Lab 4: Color Segmentation

Due Sep 21, 2020 by 11:59pm **Points** 10 **Submitting** a text entry box or a file upload

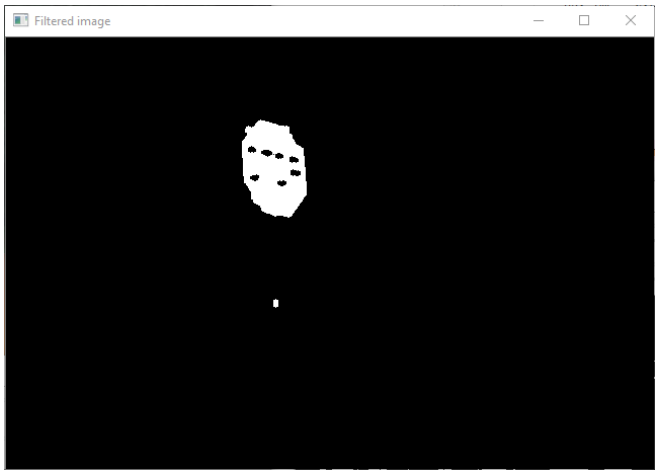
As discussed in lecture, when processing color images it is usually helpful to convert to HSV color space, because the hue and saturation values for an object are much more invariant than the intensity. In the lecture, I showed a Python/OpenCV program that processed a color image of a cup, and applied threshold operations to segment out the pixels of a certain color. In this lab, you will use that program to segment out the pixels of stop signs.

On the course website are five images of stop signs (**stop0.jpg** (https://elearning.mines.edu/courses/25410/files/2127924/download?download_frd=1) , **stop1.jpg** (https://elearning.mines.edu/courses/25410/files/2127925/download?download_frd=1) , **stop2.jpg** (https://elearning.mines.edu/courses/25410/files/2127926/download?download_frd=1) , **stop3.jpg** (https://elearning.mines.edu/courses/25410/files/2127928/download?download_frd=1) , **stop4.jpg** (https://elearning.mines.edu/courses/25410/files/2127929/download?download_frd=1)). An example of one image is shown below.

Find threshold values that best detect the points on the stop signs, while ignoring the background as much as possible. The same threshold values should be used for all five images. For example, below is the segmented output image with most of the stop sign pixels detected, and very few in the background.

Make sure you produce a binary image where the stop sign is the largest connected component in the image. (You don't actually need to run connected component labeling; it should be obvious by looking at the results that you have a connected component.) To clean up the results, and make sure that your stop sign pixels are all connected, you should do

a morphological “closing” followed by an “opening” on your binary image. For example, below is my output image after these morphological operations.



Turn in:

- State the threshold values you used for hue, saturation, and value (there should be a total of six numbers), and the size of the morphological structuring elements for closing and opening (two numbers).
- Upload the five segmented output images (one for each stop sign image), where each image shows that the stop sign is the largest connected component in the image.
- Upload your Python program.