

# Homework 4 : Entropy Coding

Name: 周睦鈞

Student ID: 311553060

## Spec:

- Run length encoding and run length decoding
  - 8x8 block-based DCT coefficients of "lena.png."
  - Quantize the coefficients with the two quantization tables.
  - Use a raster scan to visit all 8x8 blocks in these images.
  - Do the run length encoding by using a zigzag scan to visit all pixels in one block.
  - Do the run length decoding and IDCT to recover the image.
  - Compare the encoded image sizes with the two quantization tables.
- Deadline: 2024/11/11 1:19 PM
  - Upload to E3 with required files :
  - **VC\_HW4\_[student\_id].pdf**: Report PDF
  - **VC\_HW4\_[student\_id].zip**: Zipped source code (C/C++/Python/MATLAB) and a **README** file

10	7	6	10	14	24	31	37	10	11	14	28	59	59	59	59
7	7	8	11	16	35	36	33	11	13	16	40	59	59	59	59
8	8	10	14	24	34	41	34	14	16	34	59	59	59	59	59
8	10	13	17	31	52	48	37	28	40	59	59	59	59	59	59
11	13	22	34	41	65	62	46	59	59	59	59	59	59	59	59
14	21	33	38	49	62	68	55	59	59	59	59	59	59	59	59
29	38	47	52	62	73	72	61	59	59	59	59	59	59	59	59
43	55	57	59	67	60	62	59	59	59	59	59	59	59	59	59

Quantization table 1

Quantization table 2

Ref 1: <https://www.youtube.com/watch?v=Q2aEzeMDHMA>

Ref 2:  
<https://q-viper.github.io/2021/05/24/coding-run-length-encoding-in-python/>

# Explanation:

1. Divide the image into multiple 8x8 blocks
2. According to the quantization table, we apply quantization to each block after it has been transformed using the DCT
3. Do the run length encoding by using a zigzag scan to visit all pixel for each quantization block
4. Do the run length decoding and IDCT to reconstruct the image

```
def process_image_blocks(image, quant_table, block_size=8, output_dir="output"):
    os.makedirs(output_dir, exist_ok=True)
    h, w = image.shape
    processed_image, encoded = np.zeros_like(image, dtype=np.float32), [[0, 0]]

    for i in range(0, h, block_size):
        for j in range(0, w, block_size):
            block = np.float32(image[i:i+block_size, j:j+block_size])
            quantized_block = quantize(dct_1d(block), quant_table)
            zigzag, values, counts = zigzag_scan(quantized_block), *run_length_encode(zigzag_scan(quantized_block))
            encoded.extend(zip(counts, values))
            decoded_block = idct_1d(de_zigzag_scan(run_length_decode(values, counts)))
            processed_image[i:i+block_size, j:j+block_size] = decoded_block

    np.savez(os.path.join(output_dir, "encoded.npz"), np.array(encoded))
    cv2.imwrite(os.path.join(output_dir, "encoded.png"), np.array(encoded))
    return processed_image
```

# Results:

Quantization Table 1:

File size of original lena.png: 167460 bytes

File size of encoded file: 58936 bytes

PSNR of lena.png and reconstructed\_img: 37.402

Quantization Table 2:

File size of original lena.png: 167460 bytes

File size of encoded file: 35043 bytes

PSNR of lena.png and reconstructed\_img: 35.245

Original



Table1



Table2

