

Digital Video Technology

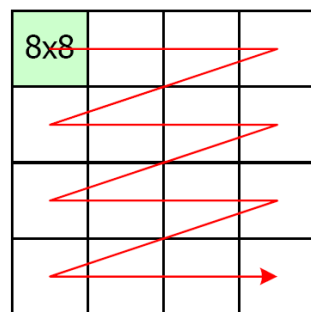
Homework #3 – Entropy Coding

2013/10/08

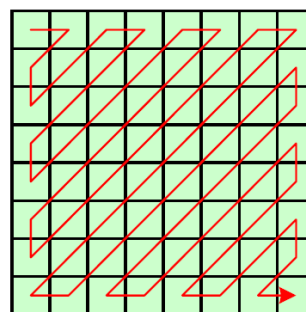
Write a program for **entropy coding** and **decoding** including re-ordering run-length coding (RLC) and variable length coding (VLC) to do data compression of data you had at homework #2 after DCT (use **8x8 block-based DCT** on hw2-step5, the quantization method could be a) 16-bit for DC and 8-bit for AC or b) JPEG quantization method (page 68–70 in "03.Compression-fundamental-2013.pdf"). Please describe your quantization method in the report). Ten test images used for this homework are **lena64.raw**, **pepper64.raw**, **baboon64.raw**, **gra1.raw**, **gra2.raw**, **wildcard.raw**, **triangle.raw**, **circle1.raw**, **circle2.raw**, **circle3.raw**. Use the probability table by investigating your own data.

RLC and RLD

1. Refer to page 31, 51 in "03.Compression-fundamental-2013.pdf".
2. Use raster scan to visit all 8x8 blocks in these images.
3. Do RLC by using zigzag scan to visit all pixels in one block shown on page 51, and generate run-length sequence shown on page 31.



Block level scan



Pixel level scan

4. Save the RLC file named "**filename.rlc**" (text file), the file content should look like this:
(0,65)(2,-3)(0,5)(1,-2)(4,2)(0,-4)(3,1)...
(0,71)(3,2)(0,2)(7,3)(1,2)(0,-2)(3,3)...
One line for one 8x8 block RLC.
5. Write a function to do RLD.

(RLC + Huffman VLC) and (Huffman VLD + RLD)

1. Refer to page 16--21, in "03.Compression-fundamental-2013.pdf".
2. After RLC, generate your own Huffman code tree and probability table according to symbol probability of RLC result. Do **NOT** use existing probability table in some Standards.
3. Save the **only** compressed file named "filename.huf".
4. For decoding, read the "filename.huf" and decode without any information already saved in encoder. That is, you have to save enough information like probability table in your compressed file.
5. Use the data you get in hw2-step5 (block-based DCT). What is the compression ratio after each process? For example, if quantization method **a)** is selected, the compression ratios after quantization (16-bit for DC, 8-bit AC), RLC (16-bit for DC, 6-bit for run-length, 8-bit AC) and VLC (final file size) should be shown. On the other hand, if quantization method **b)** is selected, the compression ratios after quantization (10-bit for DC and AC), RLC (10-bit for DC, 6-bit for run-length, 10-bit AC) and VLC (final file size) should be shown. Please note the above bit number could be re-defined by yourself with detailed descriptions in the report. Compare the compression ratio of different images and try to explain the results.

(RLC + Arithmetic encoder) and (Arithmetic decoder + RLD)

1. Refer to page 22--24 in "03.Compression-fundamental-2013.pdf", and refer to "AC_Ian.pdf" in "Supplement" (**You must read this!!**) for arithmetic coding.
2. Change VLC (Huffman Coding) to Arithmetic Coding. Embed the supplementary **AC.c** and **AC.h** into your program.(AC.c and AC.h can be modified if needed.)
3. Save the **only** compressed file named "filename.ari".
4. For decoding, read the filename.ari and decode without any information already saved in encoder.
5. What are the advantages (compression ratio, runtime, memory usage ... etc.)? Please explain them.

Requirements:

1. Deadline: **2013/10/28 11:59 PM**
-10 points / day
2. All the files need to be compressed as a single ZIP or RAR file.
Send this file to TA via FTP:
Address: **140.112.48.126** Port: **17199**
Account (password):
The same as the one used in the course website.
Examples of filename:
DVT_HW3_R02901001.zip
DVT_HW3_R02901001_Ver2.zip
3. Your program should generate "**filename.rlc**", "**filename.huf**", "**filename.ari**", and it should decode your compressed data back to image correctly, that is, your program should generate "**filename.idct.raw**" (without quantization), "**filename.rlc.raw**", "**filename.huf.raw**", "**filename.ari.raw**" and compare them.
4. Required files
 - a. **Report**, in **PDF** format
(With pictures, you can use compressed/downscaled images in report)
 - b. **Source code** (**C/C++**), all TA needs are related .cpp/.c/.h
Do NOT send entire project to TA!!
 - c. **Executable file** (*.exe)
 - d. **Text document** (*.txt): a "readme" file, which may describe how to execute your program, the compiler you use, and your output file names
5. Be sure to re-check requirements before sending your homework.
6. Any further question, please email to TA.
(吳柏辰, pcwu@media.ee.ntu.edu.tw)