Introduction to Image and Video Processing
Lab 6: Compression, DCT

Spring 2022

1 DCT based compression and reconstruction

- Read an image and compute its 2D DCT in a blockwise manner, using 8×8 blocks.
- Create a matrix that represents the zig-zag ordering of the coefficients using numbering that will keep the most significant ones.
- Keep part of the most significant DCT coefficients and reconstruct the image with only these.

2 IGS

Consider an image with values $[9, 15, 15, 32, 50, 56]$. Express these intensity values with 6 binary digits and create an IGS code for them.

3 Huffman, RLC

3.1 Huffman

Create a code for the following "image" with Huffman coding. At the last branch, assign the two digits in the order $0, 1$ (not $1, 0$), so we can have the same solutions. What is the average codeword length of your resulting code? (*Hint: Each gray level value* $0, 1, ..., 6$ *needs to be encoded. To this end, you will need to find how many times each gray level appears, i.e. find its probability of appearance and then use it in the Huffman code.*)

0	0	0	0	0	0	0	0
0	0	1	1	2	3	3	3
0	1	1	3	3	3	4	4
0	1	3	3	5	5	4	4
0	2	3	3	5	5	5	4
0	0	2	3	3	4	6	6
0	0	0	2	2	3	4	4
0	0	0	0	0	0	0	0

3.2 RLC

Create a code for the previous "image" with RLC coding applied to each row. So, for each row, create the necessary RLC pairs. What is the average codeword length of your resulting code? (*Hint: find the maximum run length in the image to determine the minimum number of bits you will use to represent each run length pair.*)

4 Variable length coding

If an image has undergone histogram equalization, is there a point in using Variable length coding for it? (*Hint: Think of the probability of appearance for each symbol in an equalized image, and think of when we use VLC.*)

Answer: Variable length coding won't lead to improvements in an image that has an equalized histogram, because the histogram equalization means all symbols have the same probability. VLC leads to compression if some words have higher and some have lower probability of appearance, but this is no longer the case after histogram equalization.

5 JPEG

In this exercise you will create JPEG compression yourselves to a *grayscale* image. You don't need to implement the entire codec, only the steps listed below (which exclude encoding):

- Read the image, subtract 128 from the pixel intensities and apply blockwise DCT on 8×8 blocks.
- Quantize the DCT coefficients with quality factor 50, using the quality matrix given below.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

- Zigzag scan the coefficients.
- Create the compressed image (dequantize the coefficients and apply inverse blockwise DCT).
- Repeat the above steps to the compressed image, but with QF 75, to get a twice-compressed image.
- Display and compare the histograms of all DCT coefficients of the first and second compressed images.
- Display and compare the histograms of the first ten DCT coefficients of the first and second compressed images.