

Automatic Trashcan Level Detection with Raspberry Pi

Zavier Basnight

Old Dominion University

Table of Contents

Mission statement	4
1. Problem Specification	4
1.1 Raspberry Pi Model B.....	5
1.2 Camera Module v2	5
1.3 Breadboard and GPIO Pins	7
1.4 Python Libraries and Packages	9
2. Initial challenges and Solutions	11
3. Conclusion	17

List of Figures

Figure 1.1 Raspberry Pi 4 Model B	5
Figure 1.2 Raspberry Pi Camera Module 2	6
Figure 1.3 Setup of Raspberry Pi and Camera Module	6
Figure 1.4 Breadboard and LED Circuits	7
Figure 1.5 Raspberry Pi 4 GPIO layout	8
Figure 2.1 Original image converted to a cropped image	11
Figure 2.2 Original cropped image converted to greyscale	12
Figure 2.3 Greyscale cropped image converted to canny filter	13
Figure 2.4 LED lights connected to breadboard	14
Figure 2.5 3 levels created by array slicing	15
Figure 2.6 Results of LED lights associated with the arrays	16

Mission Statement

In this program we will be creating a system that will detect when a trash can is full. The goal is to help communities contain its trash and protect the oceans. Sometimes, trash can make its way from land to the ocean. Debris that is dropped on the ground is then carried by storm drains, canals, or rivers which eventually ends up in our oceans causing pollution. The goal is to have a device that can specify when a trashcan reaches a certain level of fullness before it overflows.

1. Problem Specification

For this project, we will have a webcam placed at a fixed distance away from a trashcan. A filter will be used to assist the program to identify objects inside the trashcan. We will then use an algorithm that will crop a picture taken with the webcam, labeling the top and bottom of the trashcan. As the program is detecting the level of fullness a blink indicator will be used to communicate with the user on the status of levels of fullness. A green light will indicate that the level is low, yellow light will indicate a medium level, and red will indicate a high level which will communicate to the user that the trashcan is full and is ready to be replaced.

1.1 Raspberry Pi and Camera Module Model B

Raspberry Pi 4 Model B offers ground-breaking increases in processor speed, multimedia performance, memory, and connectivity compared to the prior-generation Raspberry Pi 3 Model B+, while retaining backwards compatibility and similar power consumption. For the end user, Raspberry Pi 4 Model B provides desktop performance comparable to entry-level x86 PC systems (Raspberry Pi 4 Computer Model B).

Figure 1.1

Raspberry Pi 4 Model B



1.2 Camera Module B

The Camera Module 2 can capture both high-definition video and static photos. The v2 Camera Module has a Sony IMX219 8-megapixel sensor (compared to the 5-megapixel OmniVision OV5647 sensor of the original camera) The camera works with all models of Raspberry Pi 1, 2, 3 and 4. It can

be accessed through the MMAL and V4L APIs, and there are numerous third-party libraries built for it, including the Picamera Python library. (Raspberry Pi.)

Figure 1.2

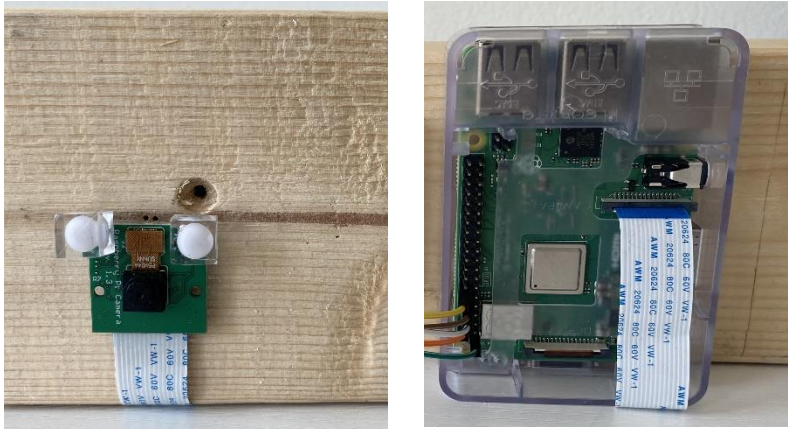
Raspberry Pi Camera Module v2



To capture the image of the trash can first the raspberry pi 4 is positioned and attached vertically to a 2 x 4 piece of wood. The camera module is wrapped underneath the bottom of the 2x4 where it is held in place by pins. Finally, the 2x4 rests on a table, allowing the camera to hold steady to capture the image for the program to run.

Figure 1.3

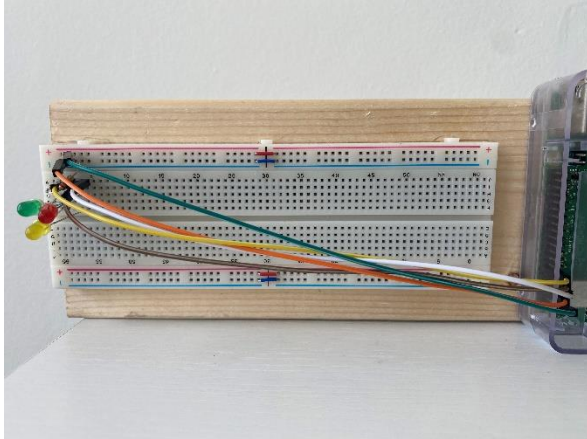
Setup of Raspberry Pi and Camera Module



1.3 Breadboard and GPIO pins

Breadboards have a unique way to logically control currents. Beneath the plastic of each row is a metal strip. The metal functions as a conductor allowing charge to flow through the row. The five holes are simply access points to the metal strip (Raspberry DIY). Each of the four rails has a single metal strip beneath the plastic. This means that connecting a power source to any point in a rail makes the current available to every other hole along the rail (Raspberry DIY).

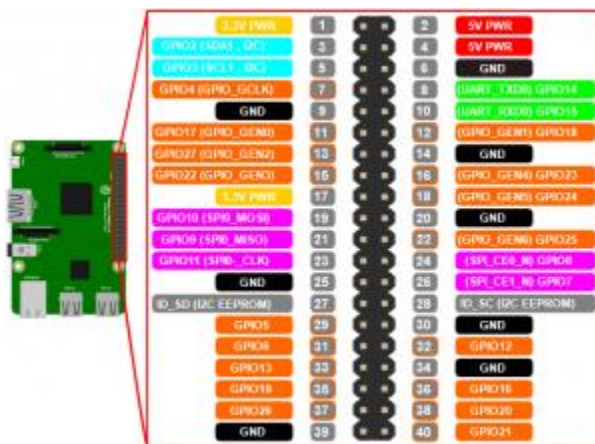
Figure 1.4
Breadboard and LED circuits



GPIO stand for General-Purpose Input/Output. These pins are used to connect the raspberry pi to the outside world. This is because it helps the raspberry pi control and monitor components of the outside world by connecting to electronic circuits. The Pi can control LEDs, turning them on or off, run motors, and many other things. It's also able to detect whether a switch has been pressed, the temperature, and light.

Figure 1.5

Raspberry Pi 3 Model B GPIO Layout



1.3 Python Libraries and Packages

The engine behind the trashcan level detection is written in Python which uses several different libraries. Down below will be a list of libraries that was used and a brief description of what they do, along with systems algorithm.

1.4.1 OpenCV

OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library. (OpenCV) Some of the algorithms that used can detect and recognize faces, identify objects, track moving objects, and produce high resolution images of an entire scene.

1.4.2 Pi Camera

The PiCamera library offers a Python interface for operating the Raspberry Pi camera module. This package is available for Python 2.7 and above, as well as Python 3.2 and higher. This package is used inside the system to start a camera preview before capturing the trashcan, to stop the camera preview, and to save the trashcan image to a file.

1.4.3 Python Imaging Library (PIL)

PIL is an additional, free, open-source library for the Python programming language that provides support for opening, manipulating, and saving many different image file formats (Educative)

1.4.4 NumPy (np)

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more (NumPy v1.23 Manual).

1.4.5 Matplotlib

Matplotlib is a data visualization and graphical charting package for Python and its numerical extension NumPy that is cross-platform. As such, it provides an open-source alternative to MATLAB. Developers may also incorporate plots in GUI programs by using matplotlib's APIs (Application Programming Interfaces).

1.4.6 Scikit-image

Scikit-image (previously scikits. image) is an open-source image processing package written in Python. Algorithms for segmentation, geometric

transformations, color space manipulation, analysis, filtering, morphology, and feature recognition.

2 Initial Challenges and Solution

The biggest challenge that emerged with the development of the trashcan level detection system is coming up with a way to set triggers when the trash reaches a certain point. The objective was to implement 3 levels that had to be created and send a signal to a LED lights which would communicate to the user the updated level of the trashcan.

First, to begin the process of developing the triggers, an image had to be captured from the webcam attached to the raspberry pi. This image would provide a 2D image that would include the trashcan and the background surrounding it. The challenge with capturing the image would be to point out and focus on the trashcan by itself, ignoring the background and the objects surrounding it. The solution for this would be to crop out the trashcan in the image itself.

Figure 2.1

Original image converted to a cropped image



Next, now that the image of the trashcan has been cropped, pixel values must be found. A pixel (short for picture element) is a small block that represents the amount of gray intensity to be displayed for that particular portion of the image (Why Do Math?). Pixel values in most images are integers ranging from 0 (black) to 255 (white). The objective for this step is to limit the pixel values as much as possible by using a greyscale that will help with edge detection when it is converted to a canny filter.

Figure 2.2

Original cropped image converted to greyscale



Furthermore, once the cropped image is converted to a greyscale it is now ready to be converted through a canny filter used by the OpenCV library. By computing the gradient of the picture using edge detection operators, the gradient calculation step determines the edge intensity around the objects that will be inserted into the trashcan. With the canny filter applied, only pixel values 0 and 255 are around the objects as seen below.

Figure 2.3

Greyscale cropped image converted to canny filter

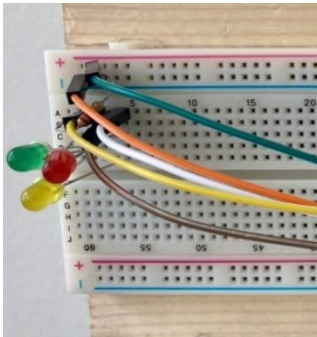


Additionally, now that the image has been converted from a greyscale to a canny filter and objects inside the trashcan has been identified, creating levels can be implemented. The objective in this step is to create 3 levels low, medium, and high. As stated in the introduction these 3 levels will communicate to the user when it is time to change the trashcan when it

reaches the high level. These 3 levels will be represented by green, yellow, and red LED lights that will blink.

Figure 2.4

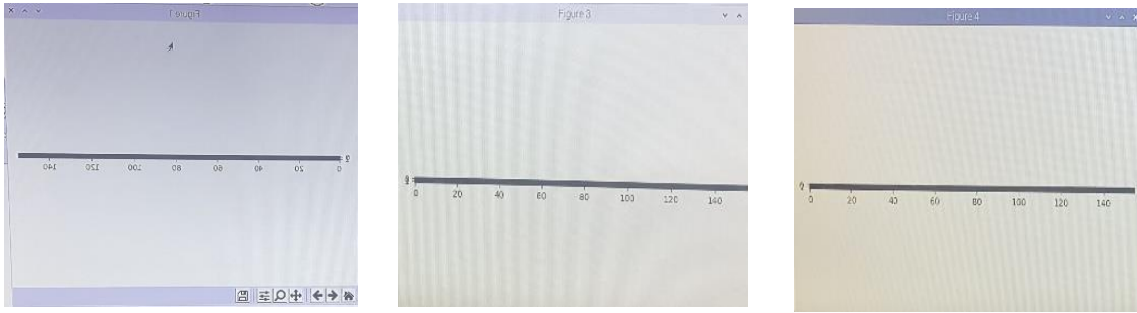
LED lights connected to breadboard



Therefore, to create the 3 levels we first need to locate pixel values. This is done by converting the canny filter image to an array using the NumPy library. With NumPy, an array is created from the image, in which case 3 sections of that image can be used by slicing the array. This can be done by identifying 2 rows and 1 column from the original cropped canny filter image. The Scikit-image is then used to assist with creating a grid to label the x and y axis as seen in Figure 2.5.

Figure 2.5

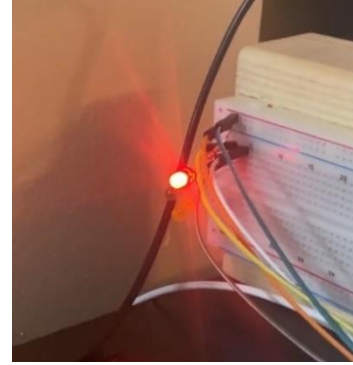
3 levels created by array slicing



From the array generated by NumPy, we then use a loop. Starting from the top left corner to the top right corner, the loop travels through each row and column while locating the pixels values. Once the script is completed, the program opens 3 windows created by the array slice as shown in Figure 2.5. While the loops capture the pixel values it also goes through a for loop that will trigger the LED light when the pixel values in each array reaches over 1,000 pixels. When the loop has completed it then enters a for loop. In the for loop with an else if statement, starting from the top to the bottom. Whichever level has over 1,000 pixel stops the loop and sends a signal to the LED light that is associated with that array as seen in Figure 2.5.

Figure 2.6

Results of LED lights associated with the arrays



Conclusion

With the trashcan level detection project, the goal of notifying the user when the trashcan was full had successfully been completed. From the start of the project the most important resources given to complete this task, was a image taken by a webcam. That image was then cropped, converted to a grayscale filter, and then through a canny filter. From this canny filter, the program was able to identify edges around the objects that were inside. The cropped canny image was then converted to an array where the trashcan level detection was able to slice the image into 3 arrays. These arrays were used to find the pixel values which would then be collected and compared to an overall pixel value of 1,000 in each array. Once the pixel value had been reached, the program stops and sends a signal to an LED light that is associated to the array. Thus, notifying the user when the trash can is full and ready to be emptied.

References

“About.” OpenCV, November 4, 2020. <https://opencv.org/about/>.

“How to Connect a Breadboard to the Raspberry Pi 4.” Raspberry DIY, October 13, 2020. <https://raspberrypi.com/raspberry-pi-breadboard-tutorial/>.

“Image Compression: How Math Led to the JPEG2000 Standard.” Why Do Math?, n.d. <https://www.whymath.org/node/wavlets/imagebasics.html>.

Matplotlib tutorial, n.d. https://www.w3schools.com/python/matplotlib_intro.asp.

Raspberry Pi. “Buy A Raspberry Pi Camera Module 2.” Raspberry Pi, n.d. <https://www.raspberrypi.com/products/camera-module-v2/>.

“What Are Some Basic Functions of Image Processing Using Pil?” Educative, n.d. <https://www.educative.io/answers/what-are-some-basic-functions-of-image-processing-using-pil>.

“What Is Numpy?#.” What is NumPy? - NumPy v1.23 Manual, n.d. <https://numpy.org/doc/stable/user/whatisnumpy.html>.

“Raspberry Pi 4 Computer Model B,” n.d. <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-4-Product-Brief.pdf>.