

作業 2

課號: C06041

課名: 數位影像處理

教師: 唐之瑋

學號末三碼: 009

姓名: 鄧翔冠

一、實驗步驟說明

本次實驗步驟整體流程如圖 3.2，詳細流程圖可參考圖 3.3～圖 3.9，本次實驗開始前，於網路上下載 Lena 的灰階圖片(*.bmp)，讀取其影像資料，開始實作影像處理的演算法，如 DCT(Discrete Cosine Transform)、Filter Mask 等等，首先會先測試演算法的程式是否正確，參考測試實驗流程圖 3.1，在 HW#2 說明的第 6 頁矩陣，經過 shift、DCT、Quantization、Inverse Quantization、Inverse DCT、shift 等等，將測試矩陣代入演算法以檢驗是否正確。

當測試矩陣完成測試後，代入實際的圖片進行運算，可得出本次實驗的圖片，如果得出的圖片不符合預期，會再回去檢查演算法擴充到整張圖片是否有誤，或是圖片處理少了哪一個步驟，最後計算圖片的 MSE(Mean Square Error) 和 PSNR(Peak Signal To Noise Ratio)，探討轉換後的圖片和原始圖片的失真情形。

二、學習目的

本次學習目的主要為如何將圖片轉換到頻域上，同時了解到基本的 JPEG 格式、Filter Mask，而轉換過程中一定會造成一些誤差，可能原因為計算上面的精度，實驗最後的 PSNR 為探討圖片的訊雜比，可以從這樣數據約略了解到圖片的失真情形，經過此次實驗，對於圖片轉到頻域、JPEG 格式和分析圖片的失真程度，有基本的能力。

三、 實驗步驟流程圖

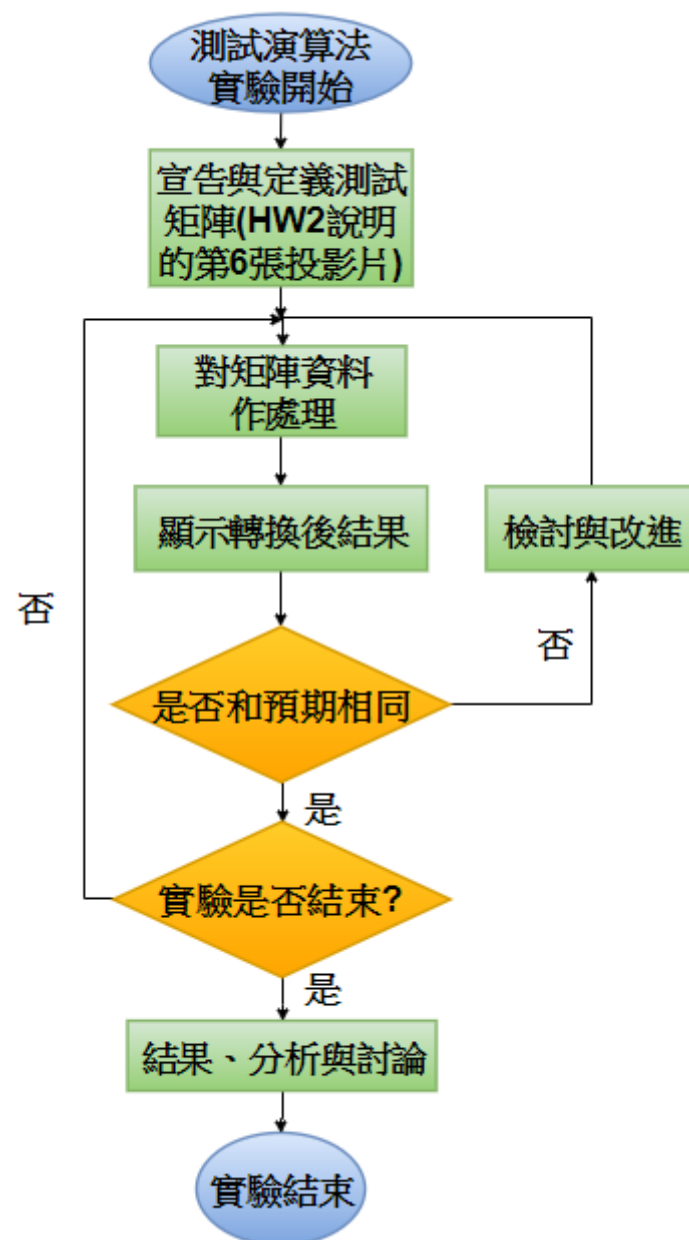


圖 3.1 測試實驗流程圖

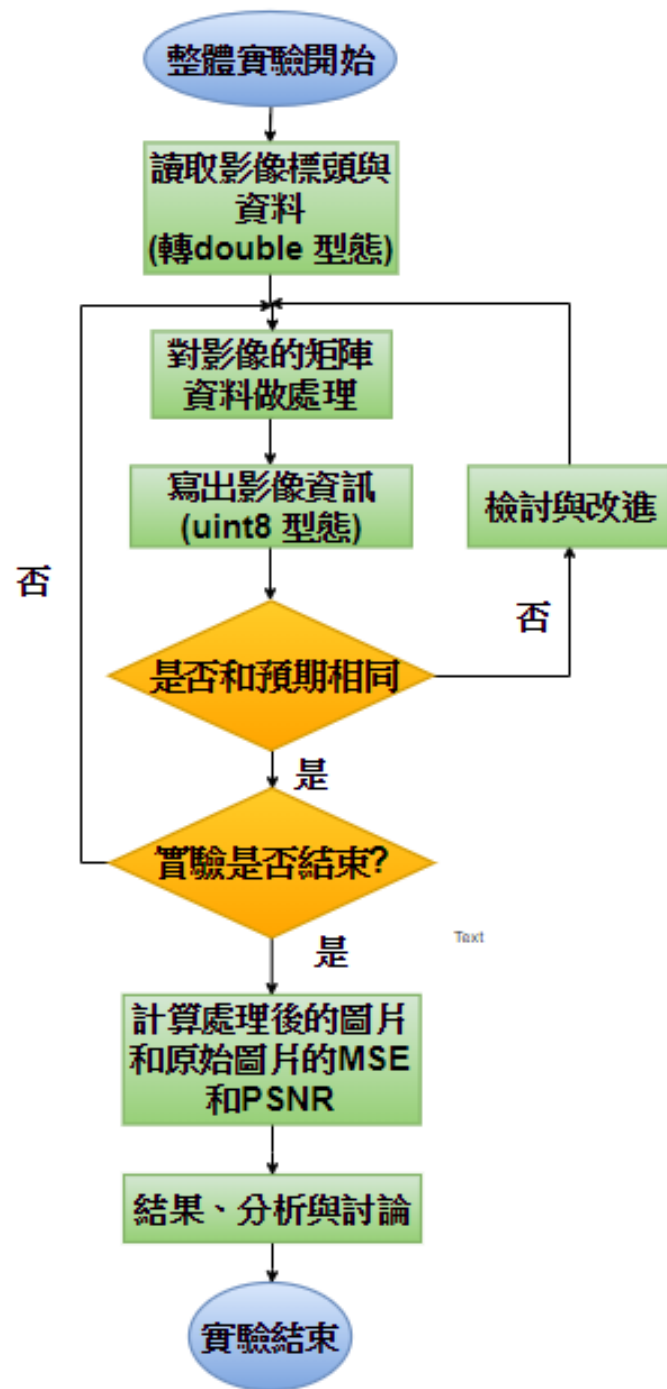


圖 3.2 實驗整體流程圖

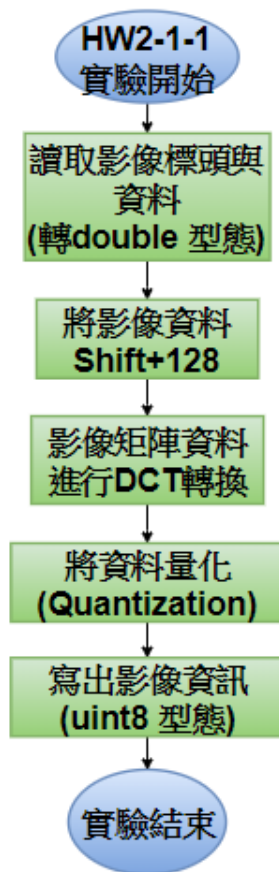


圖 3.3 HW2-1-1 處理流程圖

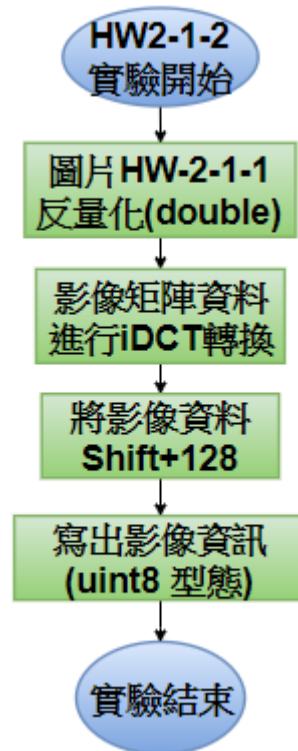


圖 3.4 HW2-1-2 處理流程圖

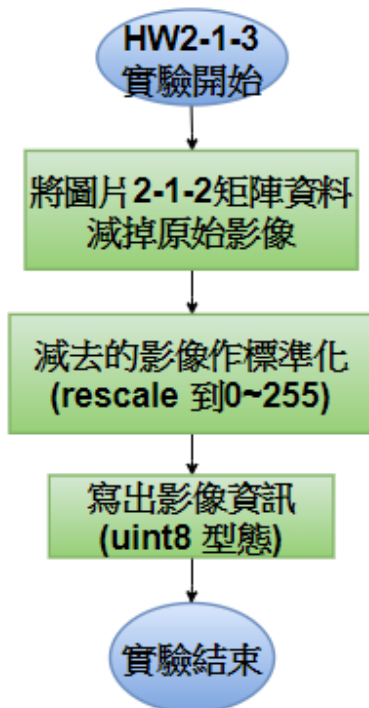


圖 3.5 HW2-1-3 處理流程圖

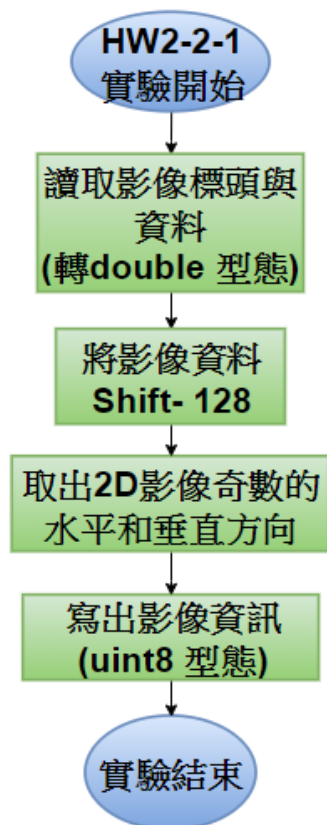


圖 3.6 HW2-2-1 處理流程圖

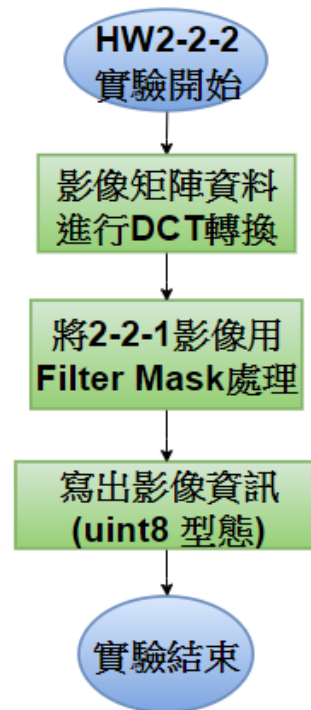


圖 3.7 HW2-2-2 處理流程圖

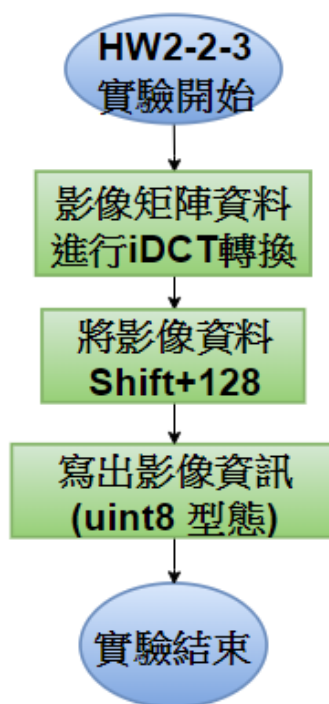


圖 3.8 HW2-2-3 處理流程圖

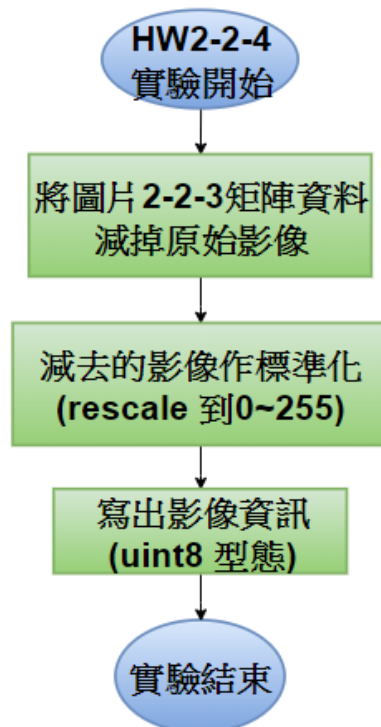


圖 3.9 HW2-2-4 處理流程圖

四、實驗結果

```
===== RUN MAIN_1 CODE =====  
===== Save image *.bmp =====  
2-1-2 MSE   : 66.572841646879468000000000000000  
2-1-2 PSNR  : 29.90 dB  
img_subtract 33.272997438499061, -36.320771034244075  
2-1-3 MSE   : 3732.85  
2-1-3 PSNR  : 12.41 dB  
  
===== RUN MAIN_2 CODE =====  
2-1-3 MSE   : 600.04  
2-1-3 PSNR  : 20.35 dB  
imgd_subtract 99.924382365105373, -123.127260988269170  
2-1-4 MSE   : 4453.45  
2-1-4 PSNR  : 11.64 dB  
===== Save image *.bmp =====  
===== END =====
```



圖 4.1 原始影像(灰階)

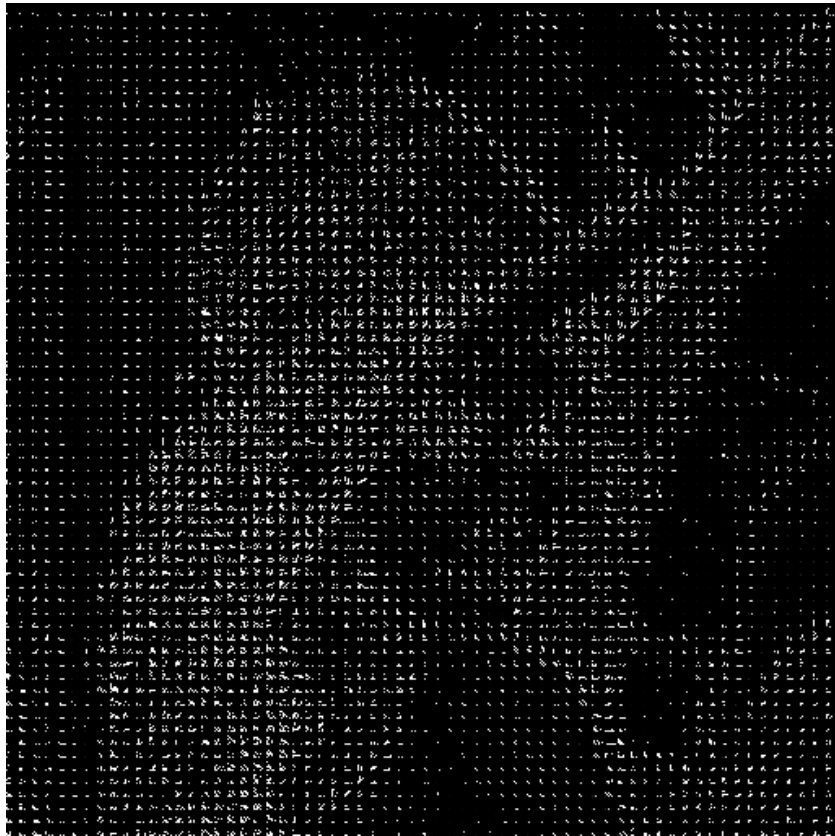


圖 4.2 HW2-1-1 原始影像經過 shift -128、DCT、Quantization 結果



圖 4.3 HW2-1-2 iQuantization、iDCT、shift +128 的結果

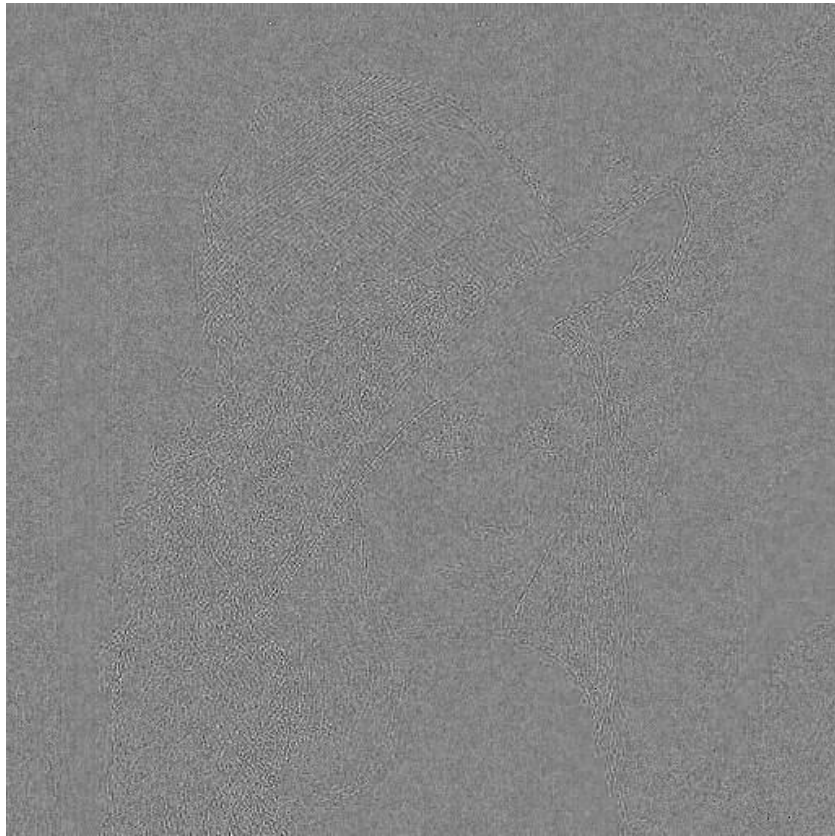


圖 4.4 HW2-1-3 圖 4.3 減圖 4.1 經過正規化的結果(0~255)



圖 4.5 HW2-2-1 原始影像圖 4.1 取奇數位置的影像

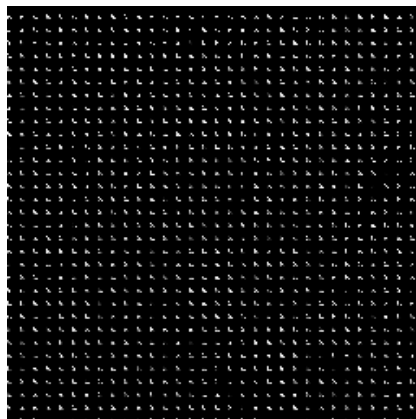


圖 4.6 圖 HW2-2-2 圖 4.5 經過 shift -128、DCT、Filter Mask 的結果



圖 4.7 圖 HW2-2-3 圖 4.6 iDCT 和 shift +128 的結果



圖 4.8 HW2-2-4 圖 4.7 減圖 4.5，再經過正規化(0~255) 的結果

4.2 備註：

1. 本次實驗量化的矩陣為：

```
QuantizationMatrix[8][8] = {
    {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, 57, 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}
};
```

2. 本次實驗 Filter Mask 的遮罩矩陣為：

```
FilterMask[8][8] = {  
    {1, 1, 1, 0, 0, 0, 0, 0},  
    {1, 1, 0, 0, 0, 0, 0, 0},  
    {1, 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},  
};
```

3. 開發環境：

- i. 電腦廠牌：Lenovo T470p
- ii. 作業系統：Windows10
- iii. CPU：Intel®Core™i7-7700HQ
- iv. RAM：16G
- v. Compiler：gcc (MinGW.org GCC-6.3.0-1) 6.3.0
- vi. IDE：文字編輯器 Atom
- vii. 執行：命令提示字元 (CMD)執行 *.exe
- viii. 程式碼與圖片：https://github.com/ZXPAY/DIP_Project

五、分析

本次實驗運用 DCT(Discrete Cosine Transform)，是一種類似 DFT(離散傅立葉)的轉換，主要差異在於 DCT 是使用實數，DCT 演算法常用於信號處理和圖像處理，由於 DCT 具有很強的能量集中特性，大多數的自然信號(包括聲音和圖像)的能量會集中在 DCT 後的低頻部分。

HW2-1-1(圖片 4.2)為是將原始圖片經過 DCT 轉到頻率域上，再進行量化所得到的結果，圖片上可以約略地看出原始影像的面貌(圖 4.1)，高頻的成分主要為灰階的變化，在圖片上黑色的地方，推測為圖片灰階變化不是那麼明顯之處，而白色、較亮處為灰階變化較明顯的地方。

HW2-1-2(圖片 4.3)為是將圖片 4.2 iDCT(Inverse Descrete Cosine Transform)，反轉換得到原始影像，此張影像和原始影像相差甚小，以肉眼難以觀察出主要變化，因此，圖 4.4 為圖 4.3 減去原始影像(圖 4.1)，再將其標準化，使其灰階落在 0 至 255，以此能讓肉眼便於觀察，圖 4.4 中較白的地方為灰階值較大的地方，也就是圖片相減後相差最大的地方，大概是 Lean 的輪廓、帽子等灰階變化較大的地方，推測為 iDCT 轉換前的 Quantization，因為取四捨五入(C 語言的 round 語法)，把小數點去除掉，導致後續反轉換後，會有誤差。而最後計算 PSNR(Peak Signal Noise Ratio)，為一個表示信號最大可能功率和影響他的表示精度的破壞性雜訊功率，本實驗計算後的 PSNR 為 **29.9 dB**，在維基百科

上面的影像壓縮、失真的參考 PSNR 為 30~50 dB，接近 30 dB，iDCT 後算是正常的失真。如果將 DCT 量化那部分的不取 round，讓整個 DCT、Quantization、iQuantization、iDCT，都以 double 的型態去作計算，可以得出圖 5.1 的圖，計算其 PSNR 可以達到 315.83 dB，圖 5.1 幾乎與原圖相同，幾乎無失真，不過在存 JPEG 或是壓縮後的影像，通常會以 8-bit(uint8)的型態儲存，在反轉換的過程中會導致一定程度的失真。

HW2-2-1(圖 4.5)，將原始影像(圖 4.1)，為取奇數水平和奇數垂直的 pixel，讓 512X512 降低為 256X256，此方式與取平均，每 2X2 pixels 取平均到 1 pixel，會讓圖片不會那麼模糊，本實驗後續有作取平均的作法，結果如圖 5.3，此圖像與圖 4.5 比較會發現此圖片較為模糊，不過依然能夠辨識的出大部分細節。

HW2-2-2 為圖片經過 DCT 和 Filter Mask 後的結果，而 Mask 的矩陣參照四的備註，8X8 的矩陣，只有左上角是 1，其餘矩陣都是 0，經過這個處理後，得到圖 4.6 的結果，讓每 8X8 的 pixel 保留低頻的部分，使得 iDCT 後得圖 4.7，高頻的部分被濾掉，及原本灰階變化明顯的地方便得不明顯。

HW2-2-4(圖 4.8)為圖 4.7 減去圖 4.5，再經過正規化(0~255)，由圖 4.8 可以很清楚的看到經過 Filter Mask，濾掉的部分，和原圖的差異性，可以很明顯地看到背景、Lean 的頭髮、帽子等等的輪廓，或者說是交界處、灰階變化明顯的地方，此方式類似於在 Spatial Domain 作高階濾波器，以取得圖片中的邊界(灰階變化處)。



圖 5.1 iDCT，計算全部以 double 型態(無取 round)



圖 5.2 圖 5.1 減去原始圖像再經過正規化(0~255)



圖 5.3 原始影像 512X512(圖 4.1)每 2X2 pixel 取平均降為 256X256

六、Reference

1. Gonzalez, Rafael C., and Richard E. Woods, "Digital image processing, " Prentice Hall, 2007.
2. 8 bits Lena.bmp download : <https://www.ece.rice.edu/~wakin/images/>
3. Discrete cosine transform Wikipedia :
https://en.wikipedia.org/wiki/Discrete_cosine_transform
4. JPEG Wikipedia : <https://en.wikipedia.org/wiki/JPEG>
5. PSNR Wikipedia :
https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio