Set up Ros Nodes for Raspberry Pi Camera On Ubuntu 18.04.5

Version 1.0

Author: Nikolai Tiong

Date: 2/2/21

This guide is for setting up the Raspberry Pi Camera, publishing the camera video stream as a topic, then subscribing to it from your computer.

Assumptions:

Raspberry Pi

* + Ubuntu 18.04.5 **32-bit (armhf) (THIS IS IMPORTANT)** is the operating system on the Raspberry Pi
  + ROS Melodic is installed on the Raspberry Pi
  + You know the IP address of the Pi, or the MAC address and can use it to find the IP address
  + SSH is enabled
  + VNC Server is setup and you can remote in via VNC Viewer
  + The Pi Camera is installed with a lens and connected to the Pi

Your Computer

* + Your computer is running Ubuntu 18.04 – 64-bit is OK on your computer
  + ROS Melodic is installed on your computer
  + You can SSH in via a terminal
  + You are using a computer connected to the same network as the Raspberry Pi
  + You know your way around Linux

# Installing the Drivers for the Raspberry Pi

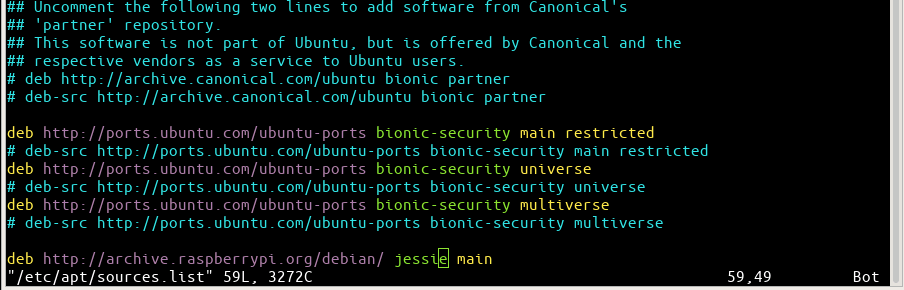
VNC into the Raspberry Pi

Open a terminal and modify the sources.list file:

sudo vim /etc/apt/sources.list

Add the following to the bottom:

deb <http://archive.raspberrypi.org/debian/jessie> main



Save the file, download the keys, run update, upgrade and install raspi-config:

sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 7FA3303E

sudo apt update

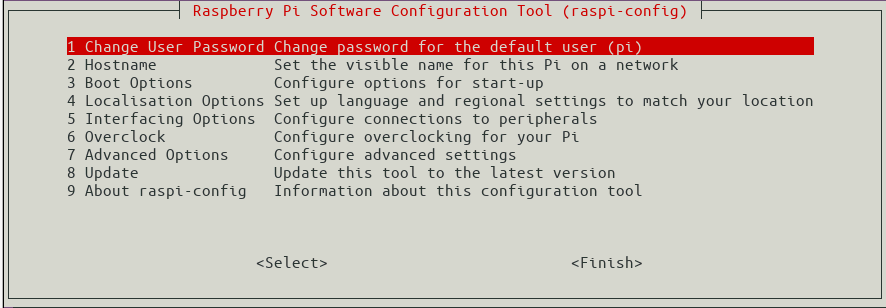
sudo apt upgrade

sudo apt-get install raspi-config

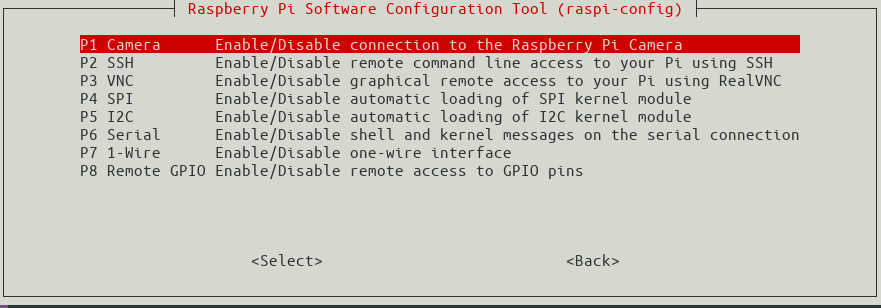
Mount the boot partition and run raspi-config

sudo mount /dev/mmcblk0p1 /boot

sudo raspi-config



Go to Interfacing Options and enable the Camera



Use the left/right arrow keys to select “<Finish>” when done.

**WARNING: IF YOU ARE USING UBUNTU 64-BIT AND TRY TO INSTALL LIBRASPBERRYPI-DEV, YOU WILL KILL THE OS. THE COMMAND WON’T WORK WITH THE JESSIE DISTRIBUTION, BUT WILL WITH BUSTER IF YOU ATTEMPT IT. TRUST ME, I TRIED TWICE.**

Install the camera libraries

sudo apt-get install libraspberrypi-dev

sudo apt update

sudo apt install python-pip

sudo pip install picamera

sudo reboot

# Test the Camera

Create a python script – cam.py

import picamera  
camera = picamera.PiCamera()  
camera.capture(‘mytest\_image.jpg’)  
camera.close()

Run it and check that the jpg is created and corresponds to what the camera is pointing at – it may be out of focus since there is no real time feedback.

# Setup the ROS Node

Clone the following GitHub repository:

cd ~/catkin\_ws/src

git clone <https://github.com/dganbold/raspicam_node.git>

cd ~/catkin\_ws/

catkin\_make

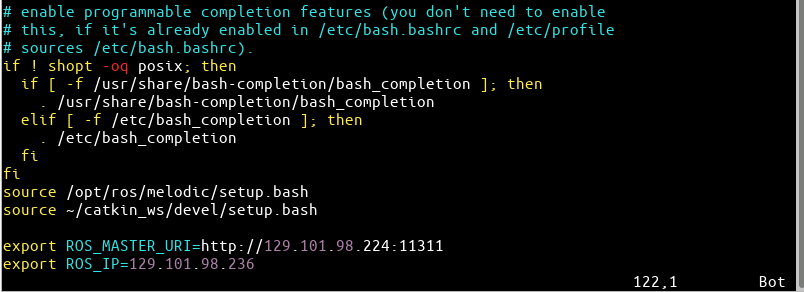
Your computer will be setup as the master node and the Pi needs to point to your computer.

vim ~/.bashrc

At the bottom of the file add the following:

export ROS\_MASTER\_URI=http://<Your IP Address>:11311

export ROS\_IP=<Raspberry Pi IP Address>

****

Save and close it then type in the terminal:

source ~/.bashrc

Go to a terminal on your computer and run roscore.

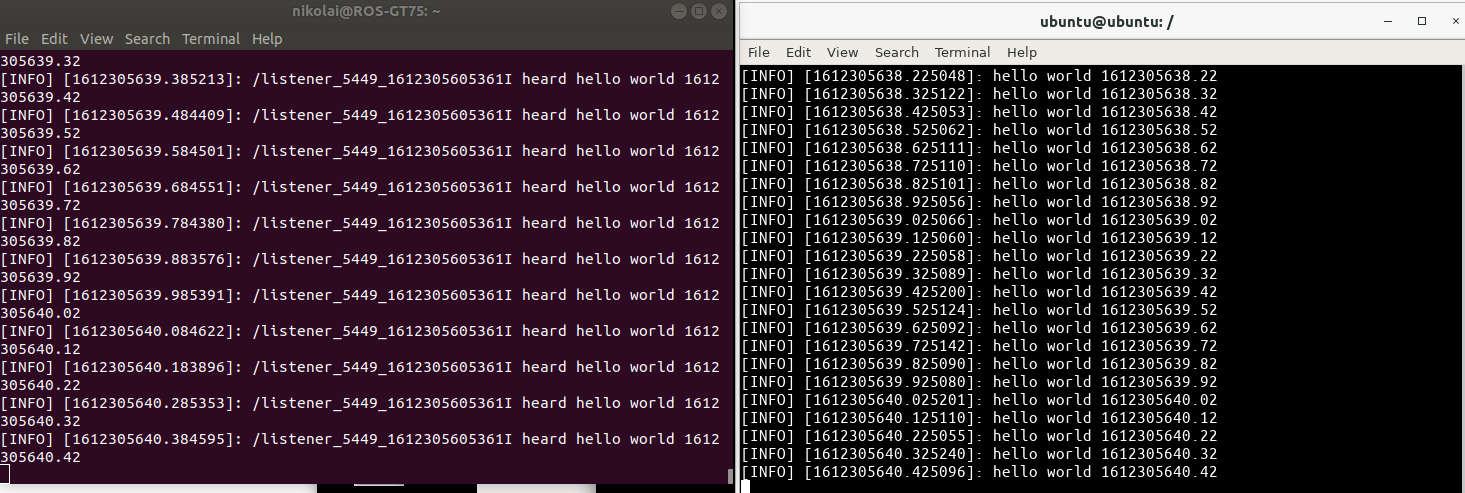
## Verify that ROS Publishing and Subscribing works

On one side, run the listener tutorial and on the other run the talker.

rosrun rospy\_tutorials talker.py

rosrun rospy\_tutorials listener.py

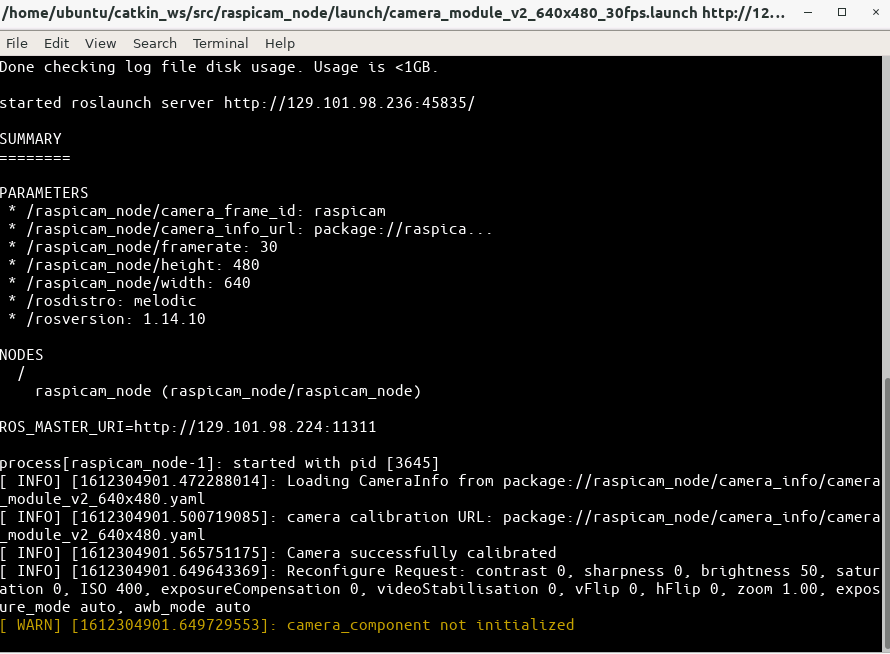
If everything is setup correctly, the subscriber will receive the messages sent from the publisher side.



## Start the Camera Publisher

Go back to the VNC Session and start the camera publisher:

roslaunch raspicam\_node camera\_module\_v2\_640x480\_30fps.launch



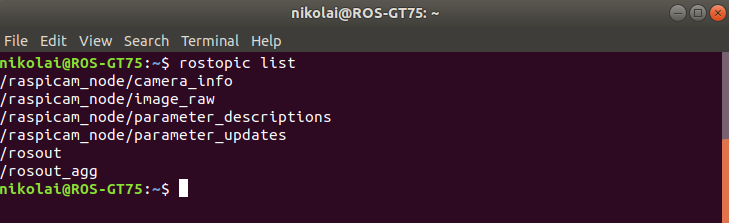
Open a new terminal and initialize the camera

rosservice call /raspicam\_node/start\_capture

# 

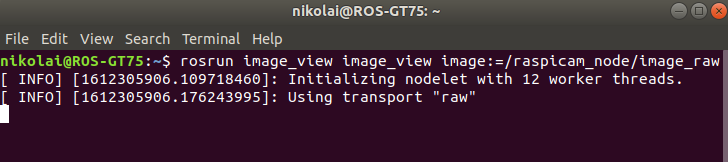
Go back to your computer terminal and check that the topics for the camera appear:

rostopic list

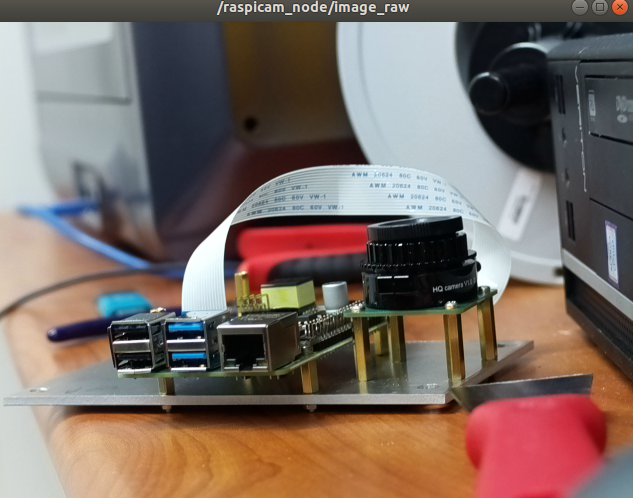


Subscribe to the image\_raw topic

rosrun image\_view image\_view image:=/raspicam\_node/image\_raw



Another window will open with the camera stream. The window may need to be expanded. Adjust the focus and iris on the camera to verify that the video looks correct.



If your computer is on WiFi, the video will have about a 1 second delay (haven’t solved this issue).

Connect to the network via a wired connection for best results.

# Running a Higher Resolution Video

The Github repository used only has 640x480 calibration settings.

Here are calibration settings for 1280x720 resolution.

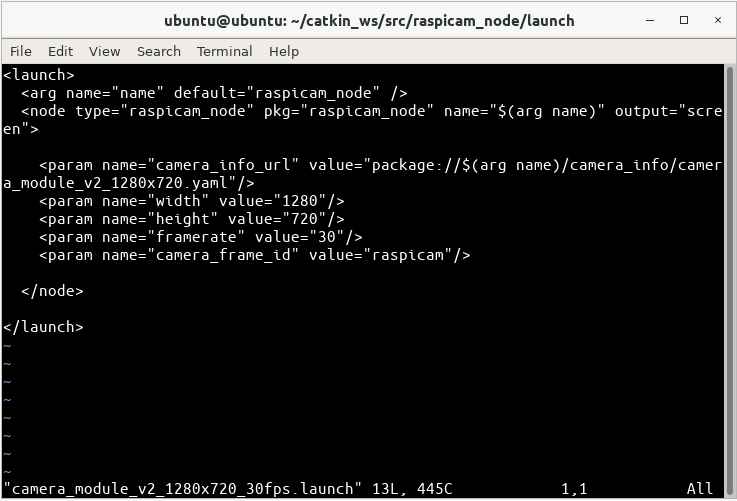
Navigate to where the launch files are located:

cd ~/catkin\_ws/src/raspicam\_node/launch

List the files using ls -l and copy the camera\_module\_v2\_640x480\_30fps.launch file

cp camera\_module\_v2\_640x480\_30fps.launch camera\_module\_v2\_1280x720\_30fps.launch

Open the 1280x720 file copied and change the camera\_info\_url from 640x480 to 1280x720, width from 640 to 1280 and height from 480 to 720.



Save the file and then go to the camera\_info folder and do an ls -l

cd ~/catkin\_ws/src/raspicam\_node/camera\_info

Copy the camera module .yaml file in the same manner as the .launch file

cp camera\_module\_v2\_640x480.yaml camera\_module\_v2\_1280x720.yaml

Open the new 1280x720 yaml file and replace the text with the following:

image\_width: 1280

image\_height: 720

camera\_name: camerav2\_1280x720

camera\_matrix:

rows: 3

cols: 3

data: [1276.704618338571, 0, 634.8876509199106, 0, 1274.342831275509, 379.8318028940378, 0, 0, 1]

distortion\_model: plumb\_bob

distortion\_coefficients:

rows: 1

cols: 5

data: [0.1465167016954302, -0.2847343180128725, 0.00134017721235817, -0.004309553450829512, 0]

rectification\_matrix:

rows: 3

cols: 3

data: [1, 0, 0, 0, 1, 0, 0, 0, 1]

projection\_matrix:

rows: 3

cols: 4

data: [1300.127197265625, 0, 630.215390285608, 0, 0, 1300.670166015625, 380.1702884455881, 0, 0, 0, 1, 0]

Save then start the camera publisher with the 1280x720 launch file.

roslaunch raspicam\_node camera\_module\_v2\_1280x720\_30fps.launch

Initialize the camera

rosservice call /raspicam\_node/start\_capture

Subscribe to the camera

rosrun image\_view image\_view image:=/raspicam\_node/image\_raw

The video will now be at 1280x720 resolution.



The above can be repeated for 1920x1080 resolution as well. I haven’t found a 1080p calibration file but using the 720p version for it seems to work.