# Running Raspberry Pi Cameras on ROSBOX2

Author: Nikolai Tiong

Date: 5/17/21

Version 1.0

# Overview

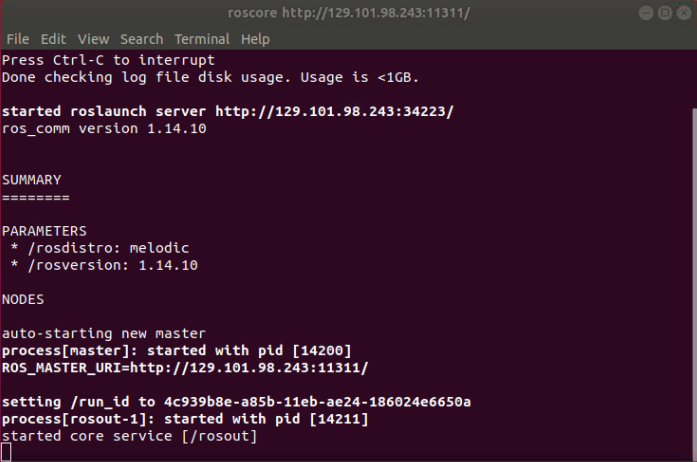
These are instructions for how to get the Raspberry Pi Cameras running.

It consists of starting up roscore on ROSBOX2, starting up the node on each Raspberry Pi Camera and starting up the websocket, webserver and video streaming server on ROSBOX2.

# Start Roscore

VNC into ROSBOX2, open a terminal window and make sure roscore is running. If it isn’t start it up.

roscore



# PiCamera

VNC into the Raspberry Pi e.g. front, above, arm and open a terminal window.

Go to

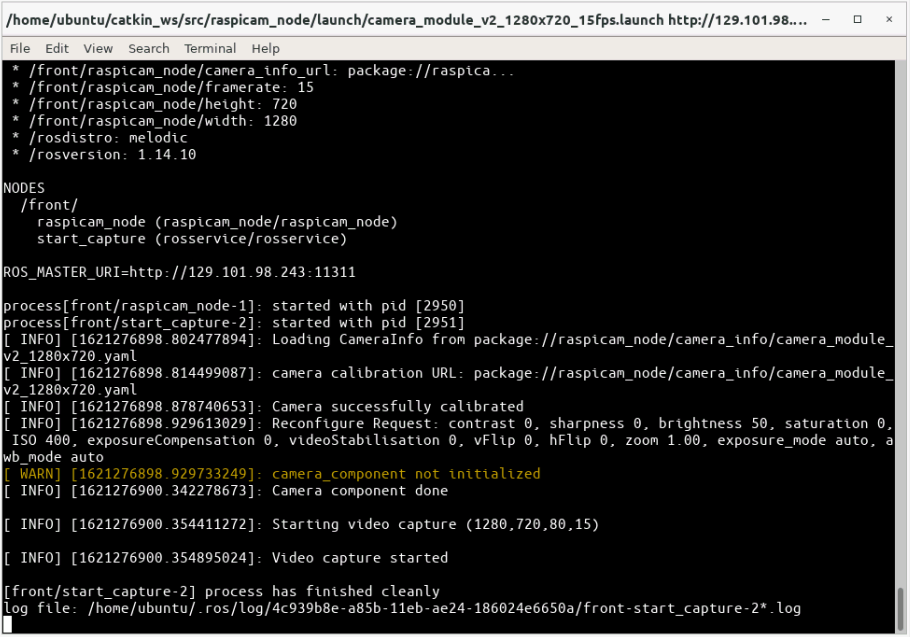
cd ~/catkin\_ws/src/camera\_scripts

Run the cam.py script

python cam.py

It should start the camera and run the launch file.

You'll get a warning about the camera not being initialized. The camera will start immediately after that warning though.



# Web Server for GUI

The webpage demonstrates that ROS can be run from any device that can run a web browser. Some camera commands are implemented as buttons that can be clicked and send a message over the network.

VNC into ROSBOX2

Open a new terminal

Go to cd ~/catkin\_ws/src/robot\_gui\_bridge

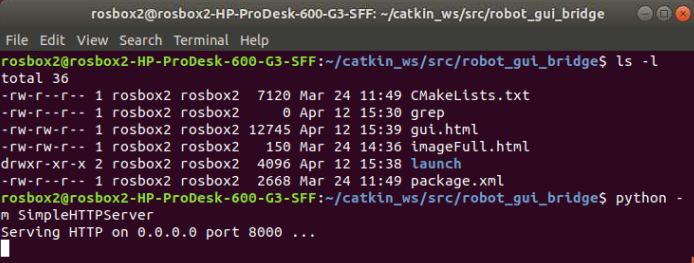
Launch the websocket

roslaunch robot\_gui\_bridge websocket.launch

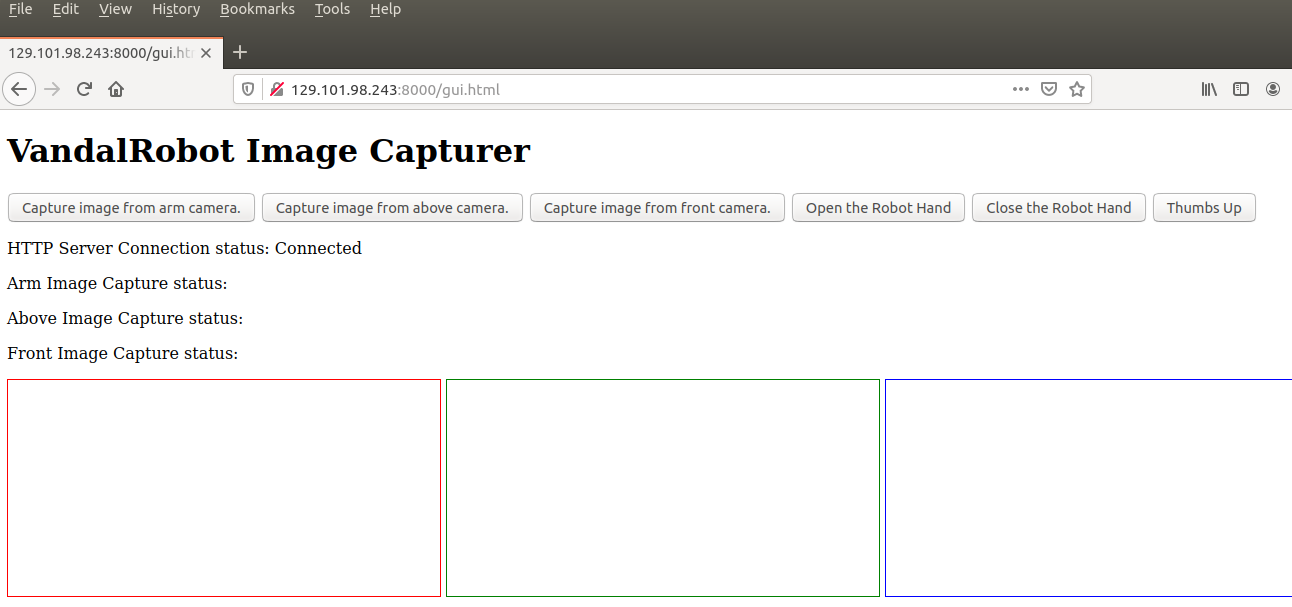
Run the python http server

python -m SimpleHTTPServer

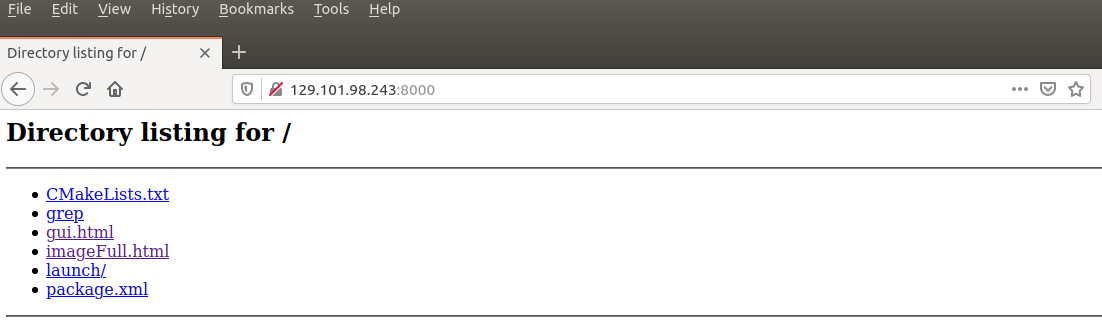
The web page to access is called gui.html. It is not accessible if you run the webserver from a different folder.



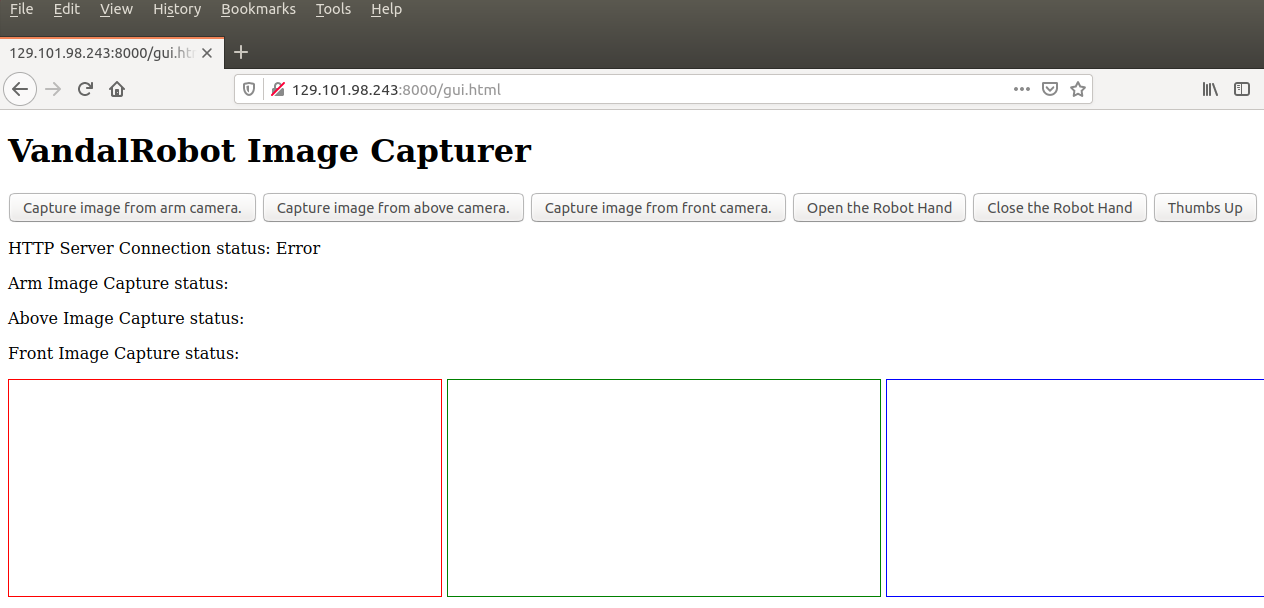
Open web browser on your computer and go to http://129.101.98.243:8000/gui.html



If gui.html is not entered in the URL you get the directory. Click on gui.html.



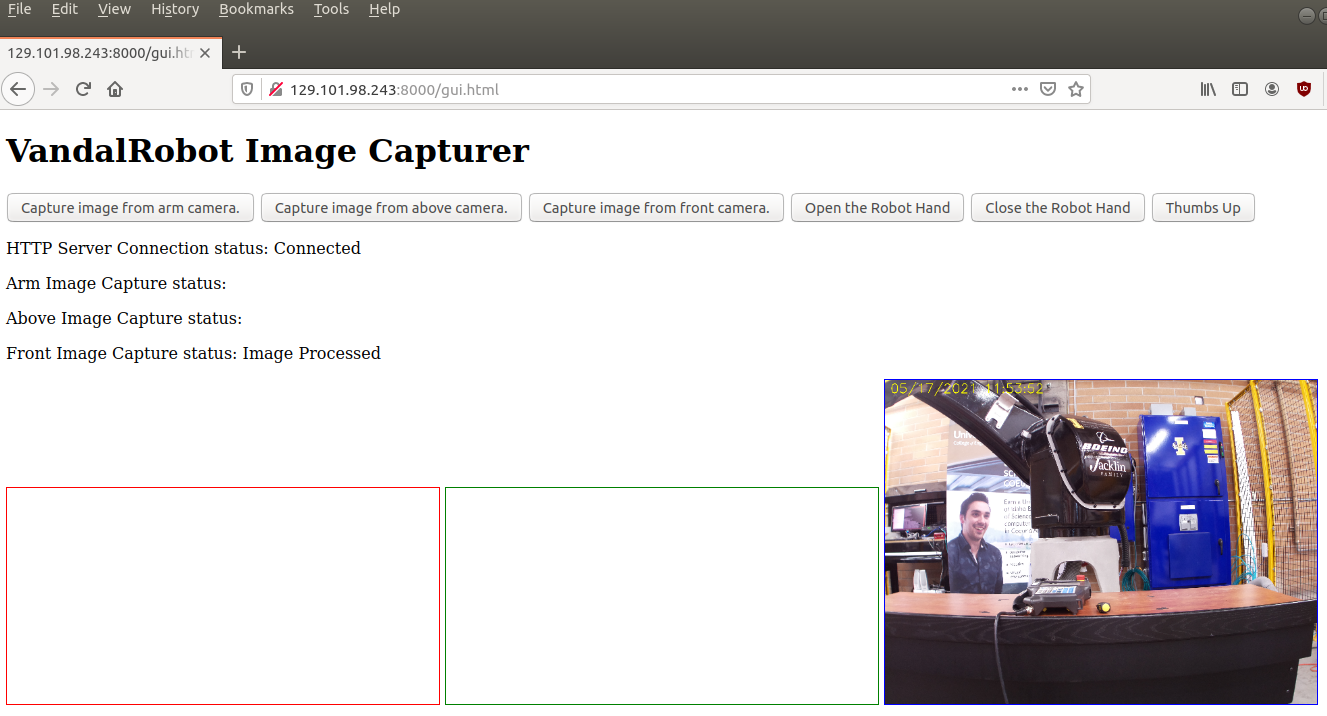
If the HTTP Server Connection status says "error" then the websocket wasn't started first.



The buttons will not do anything unless the relevant camera or hand is started up.

If camera is started up. Click on the capture image button. It will send a message off to ROSBOX2 which will forward it onto the picamera.

The video stream, will stop and then a picture will be taken, timestamped and sent back to the web page. Status messages will be received throughout the process. Messages sent when clicking a button occasionally don’t get received on ROSBOX2 and needs to be fixed.

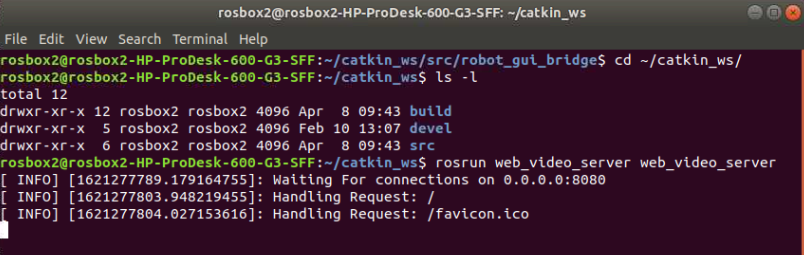


# Web Video Server for Video streaming

VNC into ROSBOX2

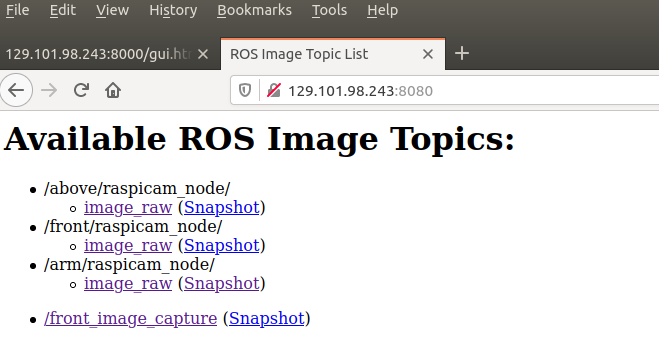
Open a terminal

rosrun web\_video\_server web\_video\_server



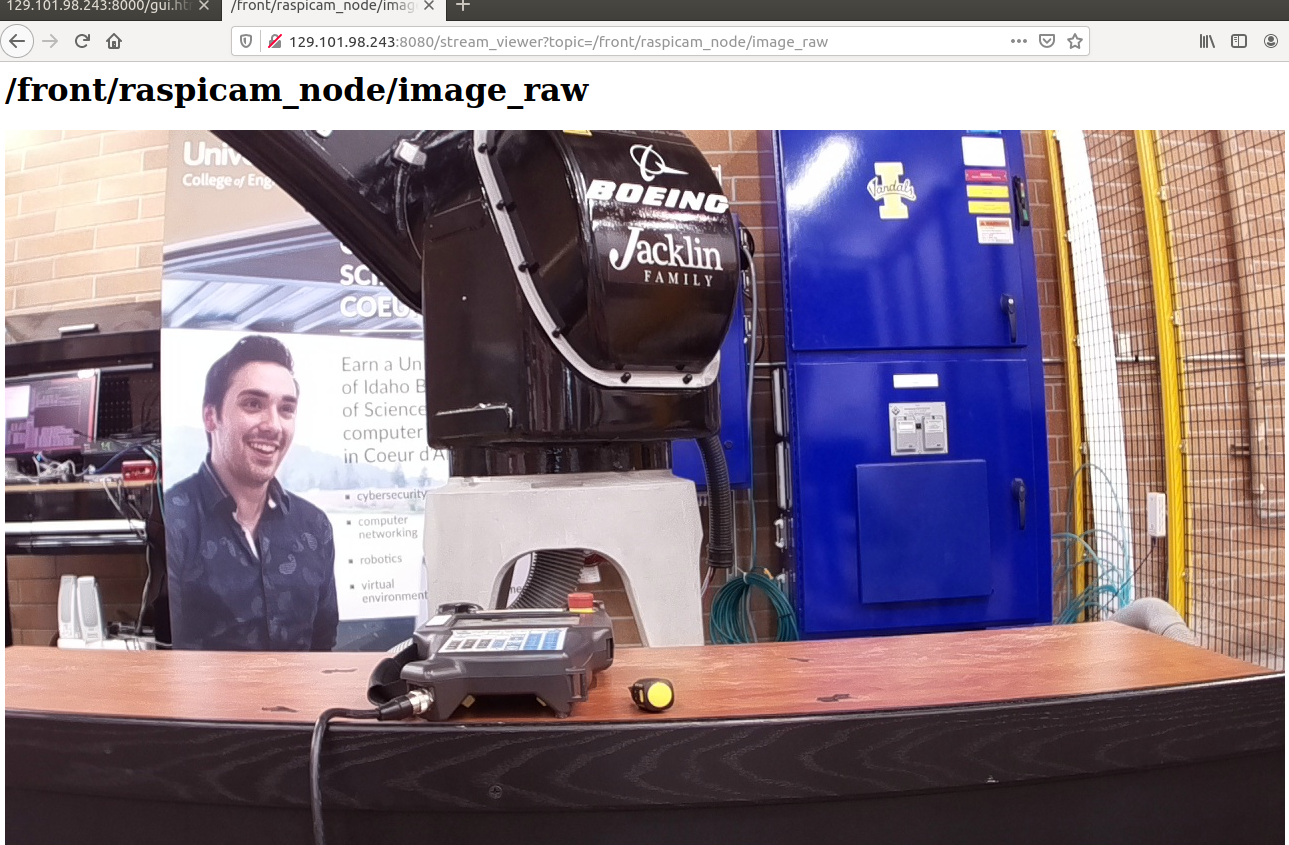
On your computer open a web browser and go to

http://129.101.98.243:8080



Available streams appear

Click on one to see live video stream



# Camera Testing Scripts

repub.py

Hard coded script that subscribed to video stream, drew a blue circle in the top left corner, then republished the stream. Displays the restream on screen.

video\_stream.py

Sophisticated version of repub.py. Adds a timestamp instead of a blue circle.

Specifies the camera and node name. Both are mandatory. Using the same node name as another person will kill their stream. Node name can be anything.

python video\_stream.py <-c <camera name> -n <node name>

video\_image.py

Hard coded script that took a capture of the video stream, added a timestamp, then saved it. Same functionality as the web page but adds file saving.

take\_picture.py

More sophisticated version of video\_image.py

Specifies the camera, format and filename via command line

python take\_picture.py -c <camera name> -f <image format> -n <file name>

Only -c is mandatory

format defaults to jpg, filename defaults to test.