Lab 4 - Real time image processing

CC3501

Task Description

Build an image processing system that locates coloured objects in a live camera feed. Your system must process live images that have been captured from the Raspberry Pi camera. A demonstration is shown in Figure 1, below.

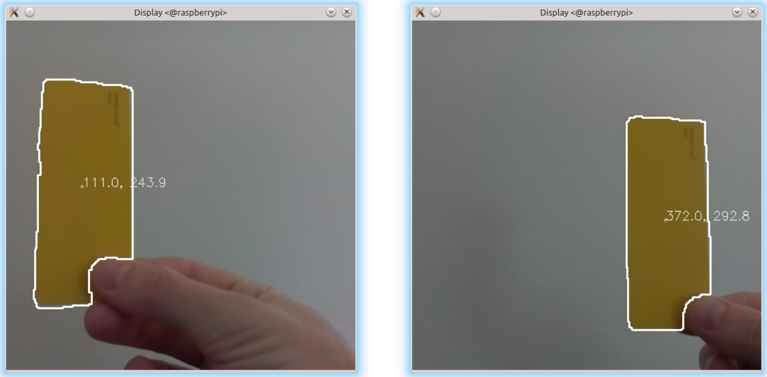

Figure 1: Demonstration of an object detection system. Identified objects are outlined in white, and the centre of mass coordinates are displayed on the image.

Your system does not need to draw outlines around the detected objects. It is sufficient to show the threshold result in a separate window, and print the (x, y) coordinates to the terminal.

Step 1: Assemble the camera ✓

Carefully connect the camera according to the instructions:

`https://www.raspberrypi.com/documentation/accessories/camera.html`

The camera is delicate! Treat it gently.

Step 2: Compile the PiCapture starter code ✓

The starter code available in the Github repository below shows you how to capture images from the camera and feed them into OpenCV.

`https://github.com/bronsonp/picapture`

Clone this repository, compile the code, and verify that the camera image can be displayed.

Refer to last week's lab for a refresher on how to download code and build it.

Step 3: Modify the code

Your task is to merge together the code from last week (which performed colour thresholding on a static image) with the code from this week (which captures a live image but doesn't do any processing).

Follow these steps:

1. Modify the PiCapture code to add in colour thresholding, based upon the program from last week. As per last week, you should implement interactive sliders to allow the detection parameters to be adjusted in real time. ✓
2. Enhance your thresholded image using morphological open and close operations, as discussed in lectures. ✓
3. Locate the detected object by calculating its centre of mass position. Print the identified location to the terminal.

Assessment

To complete this lab, demonstrate object detection that includes:

- Live displays of the camera image and the thresholded result,
- Adjustable detection parameters including the size of the structuring element for open and close operations, and
- Centre of mass coordinates printed to the terminal.

Note that you do not need to draw borders around the detected objects, and you do not need to draw the centre of mass onto the image itself. That is an extension task.

Extension Task

As an optional extension task, make your solution look like Figure 1:

- Draw a border around the detected objects. Hint—Look up the `findContours` and `drawContours` functions.
- Display the centre of mass coordinates on the image itself. Hint—Look up the `putText` function.