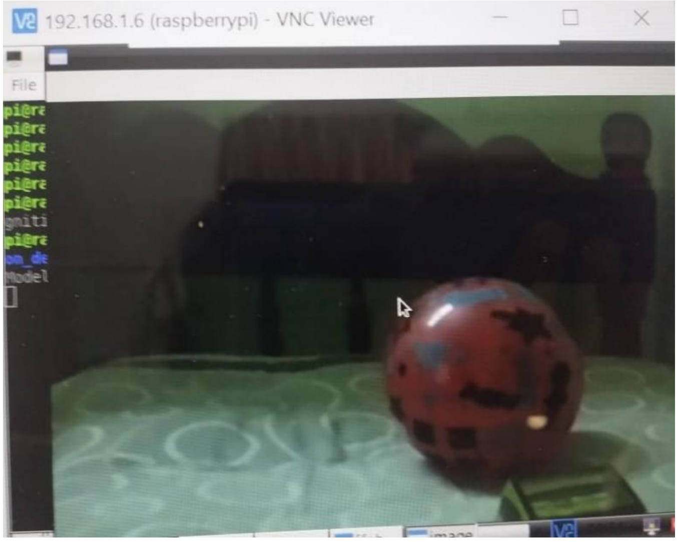


Fig 1.15:Captured image

PART-2

Enhancing with deep learning to identify and differentiate objects using trained Algorithm

- Testing of tensor flow object detection API in laptop using webcam(to capture real time image).

TENSOR FLOW OBJECT DETECTION API:

Dependencies

Tensorflow Object Detection API depends on the following libraries:

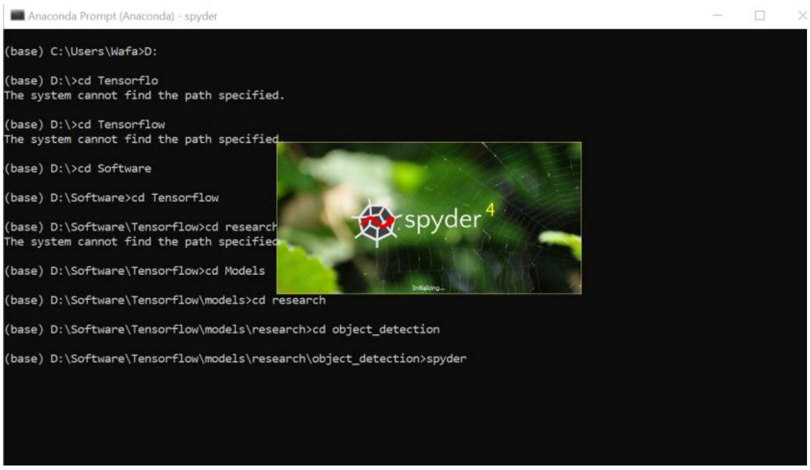
- Protobuf 3.0.0
- Python-tk
- Pillow 1.0
- lxml
- tf Slim (which is included in the "tensorflow/models/research/" checkout)
- Jupyter notebook
- Matplotlib
- Tensorflow (>=1.12.0)
- Cython
- contextlib2

- cocoapi

Protobuf Compilation

The Tensorflow Object Detection API uses Protobufs to configure model and training parameters. Before the framework can be used, the Protobuf libraries must be compiled.

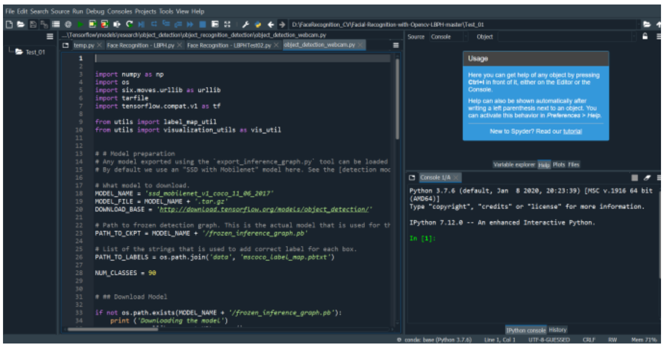
Install Anaconda for Windows, Open Anaconda prompt,Type spyder or jupiter notebook, Create a new project in spyder,and create the .py file,run section by section

- 

```
(base) C:\Users\Wafa>D:

(base) D:\>cd Tensorflo
The system cannot find the path specified.

(base) D:\>cd Tensorflow
The system cannot find the path specified

(base) D:\>cd Software

(base) D:\Software>cd Tensorflow

(base) D:\Software\Tensorflow>cd research
The system cannot find the path specifie

(base) D:\Software\Tensorflow>cd Models

(base) D:\Software\Tensorflow\models>cd research

(base) D:\Software\Tensorflow\models\research>cd object_detection

(base) D:\Software\Tensorflow\models\research\object_detection>spyder
```

. Fig 2.1:Open spyder in Anaconda

Create new project and execute the code



File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\FaceRecognition_CV\Facial-Recognition-with-Opencv-LBPH-master\Test_01
...\Tensorflow\models\research\object_detection\object_recognition_detection\object_detection_webcam.py
temp.py
Face Recognition - LBPH.py
Face Recognition - LBPHTest02.py
object_detection_webcam.py
Test_01
Source
Console
Object
Usage
Variable explorer
Help
Plots
Files
Console 1/A
IPython console
History
conda: base (Python 3.7.6)
Line 1, Col 1
UTF-8-GUESSED
CRLF
RW
Mem 71%

```python

import numpy as np
import os
import six.moves.urllib as urllib
import tarfile
import tensorflow.compat.v1 as tf

from utils import label_map_util
from utils import visualization_utils as vis_util


# # Model preparation
# Any model exported using the `export_inference_graph.py` tool can be loaded
# By default we use an "SSD with Mobilenet" model here. See the [detection moc

# What model to download.
MODEL_NAME = 'ssd_mobilenet_v1_coco_11_06_2017'
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 th
PATH_TO_CKPT = 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('data', 'mscoco_label_map.pbtxt')

NUM_CLASSES = 90


# ## Download Model

if not os.path.exists(MODEL_NAME + '/frozen_inference_graph.pb'):
    print ('Downloading the model')
```

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

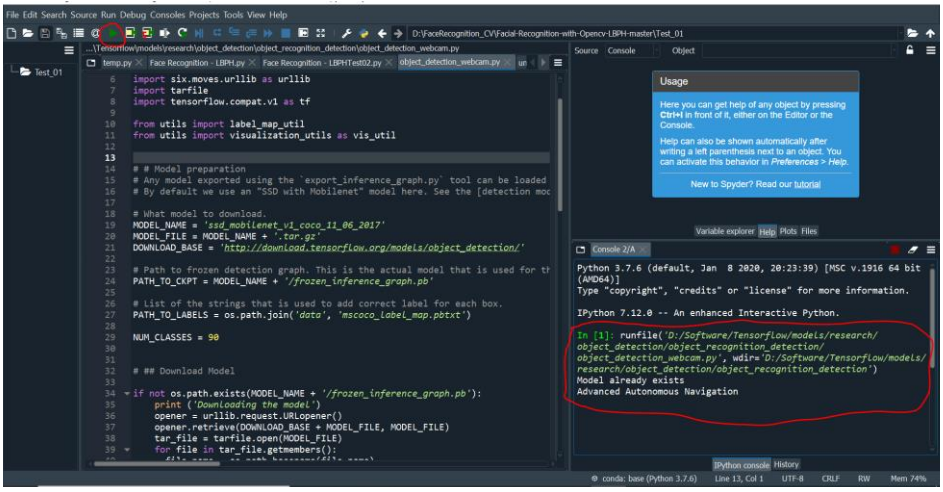
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit
(AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]:
```

Fig 2.2:Create new python project in spyder

Run the file


File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\FaceRecognition_CV\Facial-Recognition-with-Opencv-LBPH-master\Test_01
...\Tensorflow\models\research\object_detection\object_recognition_detection\object_detection_webcam.py
Test_01
temp.py
Face Recognition - LBPH.py
Face Recognition - LBPHTest02.py
object_detection_webcam.py
Source
Console
Object
Variable explorer
Help
Plots
Files
Console 2/A
IPython console
History
conda: base (Python 3.7.6)
Line 13, Col 1
UTF-8
CRLF
RW
Mem 74%

```
import six.moves.urllib as urllib
import tarfile
import tensorflow.compat.v1 as tf

from utils import label_map_util
from utils import visualization_utils as vis_util


# # Model preparation
# Any model exported using the `export_inference_graph.py` tool can be loaded
# By default we use an "SSD with Mobilenet" model here. See the [detection moc

# What model to download.
MODEL_NAME = 'ssd_mobilenet_v1_coco_11_06_2017'
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 th
PATH_TO_CKPT = 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('data', 'mscoco_label_map.pbtxt')

NUM_CLASSES = 90


# ## Download Model

if not os.path.exists(MODEL_NAME + '/frozen_inference_graph.pb'):
    print ('Downloading the 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():
```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit
(AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('D:/Software/Tensorflow/models/research/
object_detection/object_recognition_detection/
object_detection_webcam.py', wdir='D:/Software/Tensorflow/models/
research/object_detection/object_recognition_detection')
Model already exists
Advanced Autonomous Navigation
```

Fig 2.3:Run Object_detection program

- **Object_detection_webcam.py file(Program)**

```
import numpy as np

import os

import six.moves.urllib as urllib

import sys

import tarfile

import tensorflow.compat.v1 as tf

import zipfile

from collections import defaultdict

from io import StringIO

from matplotlib import pyplot as plt

from PIL import Image

 from utils import label_map_util

from utils import visualization_utils as vis_util

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

# By default we use an "SSD with Mobilenet" model here. See the [detection model
zoo](https://github.com/tensorflow/models/blob/master/object_detection/g3doc/detecti
on_model_zoo.md) for a list of other models that can be run out-of-the-box with
varying speeds and accuracies.



 # What model to download.

MODEL_NAME = 'ssd_mobilenet_v1_coco_11_06_2017'

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_CKPT = 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('data', 'mscoco_label_map.pbtxt')

NUM_CLASSES = 90

# ## Download Model

if not os.path.exists(MODEL_NAME + '/frozen_inference_graph.pb'):

            print ('Downloading the 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())

             print ('Download complete')

else:
```

```
                print ('Model already exists')


        # ## Load a (frozen) Tensorflow model into memory.

       detection_graph = tf.Graph()

       with detection_graph.as_default():

       od_graph_def = tf.GraphDef()

       with tf.gfile.GFile(PATH_TO_CKPT, 'rb') as fid:

       serialized_graph = fid.read()

       od_graph_def.ParseFromString(serialized_graph)

       tf.import_graph_def(od_graph_def, name='')

       # ## Loading label map

       # Label maps map indices to category names, so that when our convolution network
predicts `5`, we know that this corresponds to `airplane`.  Here we use internal utility
functions, but anything that returns a dictionary mapping integers to appropriate string labels
would be fine

      label_map = label_map_util.load_labelmap(PATH_TO_LABELS)

      categories = label_map_util.convert_label_map_to_categories(label_map,
max_num_classes=NUM_CLASSES, use_display_name=True)

        category_index = label_map_util.create_category_index(categories)

      #intializing the web camera device

      import cv2

     cap = cv2.VideoCapture(0)

     # Running the tensorflow session

     with detection_graph.as_default():

          with tf.Session(graph=detection_graph) as sess:

          ret = True

          while (ret):

            ret,image_np = cap.read()
```

```
    # Expand dimensions since the model expects images to have shape: [1, None, None, 3]
    image_np_expanded = np.expand_dims(image_np, axis=0)
    image_tensor = detection_graph.get_tensor_by_name('image_tensor:0')
    # Each box represents a part of the image where a particular object was detected.
    boxes = detection_graph.get_tensor_by_name('detection_boxes:0')
    # Each score represent how level of confidence for each of the objects.
    # Score is shown on the result image, together with the class label.
    scores = detection_graph.get_tensor_by_name('detection_scores:0')
    classes = detection_graph.get_tensor_by_name('detection_classes:0')
    num_detections = detection_graph.get_tensor_by_name('num_detections:0')
    # Actual detection.
    (boxes, scores, classes, num_detections) = sess.run(
        [boxes, scores, classes, num_detections],
        feed_dict={image_tensor: image_np_expanded})
    # Visualization of the results of a detection.
    vis_util.visualize_boxes_and_labels_on_image_array(
        image_np,
        np.squeeze(boxes),
        np.squeeze(classes).astype(np.int32),
        np.squeeze(scores),
        category_index,
        use_normalized_coordinates=True,
        line_thickness=8)
#      plt.figure(figsize=IMAGE_SIZE)
#      plt.imshow(image_np)
```

```
cv2.imshow('image',cv2.resize(image_np,(1280,960)))

if cv2.waitKey(25) & 0xFF == ord('q'):

    cv2.destroyAllWindows()

    cap.release()

    break
```

Results(Laptop)

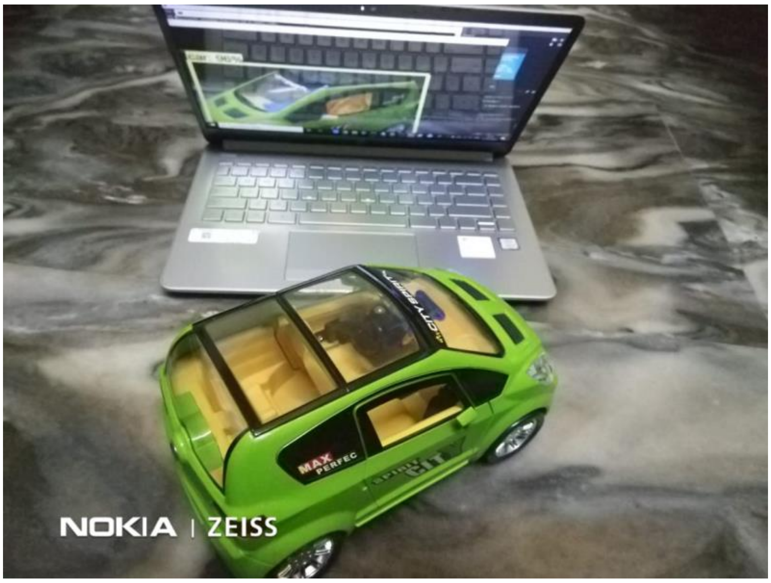


Fig 2.4:Simulation Result laptop(Detecting car)

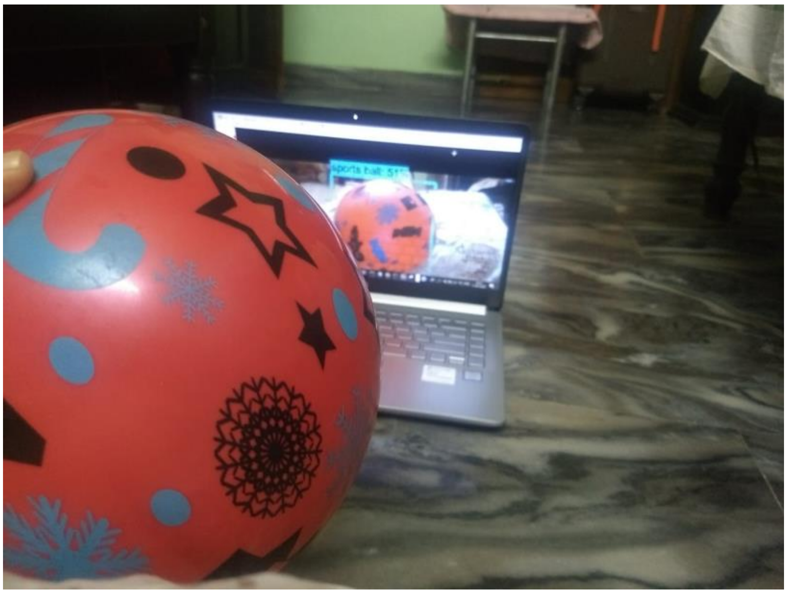
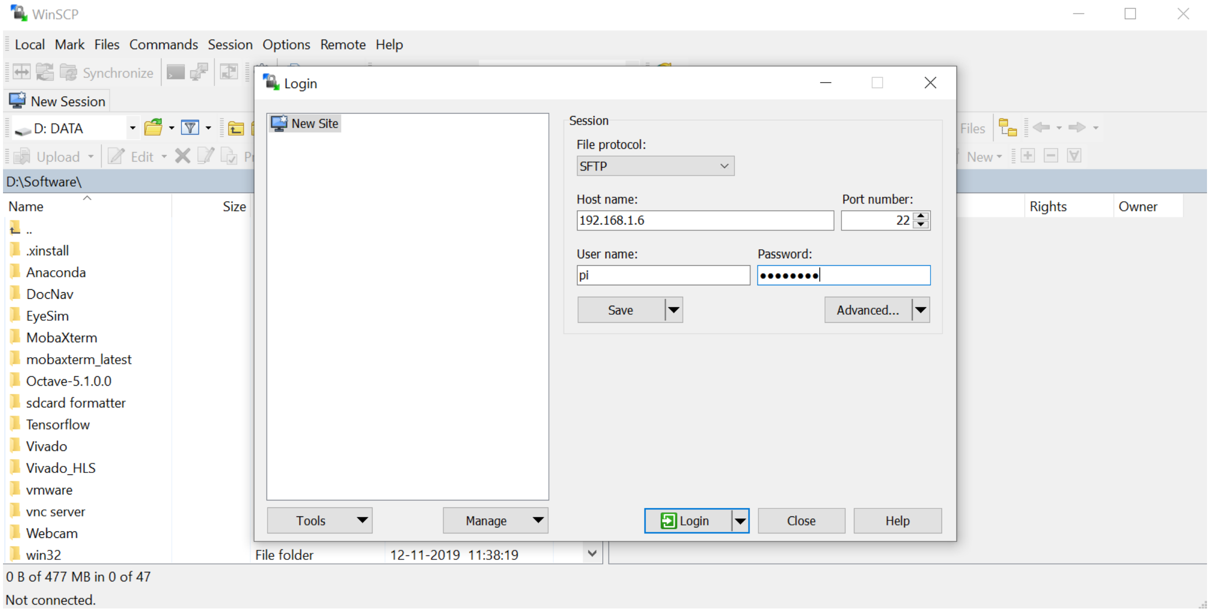


Fig 2.5:Simulation Result laptop(Detecting sports ball)

- **Deployment of the project in to Raspberrypi**
- Use Winscp for file transfer



Session

File protocol: SFTP

Host name: 192.168.1.6

Port number: 22

User name: pi

Password: ••••••••

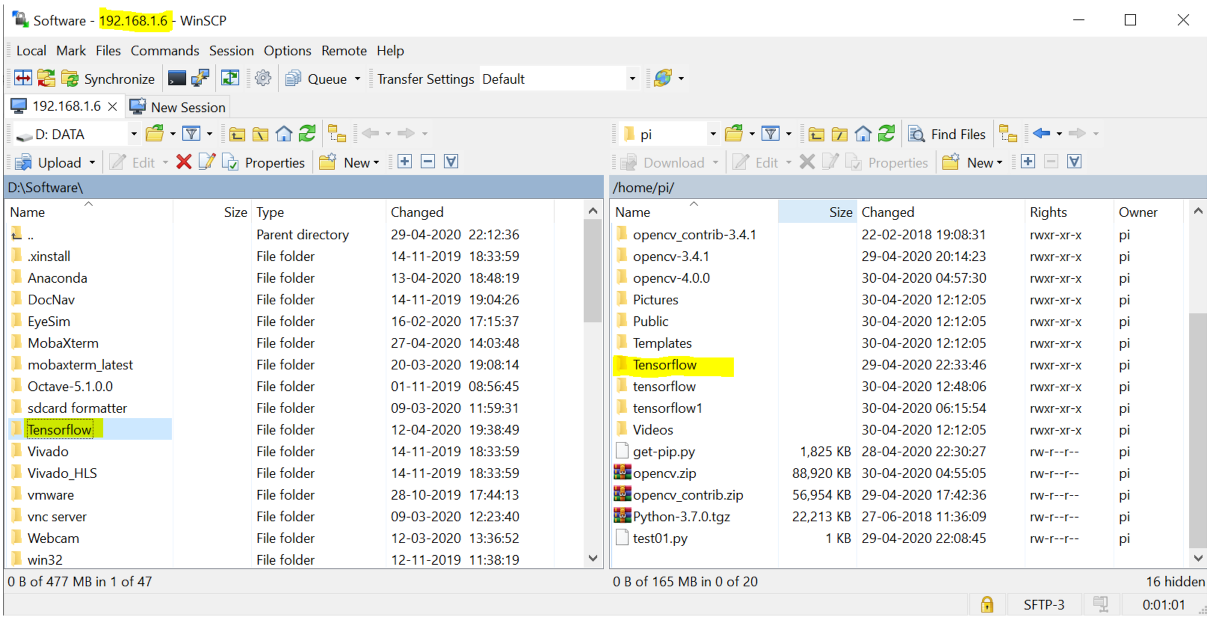
D:\Software\

Name	Size
..	
.xinstall	
Anaconda	
DocNav	
EyeSim	
MobaXterm	
mobaxterm_latest	
Octave-5.1.0.0	
sdcard formatter	
Tensorflow	
Vivado	
Vivado_HLS	
vmware	
vnc server	
Webcam	
win32	File folder 12-11-2019 11:38:19

0 B of 477 MB in 0 of 47

Not connected.

Fig 2.6:Login winscp using RaspberryPi’s ip and credaintails


Software - 192.168.1.6 - WinSCP
Local Mark Files Commands Session Options Remote Help
Synchronize
Queue
Transfer Settings Default
192.168.1.6
New Session
D: DATA
pi
Find Files
Upload Edit Properties New
Download Edit Properties New

D:\Software\

Name	Size	Type	Changed
..		Parent directory	29-04-2020 22:12:36
.xinstall		File folder	14-11-2019 18:33:59
Anaconda		File folder	13-04-2020 18:48:19
DocNav		File folder	14-11-2019 19:04:26
EyeSim		File folder	16-02-2020 17:15:37
MobaXterm		File folder	27-04-2020 14:03:48
mobaxterm_latest		File folder	20-03-2020 19:08:14
Octave-5.1.0.0		File folder	01-11-2019 08:56:45
sdcard formatter		File folder	09-03-2020 11:59:31
Tensorflow		File folder	12-04-2020 19:38:49
Vivado		File folder	14-11-2019 18:33:59
Vivado_HLS		File folder	14-11-2019 18:33:59
vmware		File folder	28-10-2019 17:44:13
vnc server		File folder	09-03-2020 12:23:40
Webcam		File folder	12-03-2020 13:36:52
win32		File folder	12-11-2019 11:38:19

0 B of 477 MB in 1 of 47

/home/pi/

Name	Size	Changed	Rights	Owner
opencv_contrib-3.4.1		22-02-2018 19:08:31	rwxr-xr-x	pi
opencv-3.4.1		29-04-2020 20:14:23	rwxr-xr-x	pi
opencv-4.0.0		30-04-2020 04:57:30	rwxr-xr-x	pi
Pictures		30-04-2020 12:12:05	rwxr-xr-x	pi
Public		30-04-2020 12:12:05	rwxr-xr-x	pi
Templates		30-04-2020 12:12:05	rwxr-xr-x	pi
Tensorflow		29-04-2020 22:33:46	rwxr-xr-x	pi
tensorflow		30-04-2020 12:48:06	rwxr-xr-x	pi
tensorflow1		30-04-2020 06:15:54	rwxr-xr-x	pi
Videos		30-04-2020 12:12:05	rwxr-xr-x	pi
get-pip.py	1,825 KB	28-04-2020 22:30:27	rw-r--r--	pi
opencv.zip	88,920 KB	30-04-2020 04:55:05	rw-r--r--	pi
opencv_contrib.zip	56,954 KB	29-04-2020 17:42:36	rw-r--r--	pi
Python-3.7.0.tgz	22,213 KB	27-06-2018 11:36:09	rw-r--r--	pi
test01.py	1 KB	29-04-2020 22:08:45	rw-r--r--	pi

0 B of 165 MB in 0 of 20

16 hidden

SFTP-3 0:01:01

Fig 2.7:copy files from laptop to Raspberry

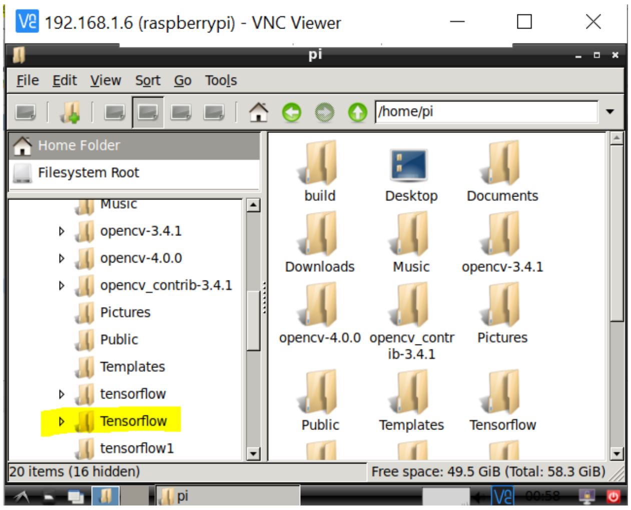
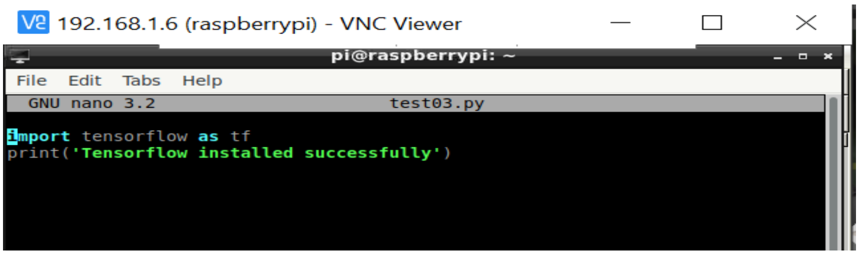
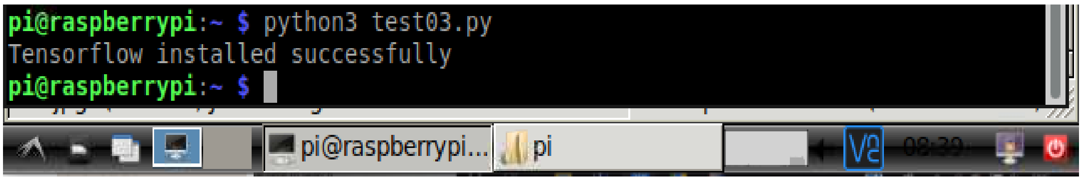


Fig 2.8:Verify files Deployed properlyin Raspberry



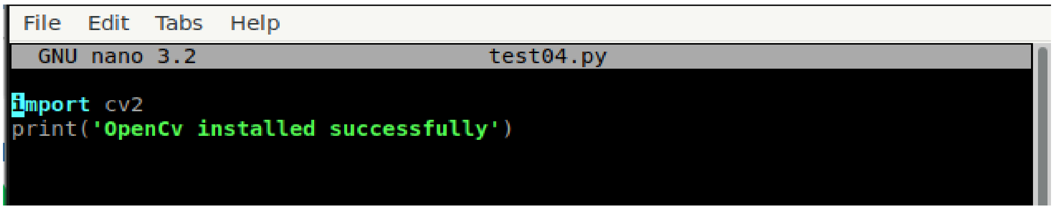
```
import tensorflow as tf
print('Tensorflow installed successfully')
```

Fig 2.9:a.Verify Tensorflow installed properly



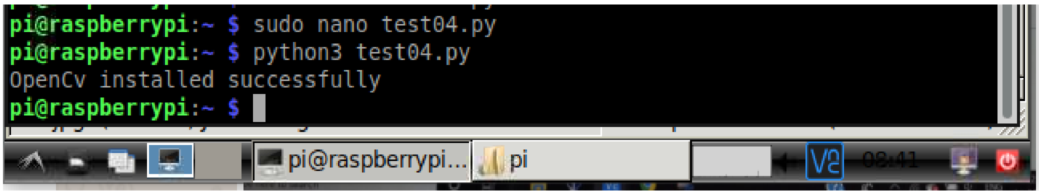
```
pi@raspberrypi:~ $ python3 test03.py
Tensorflow installed successfully
pi@raspberrypi:~ $ 
```

Fig 2.9:b.Verify Tensorflow installed properly



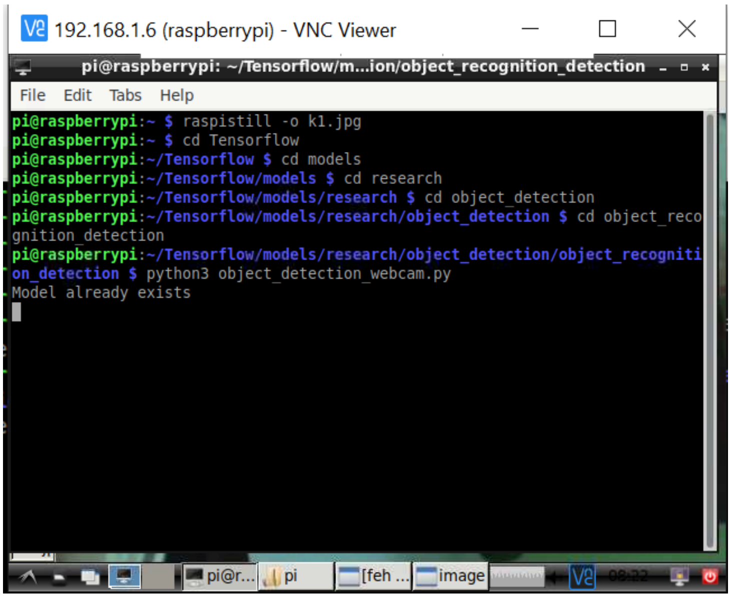
```
import cv2
print('OpenCv installed successfully')
```

Fig 2.10:a.Verify OpenCVinstalled properly



```
pi@raspberrypi:~ $ sudo nano test04.py
pi@raspberrypi:~ $ python3 test04.py
OpenCv installed successfully
pi@raspberrypi:~ $ 
```

Fig 2.10:b.Verify OpenCVinstalled properly



192.168.1.6 (raspberrypi) - VNC Viewer
pi@raspberrypi: ~/Tensorflow/m...ion/object_recognition_detection
File
Edit
Tabs
Help
pi@r...
pi
[feh ...
image

```
pi@raspberrypi:~ $ raspistill -o k1.jpg
pi@raspberrypi:~ $ cd Tensorflow
pi@raspberrypi:~/Tensorflow $ cd models
pi@raspberrypi:~/Tensorflow/models $ cd research
pi@raspberrypi:~/Tensorflow/models/research $ cd object_detection
pi@raspberrypi:~/Tensorflow/models/research/object_detection $ cd object_reco
gnition_detection
pi@raspberrypi:~/Tensorflow/models/research/object_detection/object_recogniti
on_detection $ python3 object_detection_webcam.py
Model already exists
```

Fig 2.11:Execute Object_detection program

RESULTS(RASPBERRY PI)

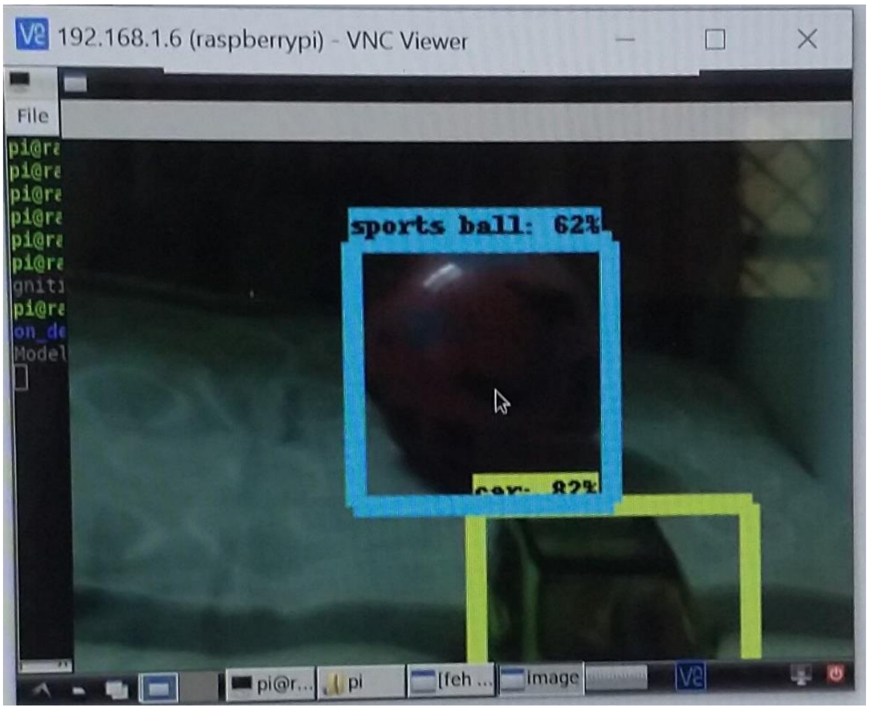


Fig 2.12:Simulation result(Raspberry)-1

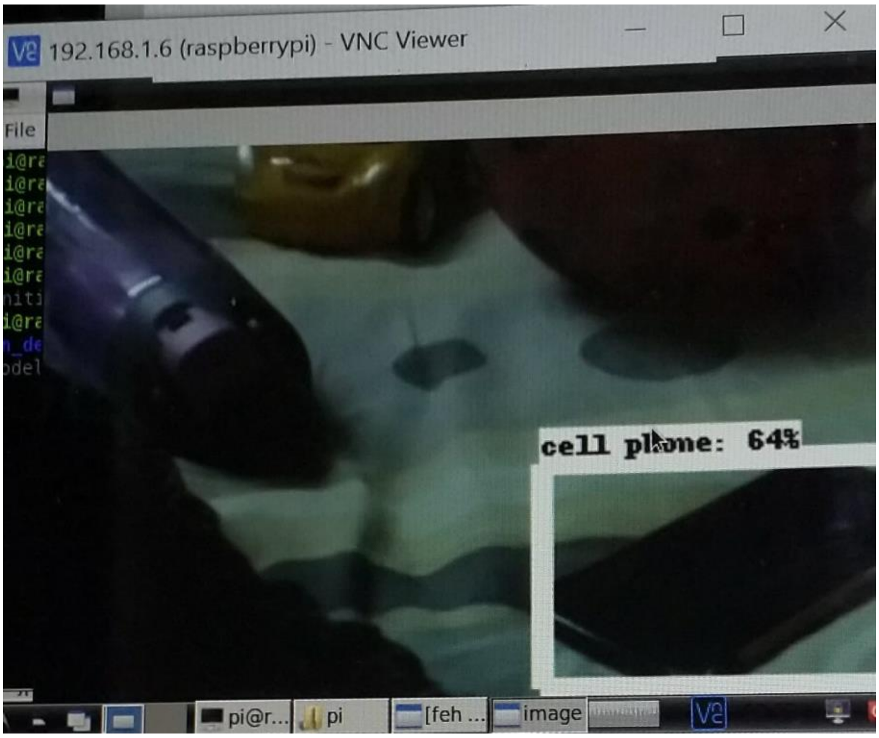


Fig 2.13:Simulation result(Raspberry)-2

CONCLUSION

- Done set up of Rasberrypi and interfaced camera module
- Achieved Object detection in laptop , deployed in Raspberrypi and Tested.

References:

- https://ieeexplore.ieee.org/document/8494053
- https://ieeexplore.ieee.org/document/9070240
- https://ieeexplore.ieee.org/document/8452728
- https://ieeexplore.ieee.org/abstract/document/7995703
- https://www.datacamp.com/community/tutorials/tensorflow-tutorial
- https://projects.raspberrypi.org/en/projects/getting-started-with-picamera/5
- https://www.raspberrypi.org/documentation/remote-access/vnc/
- Install Rasbian Buster - https://www.youtube.com/watch?v=AuvbbP7Dues
- https://www.pyimagesearch.com/2016/08/29/common-errors-using-the-raspberry-pi-camera-module/
- https://thepi.io/how-to-use-your-raspberry-pi-as-a-wireless-access-point/
- OpenCV Installation - https://pysource.com/2018/10/31/raspberry-pi-3-and-opencv-3-installation-tutorial/
- https://www.analyticsvidhya.com/blog/2016/10/an-introduction-to-implementing-neural-networks-using-tensorflow/
- https://www.youtube.com/watch?v=MoMjIwGSFVQ
- https://www.analyticsvidhya.com/blog/2018/06/understanding-building-object-detection-model-python/
- NOOBS Installation - https://www.youtube.com/watch?v=wvxCNQ5AYPg
- Static IP setup - https://www.youtube.com/watch?v=3VAlvtFWOwE