CS 659 Visual Perception and Learning

**Homework Assignment 2**

Due: 11:59pm, Feburary 23, 2024

The goal of this problem set is to become familiar with basic python commands, practice manipulating vectors and matrices, and try out basic image display and plotting functions.

1. Given a 100 x 100 uint8 matrix A representing a grayscale image, write a few lines of

code to do each of the following. You can create your own matrix A and its elements using any methods in Python.

1. Sort all the intensities in A, put the result in a single 10,000-dimensional vector x, and plot the values in x.
2. Display a figure showing a histogram of A’s intensities with 32 bins.
3. Create and display a new binary image the same size as A, which is white wherever the intensity in A is greater than a threshold t, and black everywhere else.
4. Generate a new image (matrix), which is the same as A, but with A’s mean intensity value subtracted from each pixel. Set any negative values to 0.
5. Use rand to write a function that returns the roll of a six-sided die.
6. Let y be the vector: y = [1:6]. Use the reshape command to form a new matrix z whose first column is [1, 2, 3]’, and whose second column is [4, 5, 6]’.
7. Use the min and find functions to set x to the single minimum value that occurs in A, and set r to the row it occurs in and c to the column it occurs in.
8. Let v be the vector: v = [1 8 8 2 1 3 9 8]. Use the unique function to compute the total number of unique values that occur in v.

**2.** Write a program that will average a collection of images, compute the standard deviations at each pixel, and display the results.

The images below give some examples that were generated by averaging “100 unique commemorative photographs culled from the internet” by Jason Salavon. Your program will do something similar.



There are two collects of images provided for this assignment: set1 and set2. Notice that they are all the same size within a single set.

Write code to do these things per set of images:

• Compute the average image in grayscale.

• Compute the average image in color, by averaging per RGB channel.

• Compute a matrix holding the grayscale images’ standard deviation at each pixel (i.e., X(i,j) holds the standard deviation across all the images’ gray pixel intensities at row i, column j).

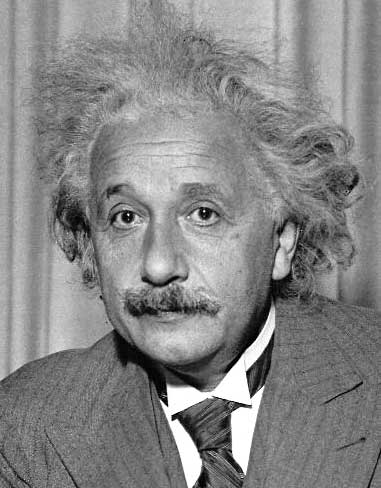
• Display each of the above.

Apply these steps to the two sets, separately. In your write-up, briefly explain the results – why do they look the way they do?

3. Please implement the following filters in Python and apply each of them over a sample image. Note that no third-party filtering functions are allowed to use. You need to write your codes to implement these filtering functions from scratch. Your function should take a graeyscale image as its input and return an image of the same dimension filtered by the following filters:

1. A 3 by 3 box filter (with 1/9 as its element)
2. A 3 by 3 filter that can sharpen the input image
3. A 3 by 3 gaussian filter with sigma =1
4. A 3 by 3 Sobel filter that can compute absolute horizontal gradient
5. A 5 by 5 filter that can translate the input image to the left-hand side by 2 pixels (so the resulted image will include all-zero pixels in its right 2 columns)

Please apply your filtering functions over the following image (q3.png) and include the filtered image in your writeup.



**Submission instructions: what to hand in**

\* Create a google co-lab script to include your codes for Questions 1, 2 and 3. Please ensure your script can access all the images in Question 2 and Question 3.

\* Prepare a PDF file including your co-lab link and your writeup for Question2 and Question 3.

Submit your files through the SDSU canvas

No Hardcopy is required.