Image Segmentation

Arnav Malhotra

Project Statement 1

This reads in the image 'hopkins1.bmp'. MATLAB views the image as a 3-D matrix, with red, green, and blue being each respective dimension. Using the given equation in the project statement, I converted the image to grayscale.

Project Statement 2

Using the built in function 'imhist', I created a histogram for the image. 'imhist' only works with the grayscale, so first I converted the image to grayscale. The number of bins corresponds to the intensity and the y-axis corresponds to the number of pixels at that intensity.

Project statement 3

The first part was the same as project statement 1 and 2. Then by looking at the histogram, I decided on the threshold of 150. I then set up a nested for loop to run through all the elements of the grayscale image matrix. Any pixel with an intensity greater than or equal to the threshold is a value of 0 (white) and anything below is a value of 1 (black). Therefore the foreground is in white and the background is in black.

Arnav Malhotra
EN.560.200
Dr. Michael Beer
Final Project – Image Segmentation

Project Statement 4

The first part is the same as before. The last part is the same as #3. I set an initial "guess" threshold at 150. Anything above the threshold is the foreground and anything below it is the background. Using the 'fit' function, I fit a Gaussian to the background. The foreground needed a Folded Gaussian (i.e. Gaussian of Type 2), so that was the one that was fitted. Since the find doesn't work for fit type objects, I had to devise a way to find the intersect of the two curves. The way I did this was I plotted all the x and y values for the range of the histogram (i.e. x = [1:256]) and then subtracted the two. The minimum value of that was found (since sometimes the intercept would not be an integer value), and the corresponding x value would be the intersect of the two curves and the new threshold. However this method had some errors. Sometimes it would oscillate between 2 numbers, therefore I put in a count to break out after a certain number of iterations. The other error was that sometimes the intersect of the two curves would be 1 and that would not allow it to calculate the y values, therefore I said if the intersect is ever below 100, it would restart the loop.

Challenge 1

Using the function 'findpeaks', I found the local maxima in the histogram. I thresholded it such that each image would only display the intensity of that peak and its neighbourhood. With that I found each object and displayed it by itself on an image.

Arnav Malhotra
EN.560.200
Dr. Michael Beer
Final Project – Image Segmentation

Challenge 2

I just segmented it the same way as in Project Statement 4. The difference was that this time both were fitted with a Gaussian curve (that is, no folded Gaussian was used) as this gave a better fit.