Prof. A. Zakhor Spring 2019

EE225B – Digital Image Processing
Homework 12 – Image Compression

Overview:

In this assignment, you explore some image compression techniques, and evaluate their performances by fidelity criteria.

Assignment specifics:

Part I: Objective Fidelity Criteria

Write a program to compute the root-mean-square error [see Eq. (8-10)] and mean-square signal-to-noise ratio [per Eq. (8-11)] of a compressed- decompressed image. This project is generic in the sense that it will be used in other projects that follow.

Download the image of Figure 1.1 from the course website and write a program to generate the results using uniform quantization and IGS quantization. Use your fidelity criteria program to characterize any loss of visual information and comment on your results.

Part II: Image Entropy

Write a program to compute the entropy of an image [see Eq. (8-7)].

Download the images of Figures 1.2(a) and 1.2(b) and use your program to estimate their entropies.

Part III: Transform Coding

Write a program to compute the information loss associated with the following transform coding schemes:

	Case 1	Case 2
Transforms	Fourier	Cosine
Subimage Size	8×8	8×8
Bit Allocation	8-largest coding	8-largest coding

[a] Use the routines developed in Part I to quantify the loss of information.

[b] Download the image Figure 1.3 and use the program to compare Cases 1 and 2.

Gradually decrease the number of retained coefficients until the reconstruction error for Case 2 becomes objectionable. That is, try 7-largest, 6-largest, … coding as the bit allocation method. Display the output images for each class as well as the computed 'loss of information' required by [a] for 8/7/6/../1 bit allocations.

Note:

For each problem, you need to:

[1] Email your source code (zip it before you email) to ee225bsp19@gmail.com if the question asks for any implementations.

[a] Make sure it is executable because I need to run your code to give you a score. Either MATLAB or Python is okay.

[b] Email title: FirstName_LastName_HW#. For example, Luya_Zhang_HW1

[2] Submit a single PDF file (not word or other formats) on Gradescope which contains:

[a] your answer for each problem;

[b] your source code;

[c] your output image.

Make sure to prepare your solution to each problem on a separate page. On Gradescope, please select and match each page to the corresponding problems.

Please do make sure that you have BOTH your output images AND codes included in the Gradescope submission AND email me your executable codes, otherwise you will lose credits.

Here are some helpful Matlab commands:

`X = fft2(x)`	Computes the 2D-DFT of the matrix `x`
`x = ifft2(X)`	Computes the inverse 2D-DFT of the matrix `X`
`X = dct2(x)`	Computes the 2D-DCT of the matrix `x`
`x = idct2(X)`	Computes the inverse 2D-DCT of the matrix `X`
`I = uint8(x)`	`I` is a matrix of integers ranging from 0..255
`imshow(I)`	Displays `I` as a grayscale image in the current figure
`I = imread('small.bmp', 'bmp')`	Reads the image file small.bmp and stores it in matrix `I`
`imwrite(I, 'result.bmp', 'bmp')`	Writes the matrix I to the image file result.bmp
`[B I] = sort(A, mode)`	Sorts the array `A` in the ascending or descending order