Advanced Image Processing: Assignment 4 (Due April 3, 2022)

Note: Please provide detailed comments for code that may be written to solve the following problems. The assignment will be evaluated not just based on the final results but also how you obtained them. Late submissions will be penalized.

Problem 1: JPEG Implementation (20 points)
Implement a toy version of JPEG through the following steps:

1. Transform: Compute an 8x8 discrete cosine transform (DCT) for every non-overlapping block in the input grey scale image.
2. Quantization: Use the following quantization matrix to quantize each DCT coefficient in a given 8x8 block.

$$Q = \begin{bmatrix} c & a & b & b & b & b & b & b \\ a & a & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \end{bmatrix}$$

   with $a = 10, b = 40, c = 20$. Note that the quantized index of the DCT coefficient $x(i, j)$ is given by

$$y(i, j) = \left\lfloor \frac{x(i, j)}{Q(i, j)} + 0.5 \right\rfloor$$

   and the reconstruction is given by $\hat{x}(i, j) = y(i, j)Q(i, j)$.

3. Lossless source coding: Use the following table to encode the quantized index corresponding to each DCT coefficient.

Quantized DCT index	Code
0	0
-1,1	10x
-3,-2,2,3	110xx
-7,-6,-5,-4,4,5,6,7	1110xxx
...	...

The output bitstream (or file) is given by the concatenation of the sequence of bits produced for each 8x8 block.

Using the JPEG implementation described above:

1. Compute the size of the output file generated for the cameraman.tif image provided to you. Also compute the mean squared error between the original image and reconstructed image. The reconstructed image is obtained by taking the inverse DCT for each block of quantized reconstructions of DCT coefficients. Calculate the compression ratio (defined as the ratio of the input image in bits and size of the output file in bits).

2. Compute the file size and mean squared error for a JPEG implementation where each DCT coefficient is merely rounded to the nearest integer without any quantization. Compute the compression ratio.

3. Find (a, b, c) in the quantization matrix where $a, b, c \in \{10, 20, \ldots, 100\}$ that minimizes the mean squared error of the reconstructed image such that the size of the output file is less than or equal to the size of the output file obtained in (1). Compare the reconstructed image with the image reconstructed in (1) both visually as well as using mean squared error.

Problem 2: 2-bit quantizer (30 points)
Consider a Laplacian source with pdf

$$f_X(x) = \frac{1}{6} e^{-\frac{|x|}{3}}.$$

1. Find the optimal quantization points and decision boundaries for a 2-bit scalar uniform quantizer obtained by minimizing the mean squared error distortion. Also compute the distortion achieved by the optimal quantizer.

2. Find the quantizaton points and decision boundaries for a 2-bit Lloyd-Max quantizer obtained with any initialization of your choice. Compute the distortion achieved in this case.

3. Compare the distortion achieved by the uniform scalar quantizer and the Lloyd-Max quantizer and explain your observation.

Problem 3: Comparison of perceptual quality measures (20 points)
In this problem, you will compare MSE, SSIM and LPIPS with respect to their performance in terms of correlation with human perception.

A. Algorithms: Evaluate the following metrics on the dataset mentioned in Part (B). You should obtain a single number as quality with respect to each of the metrics below for each image.

1. Mean squared error in pixel domain

2. Single scale structural similarity index (you can use default code available in Python or MATLAB)

3. Learned perceptual image patch similarity metric (LPIPS): This is a deep learning based measure that is available online (`https://pypi.org/project/lpips/`).

B. Performance measurement: Download the database available at
`http://ece.iisc.ac.in/~rajivs/courses/aip2016/hw5.rar`
The database comes with the following:

1. distorted images in the "gblur" folder
2. reference images in the "refimgs" folder
3. reference image name for every distorted image in the "gblur" folder in "refnames_blur"
4. human opinion scores in "blur_dmos"
5. indicator of whether the image in the "gblur" folder is an original image in "blur_orgs"

C. Questions

1. Compute the Spearman rank order correlation coefficient (again you can use library functions for this) between the dmos scores in "blur_dmos" and each metric you considered in A after having removed the scores that correspond to the original images in the "gblur" folder (as mentioned earlier, this information is contained in blur_orgs).
2. Comment on the relative performances of all the indices

Prepare a report containing the answers to all the problems and submit along with relevant code.