

# ROS Interface for PiRobot

# Requirements

- Raspberry Pi 3B - Two numbers (one for the robot & one for remote host)
- Pi Robot Kit assembled
- Raspbian Buster OS
- ROS Noetic
- Python 3.x
- Opencv 4.x
- Picamera V2

# Steps for both Robot & Remote Host

These steps are to be executed on both Robot and Remote Host:

1. Install ROS Noetic on Raspbian Buster  
<https://varhowto.com/install-ros-noetic-raspberry-pi-4/>
2. Clone this github repo in your home directory:  
`$ git clone https://github.com/swagatk/Raspi\_codes.git`

It will create a folder called 'Raspi\_codes' in your home directory.

3. Create catkin workspace in your home folder:  
`$ mkdir ~/catkin_ws/  
$ cd ~/catkin_ws/  
$ catkin_make  
$ source devel/setup.bash`

4. Create a catkin package 'pirobot' inside your catkin\_ws folder:  
`$ cd catkin_ws/src/  
$ catkin_create_pkg pirobot rospy roscpp std_msgs  
$ cd ~/catkin_ws/  
$ catkin_make  
$ source devel/setup.bash`

5. Copy the files from GIT repo to the pirobot package:  
`$ roscd pirobot  
$ mkdir scripts  
$ cp ~/Raspi_codes/keyboard_teleop/*.py ./scripts/  
$ cd ~/catkin_ws/  
$ catkin_make  
$ source ./devel/setup.bash`

This will make sure that the new files are accessible through ROS commands.

# Setup on Remote Host

1. On the Remote host, enter the following lines inside the ~/.bashrc file

```
export ROS_MASTER_URI=http://192.168.1.144:11311  
export ROS_IP=192.168.1.144
```

Replace the IP with your own IP address. Run the command 'ifconfig' to find the IP address of your raspberry pi.

2. Open a new terminal and run the following commands:

```
$ bash  
$ roscore
```

Check that the ROS\_MASTER\_URI address should as set above.

## Setup on Remote Host contd ...

3. Open another terminal and run the following command:

```
$ source ~/catkin_ws/devel/setup.bash  
$ rosrun pirobot kb_teleop.py
```

It will run a publisher to catch the key press on the remote host. You can use the arrow keys to move the robot left, right, forward and backward.

# Setup on Robot

1. On the robot system, please enter the following lines to `~/.bashrc` file:

```
export ROS_MASTER_URI=http://<previous_master_ip>:11311  
export ROS_IP=192.168.1.145 (its own ip)
```

2. Open a terminal and run the following commands:

```
$ bash  
$ source ~/catkin_ws/devel/setup.bash  
$ rosrun pirobot robot_motion.py
```

It runs a subscriber to communicate with the publisher running on the remote host.

3. Now you can press the arrow keys on remote host to control the motion of the pi robot. Make sure that both robot and remote host are connected to the same network. Also make sure that you don't get any error message while executing above steps.

# Remote System

```
pi@raspberrypi: ~/catkin_ws
File Edit Tabs Help
pi@rasp... x pi@rasp... x
[INFO] [1699875789.277441]: STOP
[INFO] [1699875789.380416]: STOP
[INFO] [1699875789.477377]: STOP
[INFO] [1699875789.577416]: STOP
[INFO] [1699875789.677592]: STOP
[INFO] [1699875789.780473]: STOP
[INFO] [1699875789.877635]: STOP
[INFO] [1699875789.977540]: STOP
[INFO] [1699875790.077502]: STOP
[INFO] [1699875790.180855]: STOP
[INFO] [1699875790.278417]: STOP
[INFO] [1699875790.377704]: STOP
[INFO] [1699875790.476318]: STOP
[INFO] [1699875790.576219]: STOP
[INFO] [1699875790.677927]: STOP
[INFO] [1699875790.778121]: STOP
[INFO] [1699875790.878257]: STOP
[INFO] [1699875790.978084]: STOP
[INFO] [1699875791.077937]: STOP
[INFO] [1699875791.177915]: STOP
[INFO] [1699875791.277985]: STOP
[

roscore http://192.168.1.109:11311/
File Edit Tabs Help
started roslaunch server http://192.168.1.109:32969/
ros_comm version 1.15.15

SUMMARY
=====

PARAMETERS
* /roscpp: noetic
* /rosversion: 1.15.15

NODES

auto-starting new master
process[roscpp]: started with pid [1385]
ROS_MASTER_URI=http://192.168.1.109:11311/

setting /run_id to 3a215f34-8217-11ee-bf6e-e45f01013852
process[roscpp-1]: started with pid [1399]
started core service [/roscpp]
```

# PiRobot

```
pi@raspberrypi: ~/catkin_ws
File Edit Tabs Help
[INFO] [1699875933.493251]: /listener_1416_1699875681266: STOP
[INFO] [1699875933.589797]: /listener_1416_1699875681266: STOP
[INFO] [1699875933.689986]: /listener_1416_1699875681266: STOP
[INFO] [1699875933.790842]: /listener_1416_1699875681266: STOP
[INFO] [1699875933.890113]: /listener_1416_1699875681266: STOP
[INFO] [1699875933.991189]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.105223]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.190724]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.290031]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.393705]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.493597]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.590028]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.689998]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.790870]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.889860]: /listener_1416_1699875681266: STOP
[INFO] [1699875934.990275]: /listener_1416_1699875681266: STOP
[INFO] [1699875935.089812]: /listener_1416_1699875681266: STOP
```



# ROS Image Publisher & Subscriber with Picamera

# Install Opencv 4.5.x on Raspberry Pi

- **For RPi 4**, follow instruction available at this site:  
<https://qengineering.eu/install-opencv-4.5-on-raspberry-pi-4.html>
- **For RPi 3**, we need to install the available debian package  
`$ sudo apt-get install python-opencv python3-opencv`
- Make sure that the cv2 is accessible from python console

```
$ python3
Python 3.7.3 (default, Oct 31 2022, 14:04:00)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> cv2.__version__
'4.5.5'
>>>
```

# Install CV\_Bridge into ROS\_COMM package

Install cv\_bridge package in your ROS Noetic (This may not be required)

```
$ cd ~/ros_catkin_ws/  
  
$ rosinstall_generator cv_bridge sensor_msgs --rosdistro noetic --deps --wet-only --tar > noetic-cv_bridge-wet.rosinstall  
  
$ wstool merge noetic-cv_bridge-wet.rosinstall -t src  
  
$ wstool update -t src  
  
$ rosdep install -y --from-paths src --ignore-src --rosdistro noetic -r --os=debian:buster  
  
$ sudo src/catkin/bin/catkin_make_isolated --install -DCMAKE_BUILD_TYPE=Release --install-space /opt/ros/noetic -j1  
-DPYTHON_EXECUTABLE=/usr/bin/python3
```

Also note that CV\_Bridge Does not work with Python 3.x and hence will not work with ROS Noetic. So, we will use a method that does not require this package at all.

# Enable/Install PiCamera

- Follow the instructions for **Raspbian Buster** available the following link:  
<https://projects.raspberrypi.org/en/projects/getting-started-with-picamera>
- Write a simple python script to test if the camera module is working:

```
from picamera import PiCamera  
from time import sleep
```

```
camera = PiCamera()  
camera.start_preview()  
sleep(5)  
camera.stop_preview()
```

# Create a New Catkin Package

```
$ cd ~/catkin_ws/src
```

```
$ catkin_create_pkg picam rospy cv_bridge sensor_msgs (# cv_bridge may not be required)
```

```
$ cd ~/catkin_ws
```

```
$ mkdir -p src/picam/scripts
```

```
$ mkdir -p src/picam/launch
```

Now copy the contents of 'Raspi\_codes/ROS/picam' to the scripts & launch folders:

```
$ cp ~/Raspi_codes/ROS/picam/*.py ./src/picam/scripts/
```

```
$ cp ~/Raspi_codes/ROS/picam/picam.launch ./src/picam/launch/
```

```
$ catkin_make
```

```
$ source devel/setup.bash
```

# Run publisher & Subscriber

- Execute the following command in one terminal for publisher:  
`$ roslaunch pirobot picam.launch`
- Execute the following command in the second terminal for subscriber:  
`$ rosrun pirobot image_subscriber.py`
- Explore rosnodes & rostopics:  
`$ rosnode list`  
`/imagepub`  
`/imagesub`  
`/rosout`  
  
`$ rostopic list`  
`/picam/camera`  
`/rosout`  
`/rosout_agg`

# Errors Debugging

If you get error with OpenCV 4.5.5 and Python3, you may edit the corresponding file and removing the last argument in that line as well.

```
$ sudoedit /opt/ros/noetic/lib/python3/dist-packages/rospy/impl/tcpros_base.py
```

Change the line no. 167 to the following:

```
(e_errno, msg) = e.args
```

```
$ rosrn pirobot image_subscriber.py
Traceback (most recent call last):
  File "/home/pi/catkin_ws/src/pirobot/scripts/image_subscriber.py", line 4, in <module>
    import rospy
  File "/opt/ros/noetic/lib/python3/dist-packages/rospy/__init__.py", line 49, in <module>
    from .client import spin, myargv, init_node, \
  File "/opt/ros/noetic/lib/python3/dist-packages/rospy/client.py", line 60, in <module>
    import rospy.impl.init
  File "/opt/ros/noetic/lib/python3/dist-packages/rospy/impl/init.py", line 54, in <module>
    from .tcpros import init_tcpros
  File "/opt/ros/noetic/lib/python3/dist-packages/rospy/impl/tcpros.py", line 45, in <module>
    import rospy.impl.tcpros_service
  File "/opt/ros/noetic/lib/python3/dist-packages/rospy/impl/tcpros_service.py", line 54, in <module>
    from rospy.impl.tcpros_base import TCPROSTransport, TCPROSTransportProtocol, \
  File "/opt/ros/noetic/lib/python3/dist-packages/rospy/impl/tcpros_base.py", line 167
    (e_errno, msg, *) = e.args
                    ^
```

# Installing Noetic-Desktop on RPi 4

- Instruction is available here:  
<https://www.hackster.io/shahizat/lidar-integration-with-ros-noetic-on-raspberry-pi-os-8ea140>
- This source is also useful: <https://wiki.ros.org/noetic/Installation/Source>



# Installing ORB\_SLAM3 on RPi 4 ROS Noetic

- Follow the instructions provided at this link:

[https://habr-com.translate.google.ru/post/689168/?\\_x\\_tr\\_sl=ru&\\_x\\_tr\\_tl=en&\\_x\\_tr\\_hl=en&\\_x\\_tr\\_pto=sc](https://habr-com.translate.google.ru/post/689168/?_x_tr_sl=ru&_x_tr_tl=en&_x_tr_hl=en&_x_tr_pto=sc)

- Incomplete !!

# MPU 6050

$$x_t = x_0 + v_x t + \frac{1}{2} a_x t^2$$

$$v_x^t = v_x^0 + a_x t$$

$$x = x_0 + vt \cos(\theta)$$

$$y = y_0 + vt \sin(\theta)$$

$$\dot{x} = v \cos(\theta)$$

$$\dot{y} = v \sin(\theta)$$

$$v = \sqrt{v_x^2 + v_y^2}$$

$$\dot{v} = \frac{1}{\sqrt{v_x^2 + v_y^2}} (v_x \dot{v}_x + v_y \dot{v}_y) = \frac{(v_x a_x + v_y a_y)}{v}$$

$$\dot{v}_x = a_x$$

$$\dot{v}_y = a_y$$