Computer Project 2: Image Transforms For Data Compression

Name: Linsong Zhan

Due Date: April 6th at 11:59pm

1. Introduction

1.1. Purpose of the computer project

The goal of this computer assignment is to study and compare two different image transforms, namely 2-D DCT and DWT, for the purpose of image data compression.

1.2. What will be accomplished or carried out?

The "Lena" image is used for this assignment.

For DCT approach, I will first partition Lena image into non-overlapping blocks using two different blocks sizes (8x8 and 16x16) and apply DCT to each block. I will concentrate on just the top-left corner of each block and zero out everything else. The image can be reconstructed by applying IDCT to the "filtered" coefficient matrix. I will compare the results of different block partitions' reconstruction using signal-to-noise ratio as well as visual evaluation of the results. After that, I want to calculate the approximate Bit rates (bits/pixel) for different percentages of zeros in the coefficients. I will try to find the optimal solution in the SNR vs Bit Rate.

For DWT approach, I will first apply 2-D DWT to the same image and display the results for two and three-level decompositions and two choices of wavelets (Daubechies (db) and Symlet (sym) here). To evaluate the result, I will then present the histograms of the sub-images and compare the data reduction ability of the two transforms in both DCT and DWT approaches. At the end, I will compare the best result of each of them.

2. Theory

Algorithm for image data compression using 2-D DCT

<u>Step 1</u>: Compute the 1-D DCT Matrix

$$C(i,j) = \begin{cases} \dfrac{1}{\sqrt{N}} & i = 0, j \in [0, N-1] \\ \sqrt{\dfrac{2}{N}} \cos[\dfrac{(2j+1)i\pi}{2N}] & i \in [1, N-1], j \in [0, N-1] \end{cases}$$

For an 8x8 block, N=8.

Step 2: Apply DCT on each block of the image, note that since the DCT is designed to work on pixel values ranging from -128 to 127, the original block is "leveled off" by subtracting 128 from each entry [1]. If $x_{(0,0)}$ is an 8x8 block from the very upper-left corner of an image. This process can be expressed as

$$x_{(0,0)}(i,j) = x_{(0,0)}(i,j) - 128, \qquad i \in [0, N-1], j \in [0, N-1]$$

Now the DCT is accomplished by matrix multiplication as

$$X_{(0,0)} = Cx_{(0,0)}C^T$$

This matrix $X_{(0,0)}$ contains 64 DCT coefficients, $X_{(0,0)}(i,j)$, where i and j range from 0 to 7. The DC coefficient, $X_{00}(0,0)$, is relatively large in magnitude, and the AC terms become lower in magnitude as they move farther from the DC coefficient.

Step 3: Since we are not doing the complete quantization of DCT coefficients, I will just reduce the coefficients of each block by just concentrating on the top-left corner of each block. Basically, I will grab a 4x4 sub block out of each block and zero out everything else. I call this new reconstructed matrix as $XR_{(0,0)}$.

Step 4: Apply the IDCT to matrix $XR_{(0,0)}$ and round each element of the IDCT result to nearest integer. After adding 128 to each element of the result, I get the decompressed version of the original 8x8 block $x_{(0,0)}$ that can be written as

$$D_{(0,0)} = round\left(C^T XR_{(0,0)} C\right) + 128$$

Step 5: To get the complete decompressed image, we can repeat all the previous steps for all the other blocks of the image. For example, the complete decompressed image D for the 8x8 block partition is written as

$$D = \begin{bmatrix} D_{(0,0)} & \cdots & D_{(0,63)} \\ \vdots & \ddots & \vdots \\ D_{(63,0)} & \cdots & D_{(63,63)} \end{bmatrix}$$

Step 6: Get the complete decompressed image for the 16x16 block partition by repeating all the previous steps. Calculate the SNR for both block partitions and compare the results. The equation of SNR is given as

$$SNR_{dB} = 10 \log_{10} \frac{\sigma_o^2}{\sigma_e^2}$$

Where σ_o^2 is the variance of original image and σ_e^2 is the variance of error image.

Step 7: Plot the figure of SNR vs. Bit Rate by changing the percentage of zeros.

For DWT approach, I will use the 'wavemenu' command which performs lots of useful features about DWT in MATLAB.

3. Result and Discussion

DCT Approach

I will do the DCT and filter process on the Lena.mat image.

Figure 1 - Lena

Each 8x8 block is dealt with the DCT, leading to the image shown in Figure 2. Each 16x16 block is also dealt with the DCT, leading to the image shown in Figure 3.

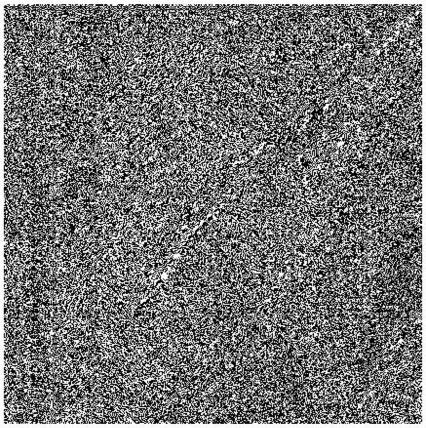


Figure 2 - DCT of Lena for 8x8 blocks

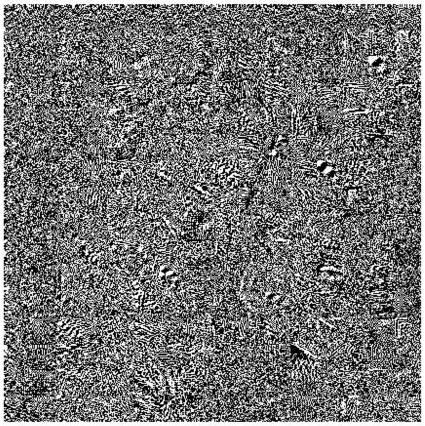


Figure 3 - DCT of Lena for 16x16 blocks

The data compression of original image is executed by keeping the top-left corner 4x4 sub block of each block and zeroing everything out. The image after this sort of filtering takes much less space to store.

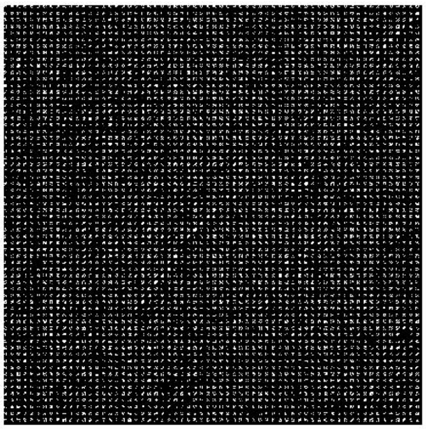


Figure 4 - Filtered DCT of Lena for 8x8 blocks

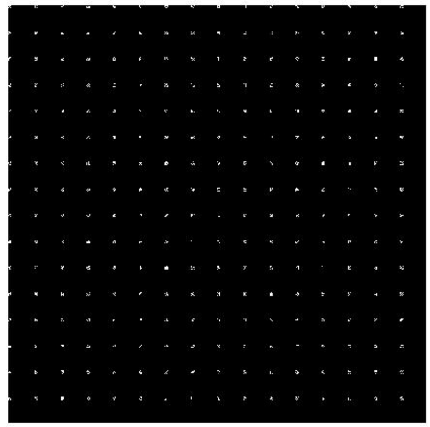


Figure 5 - Filtered DCT of Lena for 16x16 blocks

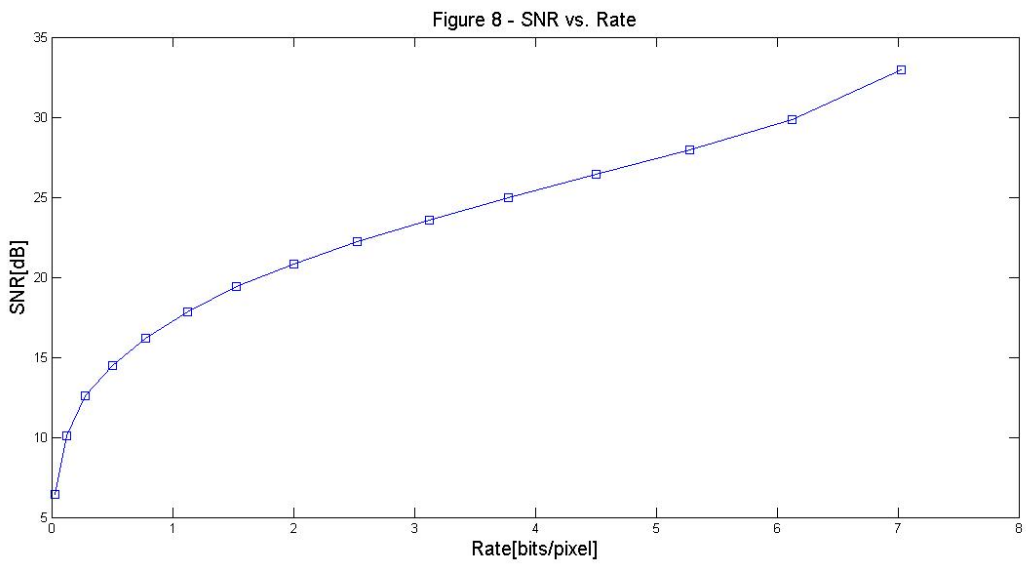
The image can now be decompressed using the inverse discrete cosine transform. Figure 6 shows a properly reconstructed image. While Figure 7 shows a terribly reconstructed image since we can see not only lots of edges missing but also some blocking artifacts near the center of the face. In the measurement of SNR, the SNR of Figure 6 is also appeared to be greater than the one of Figure 7.

Figure 6 - Decompressed image for 8x8 blocks - 75% Zeros

Figure 7 - Decompressed image for 16x16 blocks - 98.44% Zeros

Bit Rate Performance of DCT approach for 16x16 blocks

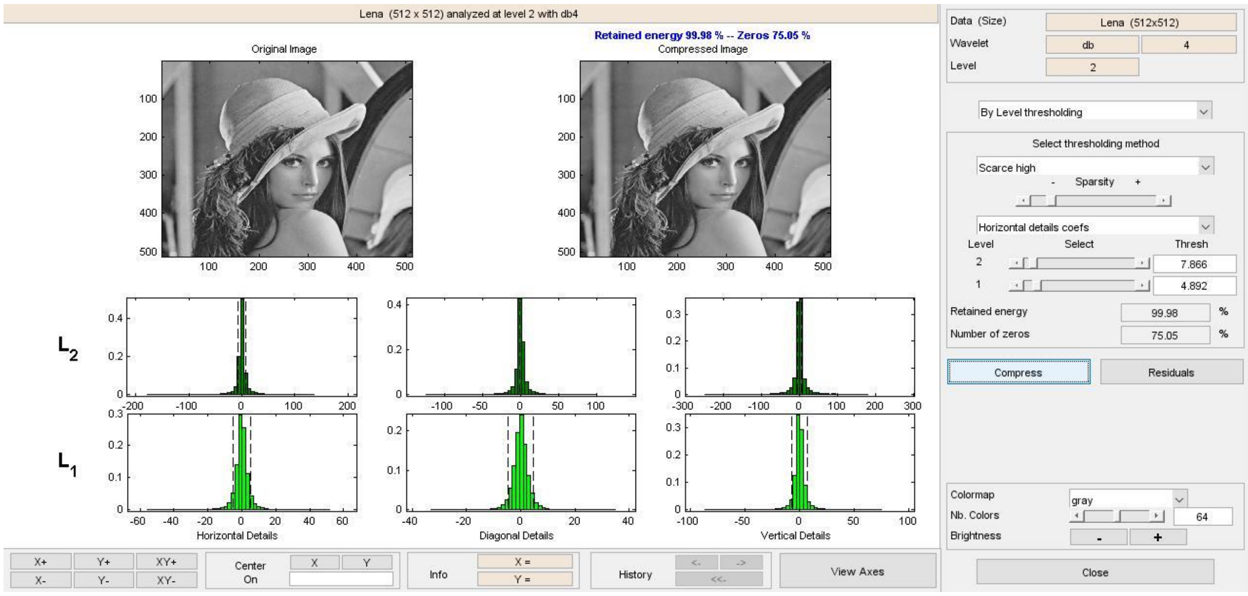
The rates are calculated based on grabbing numerous sizes of the top-left corner KxK ($K \in [1{,}15]$) sub block out of each 16x16 block. As you can see, Figure 8 shows that the SNR will not be improved that much anymore as we increase the Rate (number of coefficients retained) to a point of diminishing return. There are 15 points in the figure in total, as the optimal point for compression design appears near the 9th point in the middle.

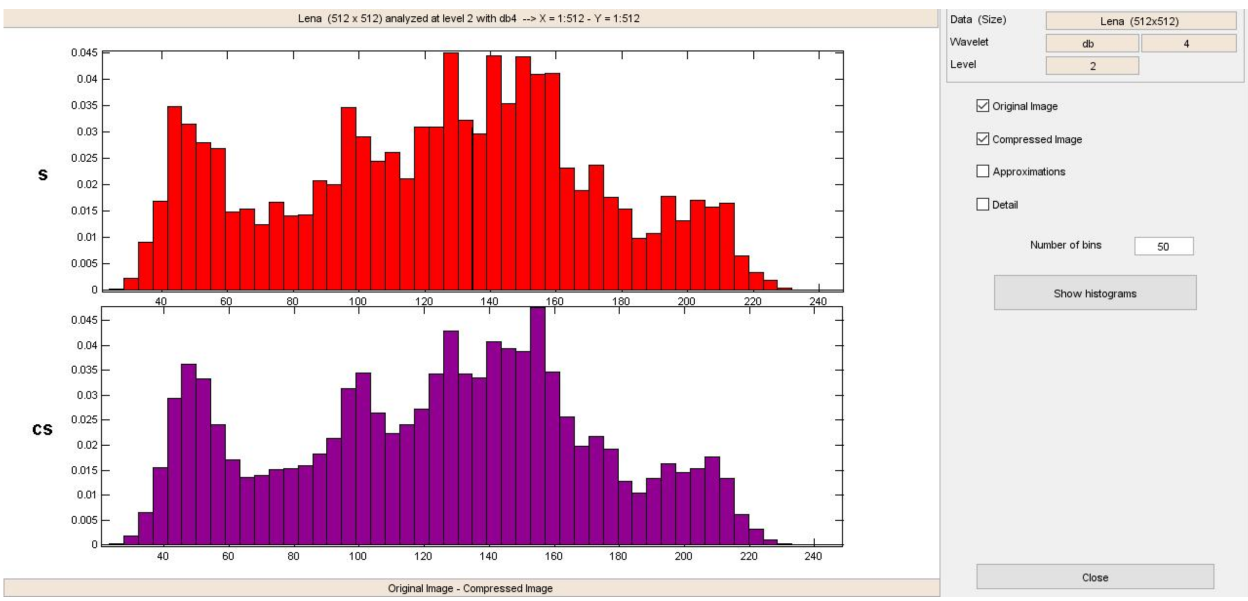


DWT Approach

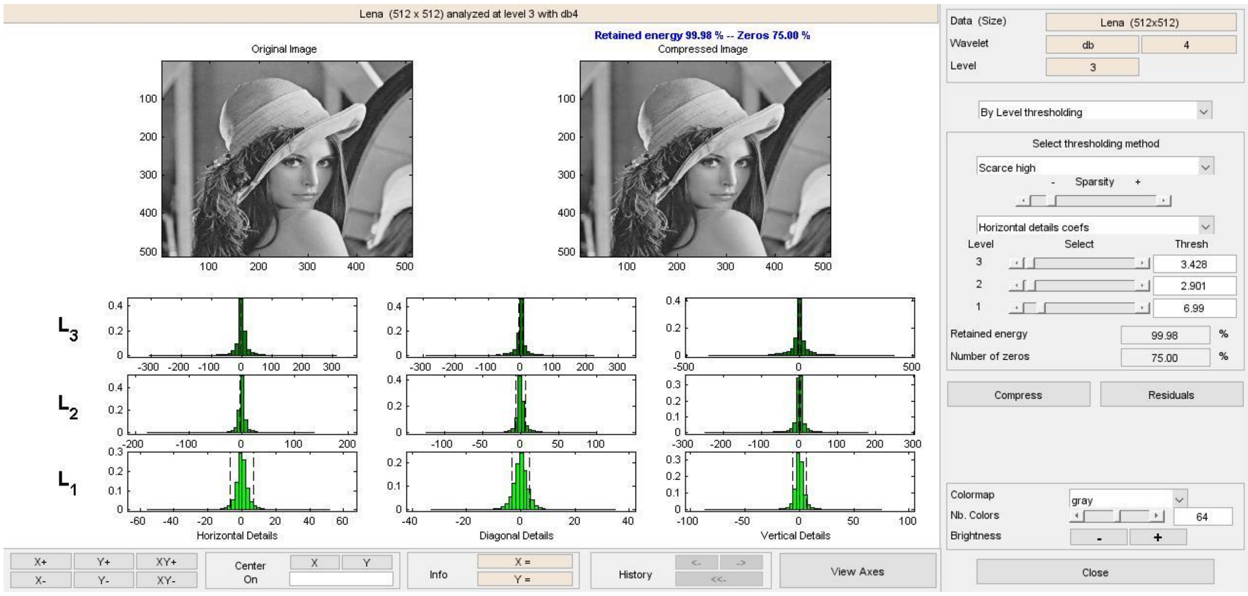
To compare with the best result of DCT approach easily, which is shown in Figure 6, I have set up the percentage of zeros for each figure produced by DWT to 75%.

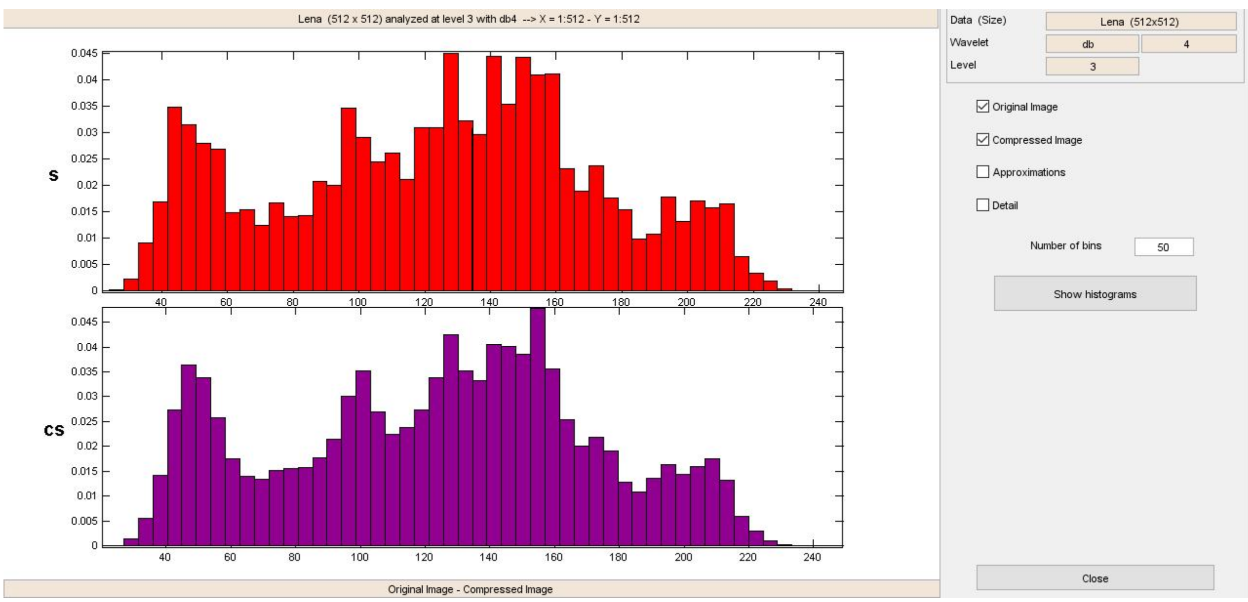
For Lena analyzed at level 2 with db4, the SNR_{dB} is equal to $27.0806\ dB$



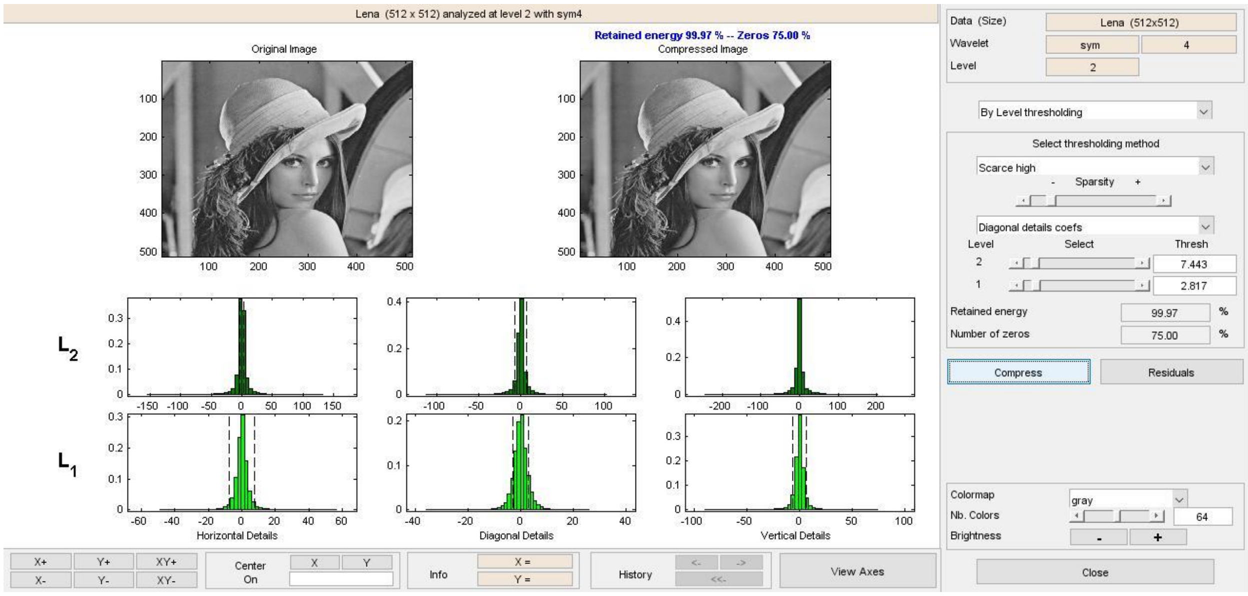


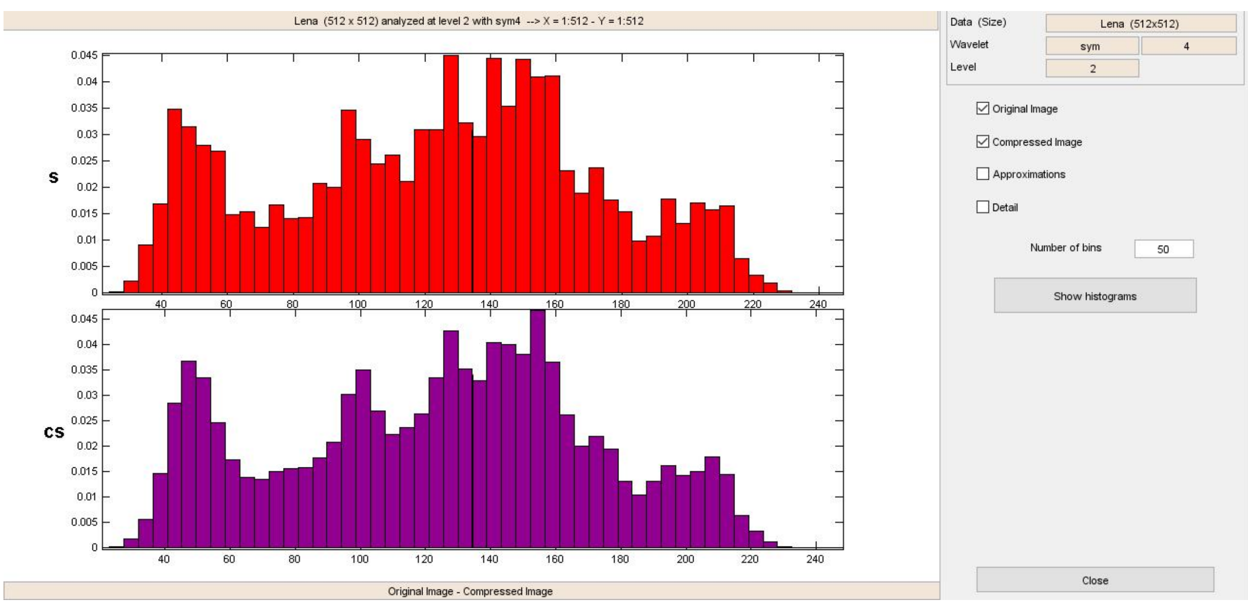
For Lena analyzed at level 3 with db4, the SNR_{dB} is equal to $27.3440\ dB$



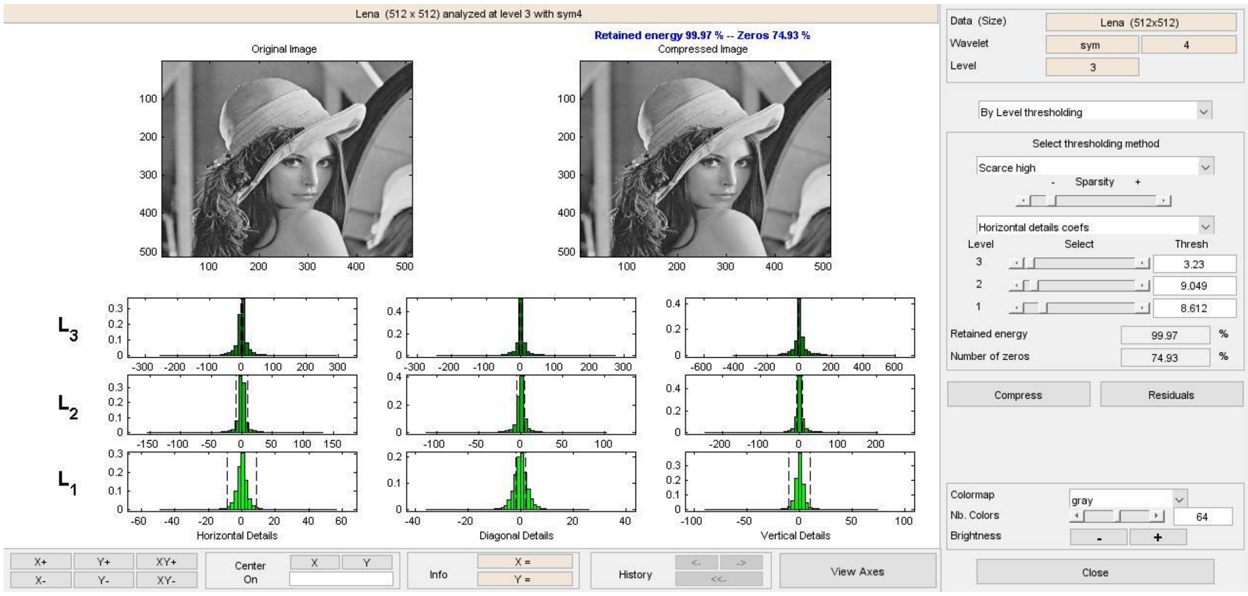


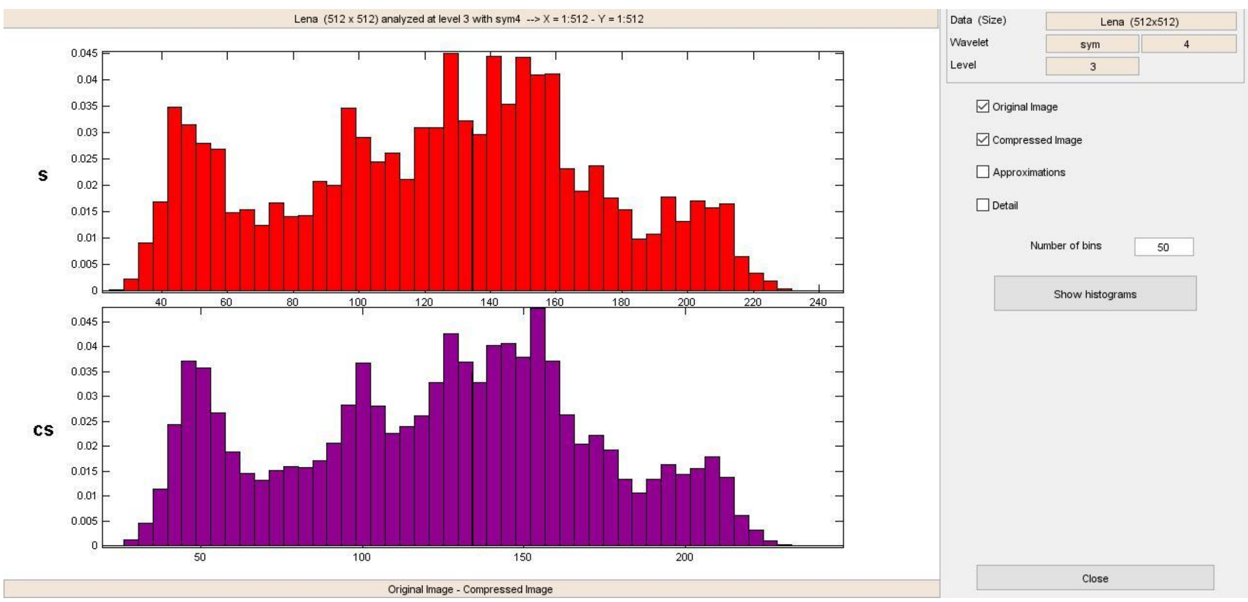
For Lena analyzed at level 2 with sym4, the SNR_{dB} is equal to 26.8903 dB





For Lena analyzed at level 3 with sym4, the SNR_{dB} is equal to $25.5671\ dB$





In conclusion, the best result of DWT is the “Lena analyzed at level 3 with db4” with the $SNR_{dB} = 27.3440\ dB$. This result is greater than the one in Figure 6. Thus, the DWT approach is a more efficient way to compress and store the image compared with the DCT approach. The coarse edges of the wavelet also show a better image because they dispel the blocking artifacts of DCT.

4. Conclusion

In this assignment, I performed two kinds of data compression technical using the DCT approach and the DWT approach. The DCT is widely used for lossy compression. The different block partitions of the image using DCT will lead to different reconstructed images. In general, the larger size partitions tend to show blocking artifacts easily. The DWT is also used for lossy compression with an advantage that it can capture both frequency and location information.

5. Reference

[1]. https://www.math.cuhk.edu.hk/~lmlui/dct.pdf