

# Object Tracking Robot Using Raspberry Pi 4

This file contains the *files and folder* list included with the .zip file and the installation instructions to successfully run the program without any package dependency error.

## Files and Folders:

- Documents : This folder *contains* the project report document in .word and .pdf format.
- Simulation : This folder *contains* the files, results and videos recorded during the simulation procedure ; contains three experiment results : One with single Goal Object ; Second with two Goal object; Third with three Goal object named *Simulation.m4v*.
- Images : This folder contains the images used in the report.
- Codes : This folder contains the individual chunks of code written to test different stages of the project.(written for python==3.6.4)
- Robot : This folder contains the final code used for the simulation purpose named *Robot.py* .All the dependencies and installation files are present inside this directory.
- Plagarism : This folder contains the plagiarism report and log file.

## Running the Object Tracking Code :

### Step 1:

The first step is to install virtual environment , to prevents any conflicts between versions of package libraries that may already be installed on your Pi. Keeping it installed in its own environment allows us to avoid this problem. For example, if you've already installed TensorFlow v1.8 on the Pi , you can leave that installation as-is without having to worry about overriding it.

Install virtualenv by issuing:

```
$ sudo pip3 install virtualenv
```

Then, create the "tflite1-env" virtual environment by issuing:

```
$ python3 -m venv tflite1-env
```

This will create a folder called tflite1-env inside the tflite1 directory. The tflite1-env folder will hold all the package libraries for this environment. Next, activate the environment by issuing:

```
$ source tflite1-env/bin/activate
```

You'll need to issue the source tflite1-env/bin/activate command from inside the /home/pi/tflite1 directory to reactivate the environment every time you open a new terminal window. You can tell when the environment is active by checking if (tflite1-env) appears before the path in your command prompt. Next, we'll install TensorFlow, OpenCV, and all the dependencies needed for both packages. OpenCV is not needed to run TensorFlow Lite, but the object detection scripts in this repository use it to grab images and draw detection results on them.

To make things easier, we wrote a shell script that will automatically download and install all the packages and dependencies. Run it by issuing:

```
$ bash get_pi_requirements.sh
```

Run the real-time webcam detection script by issuing the following command from inside the `/home/pi/tflite1` directory. (Before running the command, make sure the `tflite1-env` environment is active by checking that `(tflite1-env)` appears in front of the command prompt.) The `Robot.py` script will work with either a Picamera or a USB webcam.

```
$ python3 Robot.py --modeldir=Sample_Model_dir
```

The `Sample_Model_dir` contains the model files `.tflite` and `labelmap.txt` inside it. Then you should see the frame opening up showing the video stream taken by camera and the object detection and tracking commands given by raspberry pi being printed in command line.