ECE-178 Homework #3.
Due on Wednesday, October 23, 2024, 11:59 PM

Name:
TA:

A. Written Problems:

1. Consider an 8-level gray scale image with the following pixel intensity distribution. Compute the transformation that equalizes this histogram.

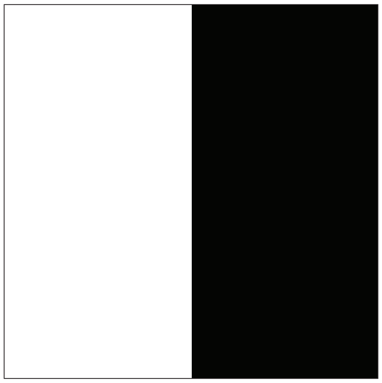
gray level	0	1	2	3	4	5	6	7
# of pixels	400	300	1500	2000	400	600	300	2000

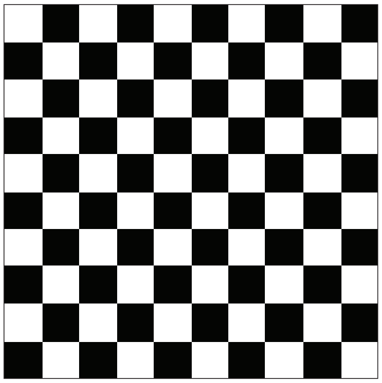
2. Match the given histogram specified in Problem 1 to the following desired histogram.

gray level	0	1	2	3	4	5	6	7
probability	0	0	0.2	0	0.3	0	0.5	0

3. Two black and white pictures, shown below, have identical histograms (the images have the same number of pixels, and identical number of black and white pixels). Suppose we perform the following operation. At each pixel location, we place 3×3 window, and replace the center value by the average of the values in the 3 × 3 window. This operation blurs the sharp boundaries between the bright and the dark regions (assume, for simplicity, the dark values are at 0 and bright values are at intensity 200.) In answering the questions below, make sure that you state any assumptions you are making. (For example, you can ignore the boundary issues as at the boundaries you can not really place the 3 × 3 window. The missing values can either be set to zero or extended from the previous row or column as the case may be).

    (a) Would the histograms of the blurred images still be equal? Explain.
    (b) Sketch the two histograms.





B. Programming Assignment:

For each result, save the final images. In your submission, please provide your code (your .py files) along with your input and output images in a single .zip or .tar file. Please use the following convention to name the zip/tar files: <First name>_<Last name>_HW2.<zip/tar>.

1. Implement the histogram equalization as discussed in class. You cannot use built-in histogram equalization code from any of the libraries. The function should have the form

   J = myHistogramEq (I)

   Where *I* specifies a gray-scale image (intensity in the range 0-255) and *J* is the resulting equalized image with intensity values in the range 0-255. Demonstrate your working code on the examples provided on GitHub and include the original and equalized images in the submitted zip file.

2. Extend the above function to work with color images. However, our version of the equalization would work by taking an RGB color image, applying the equalization to each of the RGB planes independently, and then displaying the result. The function should be of the form

   J = MyColorHistEQ(I)

   where *I* specifies the RGB color image and *J* the equalized color image.

   Demonstrate your equalized results on the examples provided on GitHub and include the original and equalized images in the submitted zip file. Also, comment on your results (e.g., are the equalized results match your expectation of how the images should look like?)

Note: The two images provided on GitHub are stored in the TIFF format and are color images. You should convert them to gray scale images first for gray-scale histogram equalization (assignment 1). For the color part (assignment 2), your code should ensure the input images to the RGB color format before processing or convert them into RGB if they are not.