MICHELLE TAN
mtan23@binghamton.edu

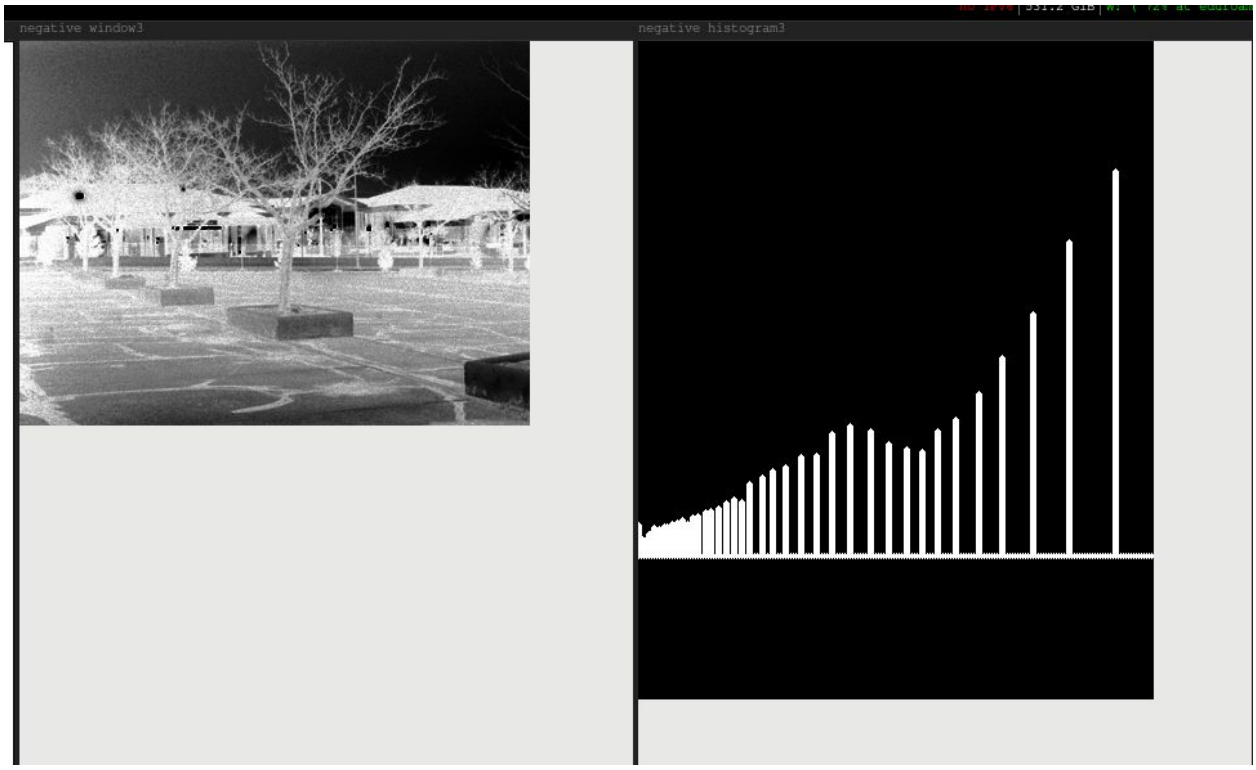
PURPOSE

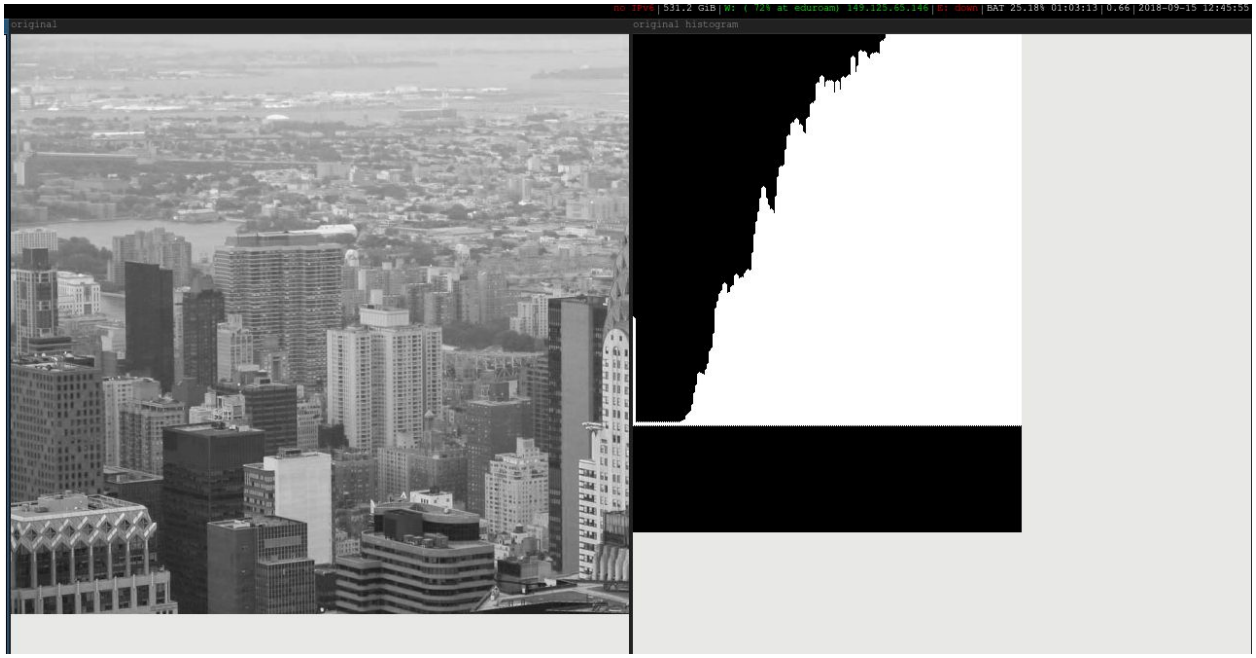
This project enhances images by equalizing the image's histogram. The project also ranks the size of image segments by coloring the largest segment 200 and the smallest segment 60. All in between segments are colored 120.

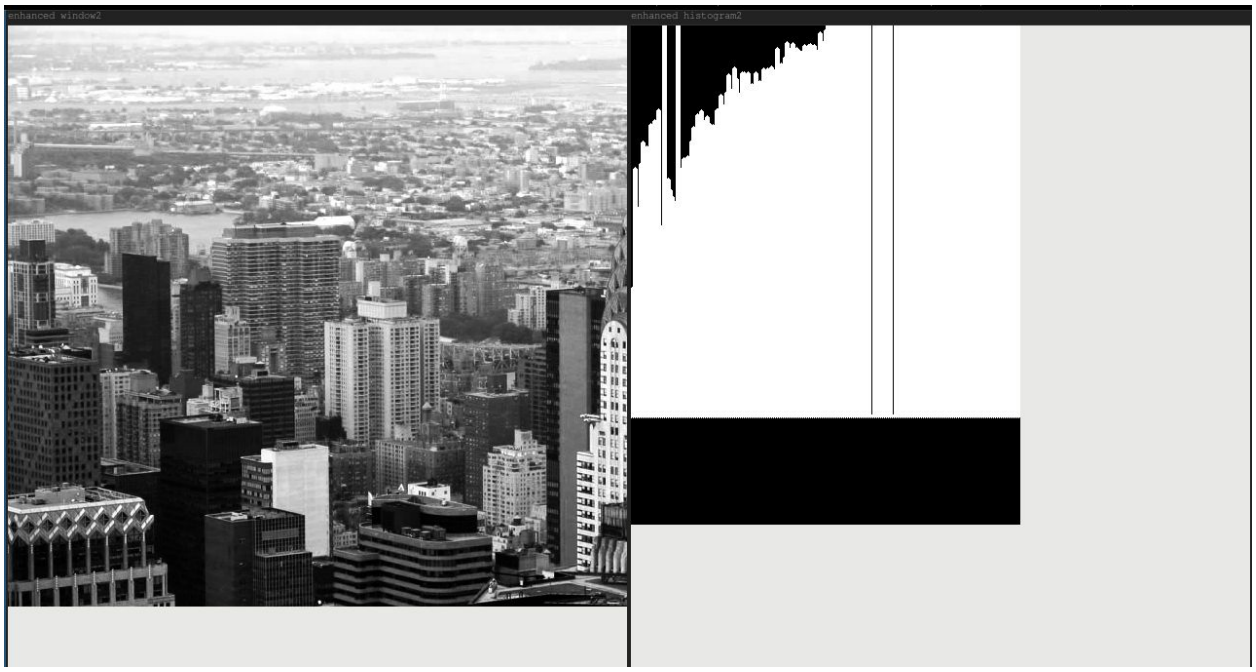
METHOD AND RESULTS

For the image enhancing first we must calculate the actual histogram by counting how many pixels of each value exist. Then add them to an array where the index shows the pixel value. Then to equalize the histogram find the probability of each color. Then add each value to the next such that at index 4 the content is the probability of getting values 1-4. Finally divide all arrays by the total number of pixels. Then multiply the new value by 255. After that round to the nearest int and that should give you the new mapping. In the images below I show the original, enhancement, and negative of the enhancement.

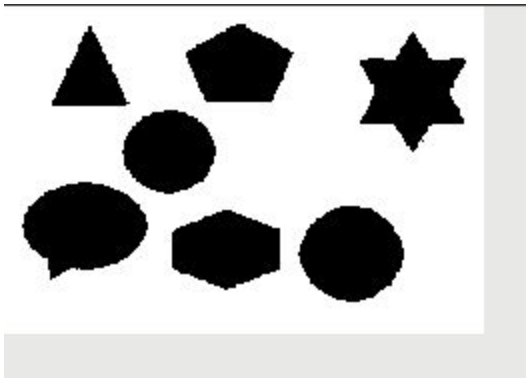
no IPV6 | 531.2 GiB | W: (72% at eduroam
enhanced window2
enhanced histogram2


negative window3
negative histogram3

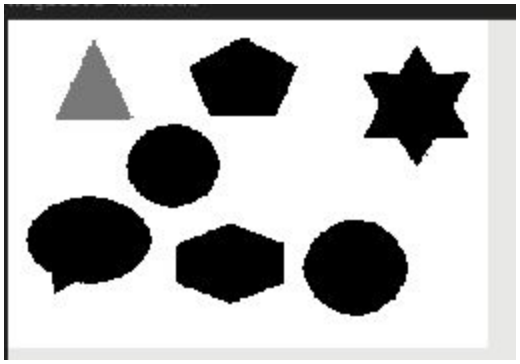

no IPv6 | 531.2 GiB | W: (72% at eduroam) 149.125.65.146 | Bt down | BAT 25.18% 01:03:13 | 0.66 | 2018-05-15 12:45:55
original
original histogram


enhanced window2
enhanced histogram2

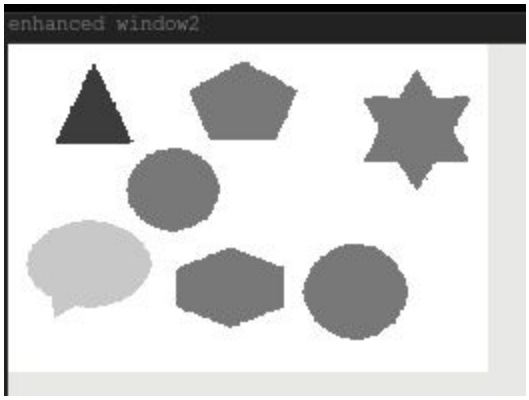
For image segmentation I used the thresholding algorithm to separate the image. To threshold the image I compare each pixel value to the average pixel value and then color it either 255 or 0 depending on . Here is an example of a thresholded image.



After thresholding the image I use a floodfill algorithm to calculate the area of each segment. Here is an example where I floodfill the triangle.



After calculating the areas I sort them and then color the largest with the gray value of 200 and the smallest with the gray value of 60. Here is a picture of a ranked shapes image and a ranked guide image.



BUG REPORT

Ranking the guide.bmp takes a while but eventually happens please give it 5 or more minutes. It shouldn’t take more than 10 minutes.

Please type in make to compile.