Computer Vision with Raspberry Pi and OpenCV

A. Giavaras

Contents

1   Robot Vision Using a Pi Camera and OpenCV

Giving a robot the ability to see things allows it to behave in ways to which humans relate well. Computer vision is an area that much research is currently devoted to, but some of the basics are already available for use in our own code, with the Pi Camera and a little bit of work. In this section, we will use the robot and camera to drive to objects.

- Setting up a Raspberry Pi Camera on your robot, in terms of both software and hardware

- Use Flask to create a web server to see what the robot sees on our laptop

- Revisiting color models and learning how to mask images with them.

- Using contours to detect the largest blob of color in an image and pointing the robot at it

- Using Haar cascades to detect faces, and pointing the pan-and-tilt mechanism at them

1.1   Set up the Raspberry Pi Camera

We will first attach the camera to the pan-and-tilt assembly. We can then use a longer cable to wire the camera into the Pi.

1.2   Wire the Camera

The Raspberry Pi has a slot specifically for the camera, the camera cable fits into this, see Figure 1. We will be wiring our camera into this slot.
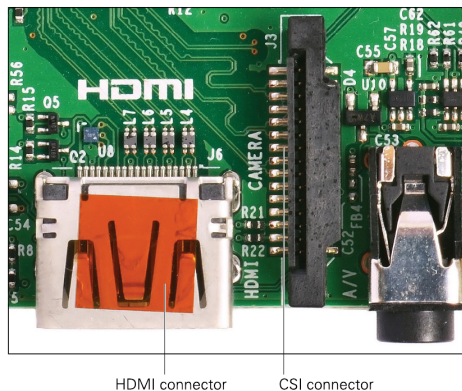

HDMI connector   CSI connector

Fig. 1:   CSI and HDMI connectors.

> *Remark* 1.1. **Troubleshoot**
>
> See the following link `https://www.techcoil.com/blog/connect-raspberry-pi-camera-module-raspberry-pi-2-raspberry-pi-3/` on how to connect the raspberry pi camera module using the long 300 mm cable.

1.3 Activate the Camera

In this section, we will set up the camera, activate it in Raspbian and get a test picture to ensure that everything works properly.

Power up the Pi and log in via PuTTY. At the terminal, type the following:

```
sudo raspi−config
```

In raspi−config, select the **5 Interfacing Options** option, and then select **P1 Camera**. You will then be asked if you would like the camera interface to be enabled. Select **Yes** and **Ok**, then **Finish**. If you are asked to reboot at this point, answer **Yes**.

1.3.1 Get a Picture from the Pi Camera

Now that we have the camera enbled, let's confirm that our setup was successful. Thus, we will use the Pi camera to take a picture for us. This will check whether all the connections are good or not. If there are problems detecting the camera, please go back and check that the cable connection is correct, that you have installed picamera, and that you have enabled the Raspberry Pi camera in raspi−config.

Reconnect to the Raspberry Pi with PuTTY, and type the following to get a picture:

```
raspistill −o test.jpg
```

raspistill takes a still image, and the −o parameter tells it to store that image in test.jpg. We can use the FileZilla client (see appendix) to download this image and verify it on our computer.

1.4 Setting up OpenCV, NumPy and PiCamera

In this section, we will install the OpenCV, `https://opencv.org/`, library on the Rasperry Pi. In case you do not know OpenCV, this is a library with a collection of tools for manipulating pictures and extracting information from

them. The name is an abbreviation of Open Computer Vision. The tools are strung together to make useful behaviors and pipelines for processing images. To be able to run our code on the Raspberry Pi, we will need to install the Python OpenCV library.

We will also install NumPy, `http://www.numpy.org/`, the numeric Python library. This lets us to do manipulations on large blocks of numbers.

Finally, we will install PiCamera. `https://picamera.readthedocs.io/en/release-1.13/`, a library for manipulating our Pi camera.

1.4.1 Install NumPy

We can install NumPy using pip

```
pip3 install numpy
```

1.4.2 Install OpenCV

We can install OpenCV via terminal

```
sudo apt install python-opencv opencv-data
```

> *Remark* 1.2. Have a look at the following links should you have problems in setting up OpenCV in Raspberry Pi `https://blog.piwheels.org/how-to-work-out-the-missing-dependencies-for-a-python-package/`

1.4.3 Install PiCamera

Finally, we will install the PiCamera library so that we can interact with the Pi camera

```
sudo pip3 install picamera[array] numpy
```