Homework 5

(Total 250 pts)

Due 5:00 pm, April 25, 2025 (Friday)

Submit five files on Canvas ("HW5.pdf", "Q1.m", "Q2.m", "Q4.m", and "Q5.m")

1. (50 pts) Rate/distortion tradeoffs of JPEG compression on a digital image.
   (a) Load into Matlab the following image:
   http://www.ece.uah.edu/~dwpan/course/ee614/data/lena.mat
   Show the image.
   What is the entropy, *H*(*I*), (in bits/pixel) of this image *I*? How many pixels, (*Num_Pixels*) are there in this image?
   (b) JPEG-compress the above image using the Matlab function `imwrite`, with the "Quality" set to 15. What is the size, *FS*, (in bytes), of the resulting JPEG-compressed file?
   (c) What is the coding rate (*CR*), in bits/pixel, of the JPEG scheme above, where $CR = FS \times 8/(Num\_Pixels)$.
   (d) Decompress the compressed file in (b) using the Matlab function `imread`. Calculate the distortion of the reconstructed file against the original image in (a). What is the PSNR (peak signal-to-noise ratio) in dB?
   (e) Fill in the following table with results obtained above

H(*I*) (bits/pixel)	*Num_Pixels*	*FS* (bytes)	*CR* (bits/pixel)	*PSNR* (dB)

   (f) Repeat the steps (b) through (d) by gradually increasing the "Quality" setting from 15 to 95, with a step size of 10. Plot the resulting rate-distortion curve (for the range of "Quality", [15, 25, …, 85, 95]) using the following Matlab commands. Attach the curve.

```
plot(CR, PSNR,'-o'); grid;
xlabel('Coding Rate (Bits/Pixel)');
ylabel('Peak Signal-to-Noise Ratio (dB)');
title('Rate/Distortion Curve of the Image using JPEG')
```

   (g) Show three reconstructed images, corresponding to "Quality" setting of 15, 75, and 95, respectively.
   (h) Enclose the Matlab scripts you wrote in the main submission file "HW5.pdf". Also, submit the M file "Q1.m" separately on Canvas.

2. (40 pts) Block-based lossy compression using DCT and quantization.
   (a) Load into Matlab the following image *I*:
   http://www.ece.uah.edu/~dwpan/course/ee614/data/lena.mat

   (b) Evenly partition the image into blocks of 8×8. Load the 8×8 block at the upper left corner of the image into a matrix (*block*).

```
block = I(1:8, 1:8);
```

Show the content of this 8×8 block.

(c) Apply DCT on the above image block, using function `dct2` in Matlab. Show the DCT coefficients *dct_coef*(i, j).

(d) Load the following quantization matrix into Matlab:
http://www.ece.uah.edu/~dwpan/course/ee614/data/Quant_Matrix.mat
then quantize the DCT coefficients, where the quantized coefficients
quant_coef (i, j) = round (*dct_coef*(i, j) / *Quant_Matrix*(i, j)).
Show the quantized coefficients.

(d) De-quantization using the same Quantization_Matrix, i.e.,
de_quant_coef(i, j) = *quant_coef* $(i, j) \times$ *Quant_Matrix*(i, j).
Show the de-quantized coefficients.

(e) Obtain a reconstructed block, *block_recon*, by first applying the inverse DCT (using `idct2` in Matlab) on the de-quantized coefficients obtained in (d), then convert the result of inverse DCT to *uint8* type. Show the content of *block_recon*.

(f) What is the distortion (in Mean Square Error, MSE) of the reconstructed block (*block_recon*) when compared with the original image block (*block*)?

(g) Repeat the above procedure for all other 8×8 blocks to obtain the entire reconstructed image (*I_recon*). Show the reconstructed image.
What is the distortion (in terms of PSNR in dB) of the reconstructed image when compared with the original image? Highlight or circle your answer.

(h) Enclose the Matlab scripts you wrote in the main submission file "HW5.pdf". Also, submit the M file "Q2.m" separately on Canvas.

3. (30 pts) In a lossy compression scheme, the input to the encoder is a random variable X, which takes value from the alphabet consisting of four distinct symbols: $\{0, 1, 2, 3\}$. Assume that X has the following distribution:

i	**0**	**1**	**2**	**3**
Prob ($X = i$)	1/8	1/8	1/4	1/2

Suppose the encoder encodes each input value by shifting out the least significant bit. The output of the encoder is then fed to the decoder. The decoder reconstructs X by doubling its input. If Y is the output of the decoder, then Y has a reconstructed alphabet of $\{0, 2\}$.
(a) Determine the numerical value of the conditional entropy $H(Y|X)$.
(b) Assume that the distortion is defined as $d(X = i, Y = j) = (i - j)^2$, where $i = 0, 1, 2, 3; j = 0, 2$. Determine D, the expected value of the distortion.

4. (50 pts) Image dataset compression and reconstruction using Sparse Autoencoder.
(a) Load in the training and test datasets:

```
>> XTrain = digitTrainCellArrayData;
>> XTest = digitTestCellArrayData;
```

(b) Train an autoencoder using the following command:
```
>> autoenc = trainAutoencoder (XTrain, hiddenSize, 'MaxEpoc', 100);
```
where hiddenSize = 144.

(c) Plot the weights of the autoencoder obtained in (b). Attach the screenshot of the plot.

(d) Use the "predict" function in Matlab to reconstruct the test image dataset using the trained autoencoder obtained in (b). Name the reconstructed dataset 'XReconstructed'.

(e) Use the 'subplot' command to plot the first image in the two datasets: 'XTest{1}' and 'XReconstructed{1}' side by side. Attach the screenshot of the plot.

(f) Measure the MSE (mean square error) between the entire datasets: 'XTest' and 'XReconstructed'.

(g) Repeat steps (b) through (f) for two other hiddenSize values of 36, and 9, respectively.

(h) Fill in the table below with your results.

hiddenSize	144	36	9
MSE			

(i) Enclose the Matlab scripts you wrote in the main submission file "HW5.pdf". Also, submit the M file "Q4.m" separately on Canvas.

(j) Comment on your results.

5. (80 pts) Independent Study on Arithmetic Coding.

(a) Read the following paper: "Arithmetic coding for data compression" by Witten *et al.*, https://dl.acm.org/doi/pdf/10.1145/214762.214771, where there is an illustrating example based on an information source with six distinct symbols (refer to the alphabet shown in Table I. "Example Fixed Model for Alphabet (a, e, i, o, u, !)" in the above paper). Suppose we want to apply arithmetic coding on the following message consisting of five symbols: '***aiou!***'. Sketch the arithmetic coding stages similar to Figure 1b in the above paper. Indicate clearly the lower and upper limit values of the sub-intervals in each stage.

(b) Use the Matlab function *arithenco* to encode the *lena* image http://www.ece.uah.edu/~dwpan/course/ee614/data/lena.mat.
How many distinct symbols (pixel values) does the image have? What is the size (in bits) of the coded bitstream?
Hint: Type "edit arithenco" on Matlab to see an example of how to use this function.

(c) What is the redundancy (relative to the entropy of the source image)?

(d) Use the Matlab function *arithdeco* to decode the coded bitstream to recover the original image. Conduct a lossless check. Show the screenshot of the lossless check result.

(e) Fill in the table below with your results.

Number of distinct pixel values in the image	Coded Bitstream Size (in bits)	Entropy of the Image (bits/pixel)	Redundancy (bits/pixel)

(f) Enclose the Matlab script you wrote in the main submission file "HW5.pdf". Also, submit the M file "Q5.m" separately on Canvas.

(g) Comment on your results.