Jeremy Midvidy, jam658
EECS 332, Fall 2018
MP3 Report

About My Code:

My code for MP3 can be found in this folder, in the python file, `MP3.py`. The file is callable from the command line with the command `$ MP3.py PATH_TO_IMAGE`.

After running, it will write a result image named "`PATH_TO_IMAGE_result `" of the classified images to the folder named "results." Additionally, it will add two histograms to the results folder, the histogram of pixel intensities before the equalization and the histogram of pixel intensities after the equalization.

I wrote my code in Python 3.7, and it assumes python packages `numpy`, `imageio`, `matplotlib`, and `sys` are installed.

Results:

I ran my code on the given image file, "moon.bmp." In short, my program produced the correct result for the given test case.

To implement the histogram equalization, my program executed the following logic:

(1) Read the given image into a np array. Then convert the image from rgb to grayscale.
(2) Make histogram bins dictionary from the given 2-d array:
    (a) Scan the array. For each cell, if the current cell (representing a pixel intensity) is in the bins dictionary, increment the value of the cell at the bins dict. If the current pixel intensity is not in the dictionary, add it to the dictionary with an initial value of 1.
(3) With the histogram constructed, normalize the value of each pixel by dividing its number of occurrences by the total number of pixels in the array.
(4) Then, quantize each value. I iterate through the dictionary in ascending order, and for each element, set it equal to the total value of all previous probabilities plus the value of the current element. (this is the transfer function).
(5) Then I go through the quantized dictionary and multiply each quantized value by the largest pixel intensity value (255 in moon.bmp)
(6) Finally, I am ready to construct a new image. I scan again through the 2-D array, and for each cell, I lookup it's assignment in the adjusted-quantized dictionary, and then assign the given cell the corresponding label.
(7) Finally, I save the final image to the results folder. Additionally, I save the pixel-intensity histograms of the imputed image and the final image in those folders as well.

For any given input, when called from the command line `$MP3.py path`, the final image will be saved as "path.png", and the two histograms will be saved as "path_historgram after equalization.png" and "path_histogram before equalization.png".

Results Analysis for moon.bmp