ELECTRICAL ENGINEERING DEPARTMENT
COLORADO SCHOOL OF MINES

Name Eric Sobczak

Goal for exercise – "Introduction to OpenCV"

My goal for this exercise was to refamiliarize myself with using Open CV on raspberry Pi, and learn how ArUco detection works, and how to perform it.

What do you know about OpenCV?

Coming into this assignment, I was fairly familiar with using Open CV on the PI. Previously, I had worked through the tutorials of Py Image Search to perform object/face detection within Open Cv. These tutorials, however, were for Python 2, and were more focused on getting Open CV built/running instead of what the code was doing.

After this assignment, I have a much stronger understanding of how Open CV handles images, and what the various function calls actually do. I also know now how the ArUco markers works and how I can use them.

List all resources and what specifically you used or learnt from that resource to complete the challenge exercises.

Most instrumentally, I used the blog of PyImageSearch to understand how to get Open CV working and capturing images on the raspberry Pi. The example coded provided in these blogs are well explained and provided me a foothold to get started.

For the more complex Agrico pose estimation, I used the blog blog.dgut.top. While this in Chinese, the code contains the necessary function calls to do camera calibration. Along with google translate and the official OpenCV documentation I got my own calibration working.

In addition to these resources, I used online forums like stack overflow to solve specific niche issues within my code. These mostly revolved around what parameters need to be passed into functions or array handling. These sources are detailed in the footer of my code.

Name Eric Sobczak

Compile a list of all documentation created. Provide file name and a short description of that file.

All of my documentation is within “documentation.pdf” submitted on Canvas. This file contains both answers to the assignment questions and the code for all the exercises. The header of the exercise code provides both run instructions and the basics of how it works. The footer contains the various online resources I used to write the code

Provide an example of something that you would do differently or you could improve upon during the course of this exercise.

My research led me to find that there are a few different ways to capture video from the Pi Camera. These include `capture_continuous` with `picamera`, using the library `imutils.video` or even the new `picamera2`. I need to do some further investigation and find which of these techniques will be able to use the least resources at the fastest capture time. I have read it might even be faster to run still grayscale captures, rather than perform a greyscale calibration. I also wish I did a better job of using function throughout all of my code.

On a scale of 1-5, what is your comfort level with OpenCV after going through this exercise? (1 being least comfortable and 5 being most comfortable).

1 2 3 [✓ 4] 5