資料壓縮 HW2

學號：113522118 姓名：韓志鴻

1. 運作流程
   1. **影像檔案與資料夾準備**：
      1. 讀取輸入圖像：根據指定資料夾(image/)，取得所有待壓縮的.raw格式圖像。
      2. 判斷圖像類型：根據檔名判斷為灰階或彩色(rgb)。
      3. 建立輸出資料夾：建立儲存重建.raw檔與.jpeg圖檔的資料夾(result / raw、result / jpeg)。
   2. **Quality Factor (QF) 參數設定：**
      1. 設定多組 QF 值：如 90、80、50、20、10、5，分別代表不同壓縮品質等級。
      2. 每一張影像都會針對每一組 QF 進行完整壓縮與重建流程。
   3. **動態產生量化表：**
      1. 根據 QF 調整量化表：利用 JPEG 標準計算方式，根據不同 QF 產生亮度(Y)與色差(C)的量化表，決定每一區塊頻率係數的壓縮強度。
      2. QF 越高，保留資訊越多，壓縮損失較小；QF 越低，壓縮率提升但畫質損失增加。
   4. **分塊與前處理**
      1. 將影像切分為 8×8 的區塊：無論灰階或彩色，每一通道都進行分塊。
      2. 像素值減 128：將像素資料中心化至 [-128,127]，準備做後續DCT。
   5. **離散餘弦轉換 (DCT)**
      1. 對每一個 8×8 區塊執行 2D DCT：將空間域像素轉換為頻率域資料，低頻資訊集中在左上角，高頻資訊集中在右下角。
   6. **量化**
      1. 將 DCT 結果進行量化：每個係數除以對應的量化表元素，四捨五入成整數，降低高頻資訊來達到壓縮目的。
   7. **Zigzag 掃描**
      1. 將 8×8 量化係數以 Zigzag 順序攤平成一維陣列：優先排列低頻(左上)到高頻(右下)，利於 Run-Length 編碼。
   8. **霍夫曼編碼與壓縮(Entropy Coding)**
      1. DC 分量編碼：以差分方式紀錄區塊間 DC 值變化，並根據標準 Huffman 表進行編碼。
      2. AC 分量編碼：進行 Run-Length 編碼壓縮連續零，再用 Huffman 表壓縮。
      3. 所有區塊資料串接成完整壓縮位元流。
   9. **影像重建**
      1. 反量化：將壓縮過的量化係數乘回原本的量化表。
      2. 反 DCT (IDCT)：將頻率域還原成空間域像素。
      3. 像素加回 128 並 clip 到 [0,255]：確保像素在正常範圍內，產生重建影像。
      4. 儲存重建的 .raw 及 .jpeg 圖像。
   10. **壓縮品質與效率評估**
       1. 計算 PSNR：評估壓縮前後畫質差異。
       2. 計算壓縮率：比較壓縮檔案與原始影像的大小，得到壓縮率與 bpp (每像素平均位元數)。
       3. 列印統計：輸出每張影像、每個 QF 下的 PSNR、壓縮後大小、壓縮比與 bpp 等資訊。
   11. **輸出結果與紀錄**
       1. 將所有重建與壓縮結果存檔。
2. 壓縮結果
   1. Baboon.raw (灰階)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **QF** | **PSNR (dB)** | **Bytes** | **Ratio** | **bpp** |
| 5 | 21.52 | 13762 | 19.05 | 0.420 |
| 10 | 23.42 | 23807 | 11.01 | 0.727 |
| 20 | 25.26 | 38172 | 6.87 | 1.165 |
| 50 | 28.23 | 64053 | 4.09 | 1.955 |
| 80 | 32.60 | 96413 | 2.72 | 2.942 |
| 90 | 37.12 | 127437 | 2.06 | 3.889 |

|  |  |
| --- | --- |
| **QF 5** | **QF 10** |
|  |  |
| **QF 20** | **QF 50** |
|  |  |
| **QF 80** | **QF 90** |
|  |  |

* 1. BaboonRGB.raw (彩色)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **QF** | **PSNR (dB)** | **Bytes** | **Ratio** | **bpp** |
| 5 | 20.38 | 28409 | 83.05 | 0.289 |
| 10 | 21.60 | 49141 | 48.01 | 0.500 |
| 20 | 22.68 | 80073 | 29.46 | 0.815 |
| 50 | 25.32 | 145455 | 16.22 | 1.480 |
| 80 | 29.97 | 241119 | 9.78 | 2.453 |
| 90 | 34.87 | 335122 | 7.04 | 3.409 |

|  |  |
| --- | --- |
| **QF 5** | **QF 10** |
|  |  |
| **QF 20** | **QF 50** |
|  |  |
| **QF 80** | **QF 90** |
|  |  |

* 1. Lena.raw (灰階)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **QF** | **PSNR (dB)** | **Bytes** | **Ratio** | **bpp** |
| 5 | 27.33 | 8310 | 31.55 | 0.254 |
| 10 | 30.41 | 12410 | 21.12 | 0.379 |
| 20 | 32.97 | 18808 | 13.94 | 0.574 |
| 50 | 35.83 | 32528 | 8.06 | 0.993 |
| 80 | 38.58 | 53505 | 4.90 | 1.633 |
| 90 | 40.89 | 76200 | 3.44 | 2.325 |

|  |  |
| --- | --- |
| **QF 5** | **QF 10** |
|  |  |
| **QF 20** | **QF 50** |
|  |  |
| **QF 80** | **QF 90** |
|  |  |

* 1. LenaRGB.raw (彩色)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **QF** | **PSNR (dB)** | **Bytes** | **Ratio** | **bpp** |
| 5 | 26.24 | 18816 | 125.39 | 0.191 |
| 10 | 28.61 | 27231 | 86.64 | 0.277 |
| 20 | 30.42 | 40928 | 57.65 | 0.416 |
| 50 | 32.52 | 72444 | 32.57 | 0.737 |
| 80 | 34.93 | 123950 | 19.03 | 1.261 |
| 90 | 37.23 | 183040 | 12.89 | 1.862 |

|  |  |
| --- | --- |
| **QF 5** | **QF 10** |
|  |  |
| **QF 20** | **QF 50** |
|  |  |
| **QF 80** | **QF 90** |
|  |  |

1. 執行方式
   1. python main.py
   2. 執行後會產生result資料夾，將結果存成.raw檔並放在raw子資料夾。另外為了方便觀察，也存成.jpeg檔並存入jpeg子資料夾。
2. 參考資料
3. Microsoft Word - T081E.DOC, <https://www.w3.org/Graphics/JPEG/itu-t81.pdf>
4. 深入学习JPEG压缩原理与过程, <https://blog.csdn.net/eieihihi/article/details/143068225>
5. DC系数和AC系数, <https://blog.csdn.net/qingkongyeyue/article/details/58130001>