Compe 565, Semester 2021

HW 2: Project on JPEG based Image Compression

Prepared by

Ethan Nagelvoort (821234668)

E-mail: enagelvoort@sdsu.edu

Electrical and Computer Engineering

San Diego State University

Table of Contents

List of figures

Introduction

Quantization helps programmers make a single image more compressible. Quantizing an image involves the use of Discrete Cosine Transformation (DCT) values, which helps remove spatial correlation among adjacent pixels. This allows for better entropy coding. These two concepts are a part of the process of encoding and decoding of images. Encoding will allow an image to be more compressible and decoding will return it to its original state.

In this homework, I utilized MatLab to encode an image which involved computing the 8x8 DCT transform coefficients of the luminance and chrominance components of the “Flooded House” image from homework1. I then quantized the image using the luminance and chrominance quantizer matrices given in lecture. Then I had to decode this image to return it back to its original form. I did this by computing the inverse quantized images and reconstructing the image by computing Inverse DCT coefficients.

Procedural Section

1. Encoder - Compute the 8x8 block DCT transform coefficients of the luminance and chrominance components of the image.

   I first imported the “Flooded House” image which will be used throughout this homework. I then converted it to YCbCr and subsampled Cb and Cr to 4:2:0. I then got the DCT by padding spatial 8x8 blocks and extracting the 2 blocks in the 6th row from top of the luminance component. I did this by looping from rows 41 to 48 and columns 1 to 8 and 9 to 16. I then displayed these blocks to get the DCT matrices for each block.

2. Encoder - Quantize the DCT image by using the JPEG luminance and chrominance quantizer matrix from the lecture notes.

   I then got the DC DCT coefficients by quantizing the luminescent component with the luminescent quantization matrix. I also quantized the chromance component with the chrominance quantization matrix. I then extracted the needed blocks and obtained the DC DCT coefficients through (1,1). I then performed two zigzag operations on both blocks to obtain the needed AC DCT coefficients.

3. Decoder - Compute the inverse Quantized images obtained last step.

   In order to get the inverse blocks, I multiplied the luminescent quantization matrix and chrominance quantization matrix to the blocks to reverse the effects of when I divided them into the blocks. I then displayed these inverse blocks.

4. Decoder- Reconstruct the image by computing Inverse DCT coefficients.

   I then used the DCT coefficients and linear interpolation to reconstruct the image back into its normal RBG image. I also got an error image by subtracting the originals y component to the reconstructed images y component. I then also got the PSNR by calculating MSE and then PSNR by using the formulas found in lecture.

Results

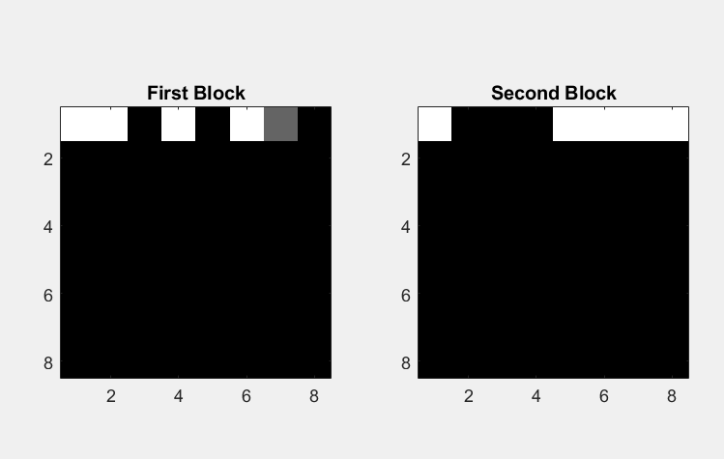

Figure 1: DCT transformed image blocks of the first 2 blocks in the 6th row from top for the luminance component.

```
The DCT coefficient matrix of block 1:
   1.0e+03 *

    1.6880    0.1472   -0.0323    0.0142   -0.0060    0.0043    0.0004   -0.0010
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0

The DCT coefficient matrix of block 2:
   1.0e+03 *

    1.7120   -0.2099   -0.1457   -0.0611    0.0180    0.0669    0.0820    0.0525
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
```

Figure 2: DCT coefficient matrices of both block

```
The DC DCT coefficient of quantized block 1:
105
The DC DCT coefficient of quantized block 2:
107
```

Figure 3: DC DCT coefficient for both the quantized blocks.

```
The AC DCT coefficients of block1:
  Columns 1 through 18

    13     0     0     0    -3     1     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 19 through 36

     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 37 through 54

     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 55 through 63

     0     0     0     0     0     0     0     0     0
```

```
The AC DCT coefficients of block2:
  Columns 1 through 18

   -19     0     0     0   -15    -4     0     0     0     0     0     0     0     1     2     0     0     0

  Columns 19 through 36

     0     0     0     0     0     0     0     0     2     1     0     0     0     0     0     0     0     0

  Columns 37 through 54

     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 55 through 63

     0     0     0     0     0     0     0     0     0
```

Figure 4: AC DCT coefficients for both blocks

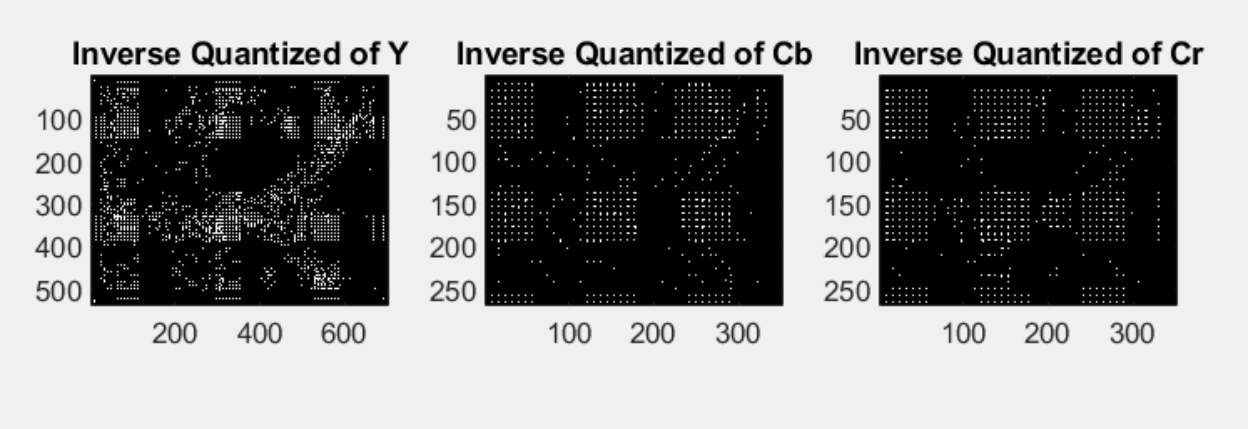

Figure 5: Inverse of the quantized Y, Cb, and Cr blocks

Figure 6: The original RGB image vs the reconstructed image

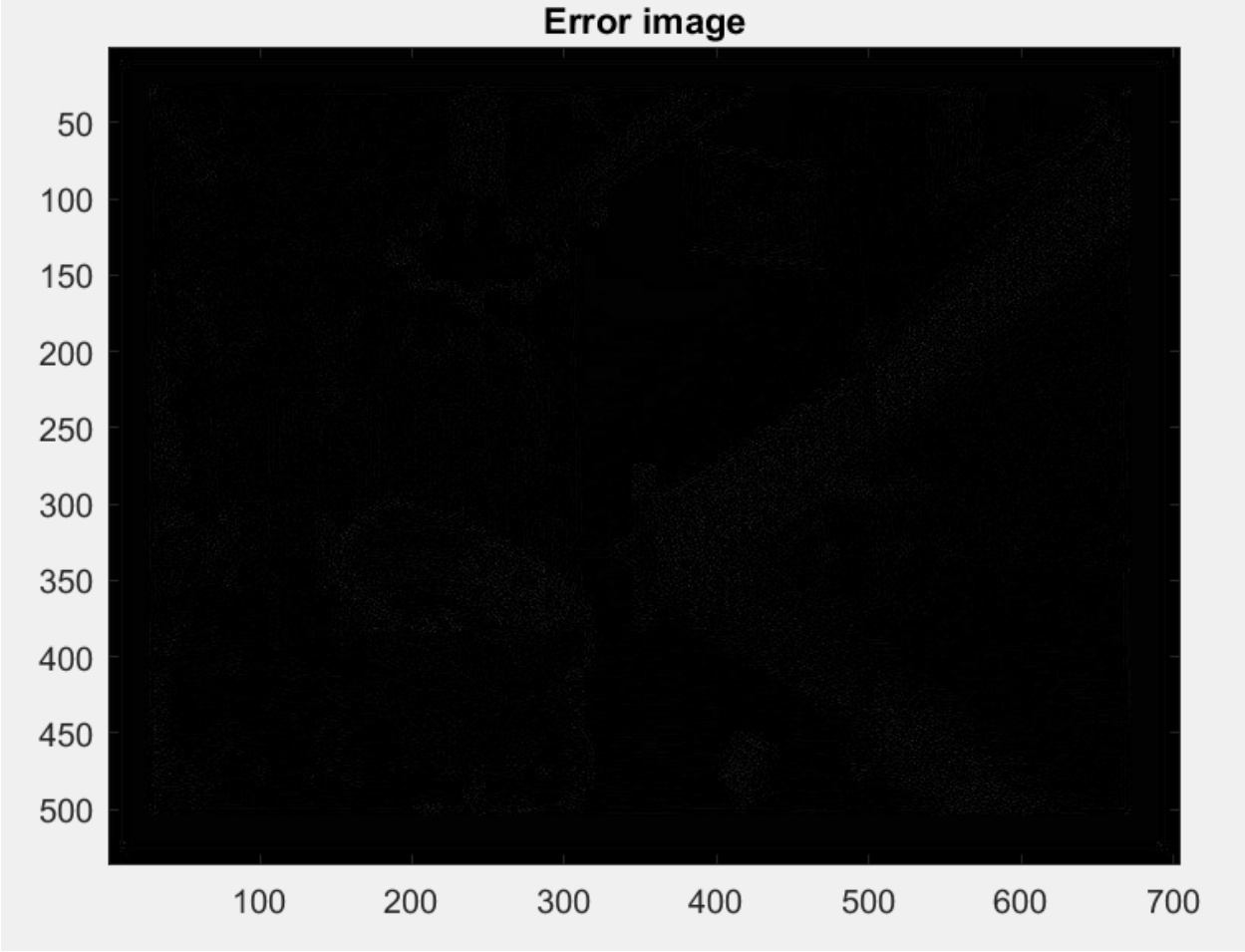

Figure 7: Error Image

The image here is very dark and so not much error has occurred when reverting the image back into RGB.

```
The PSNR value of the luminance component is:
   33.1427
```

Figure 8: PSNR value of the luminance component

PSNR is around 33, which is typical for normal images, so nothing drastic has occurred to the reconstructed image.

Conclusion

In this assignment, I learned that quantizing images can help with image compression. This means that DCT values are very important to having an easy compression. I also learned how to obtain an error image to see how badly an image was quantized and changed in the process. With this, I can now determine how effective was the decoding process a programmer used after he encoded it to compress the image. I also learned how to quantize using DCT coefficients and the zigzag method.

Reference

[1] S. Kumar, “Lecture Topic #4: Image Compression”, Compe 565, 2021

[2]S. Kumar, “Lecture Topic #5: Video Compression”, Compe 565, 2021