

# Histogram Equalization

## Homework Report

## ● Technical description

### (a) Global Histogram Equalization

直方圖均衡化(Histogram Equalization)可以用來處理圖片局部過暗的方式，把原本集中在某區塊的機率函數(PDF)平均分布在所有顏色上面，並藉由像素個數的累積特性(CDF)將灰階直方圖進行拉伸([0,255])，在拉伸過程中，像素個數之間的關係不會改變，只是重新分配到新的灰階值上，以增強圖片的對比度。

$$p_r(r_k) = n_k / MN \quad k = 0, 1, \dots, L-1$$

$$s_k = T(r_k) = (L-1) \sum_{j=0}^k p_r(r_j) = \frac{L-1}{MN} \sum_{j=0}^k n_j \quad k = 0, 1, 2, \dots, L-1$$

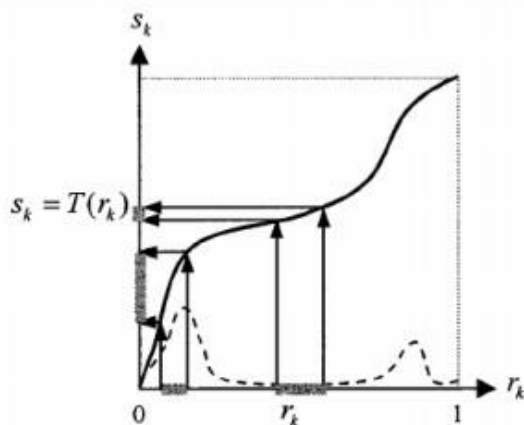
演算法：

Step 1. 計算影像灰階統計直方圖

Step 2. 從灰階統計直方圖計算累增直方圖

Step 3. 從累增直方圖計算等化分布直方圖

Step 4. 以此等化分布直方圖當作映射函數，重新指定影像每一 pixel 的灰階值。



## (b) Local Histogram Equalization

在網上查到有關於 local 的做法有很多，以下是在參考的 paper 看到的方式：每一個 pixel 與鄰近 pixel 的灰階值比較，決定其排序。再依此一排序的正比關係指定一個新的灰階值給這個 pixel。Local HE 影像增強方法是根據區域性(而非整張影像)的資訊來增強對比。

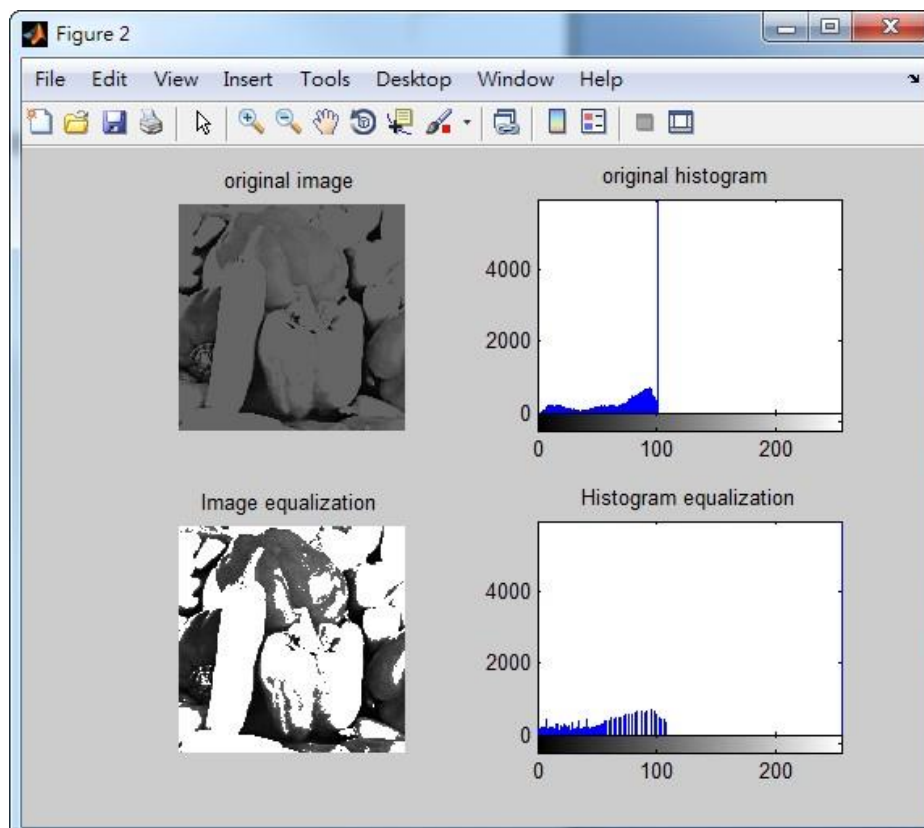
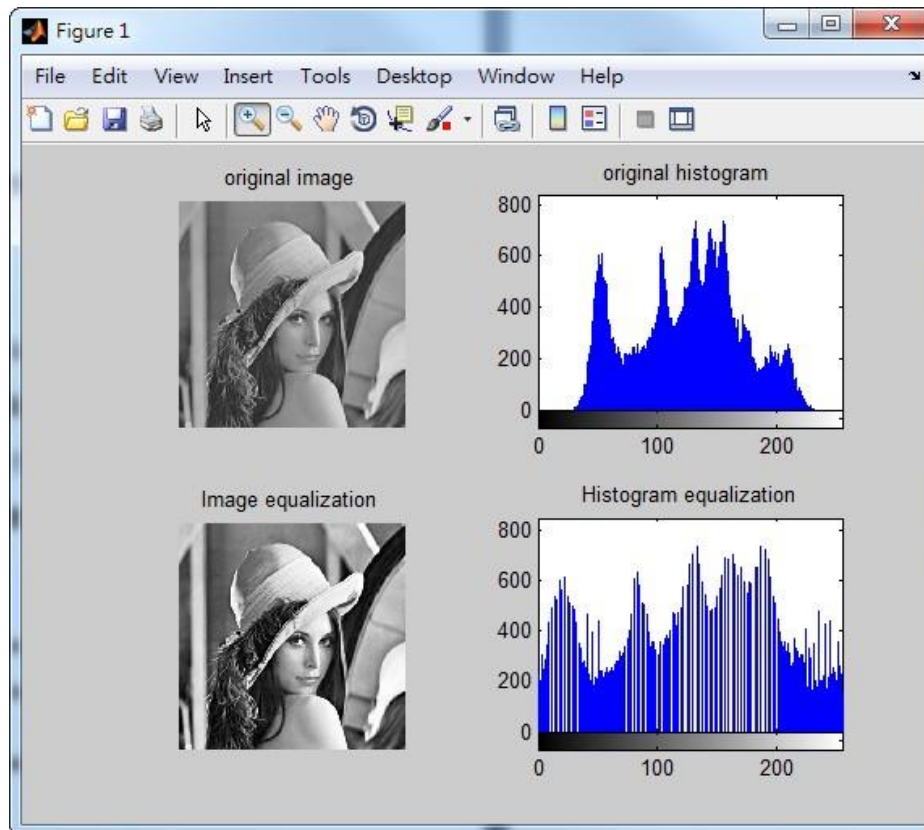
```
for each (x,y) in image do
{
    rank = 0
    for each (i,j) in contextual region of (x,y) do
    {
        if image[x,y] > image[i,j] then
            rank = rank + 1
        }
    output[x,y] = rank * max_intensity / (# of pixels in contextual region)
}
```

而此次作業按照題目規定，是將 256\*256 的圖，切割成 16 塊 64\*64 大小的圖片各自去做 Histogram Equalization，也就是對原圖 64\*64 的每一區塊做上述 Global Histogram Equalization 的方法，總共做 16 次。

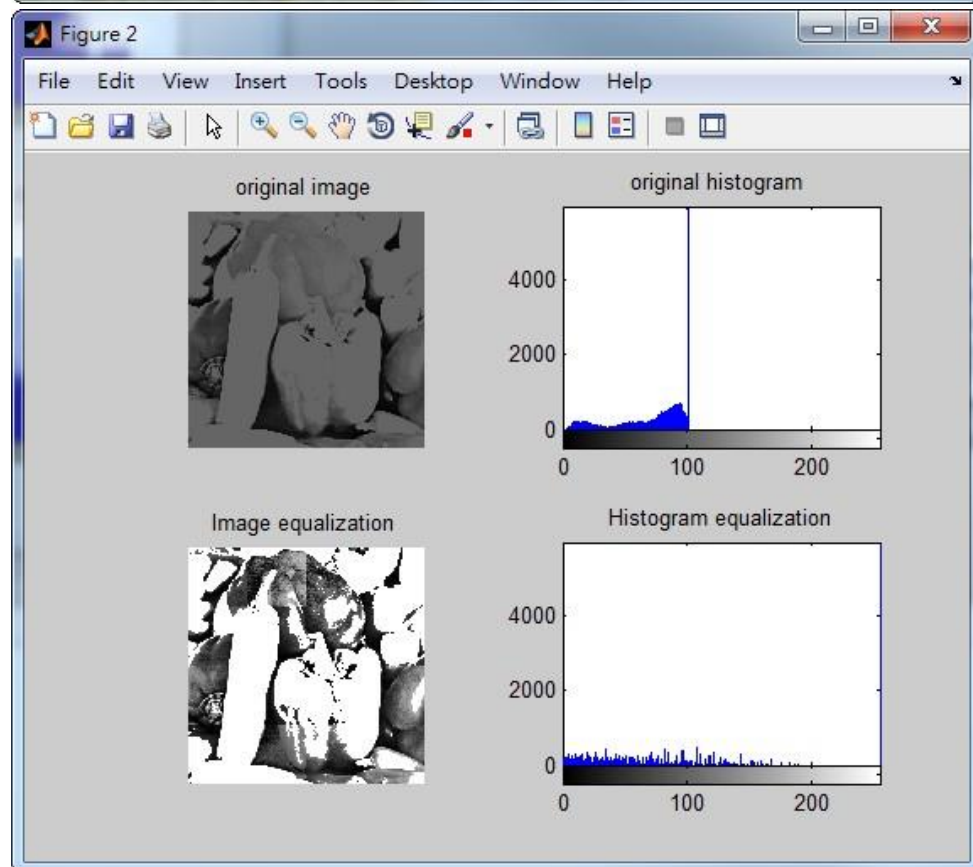
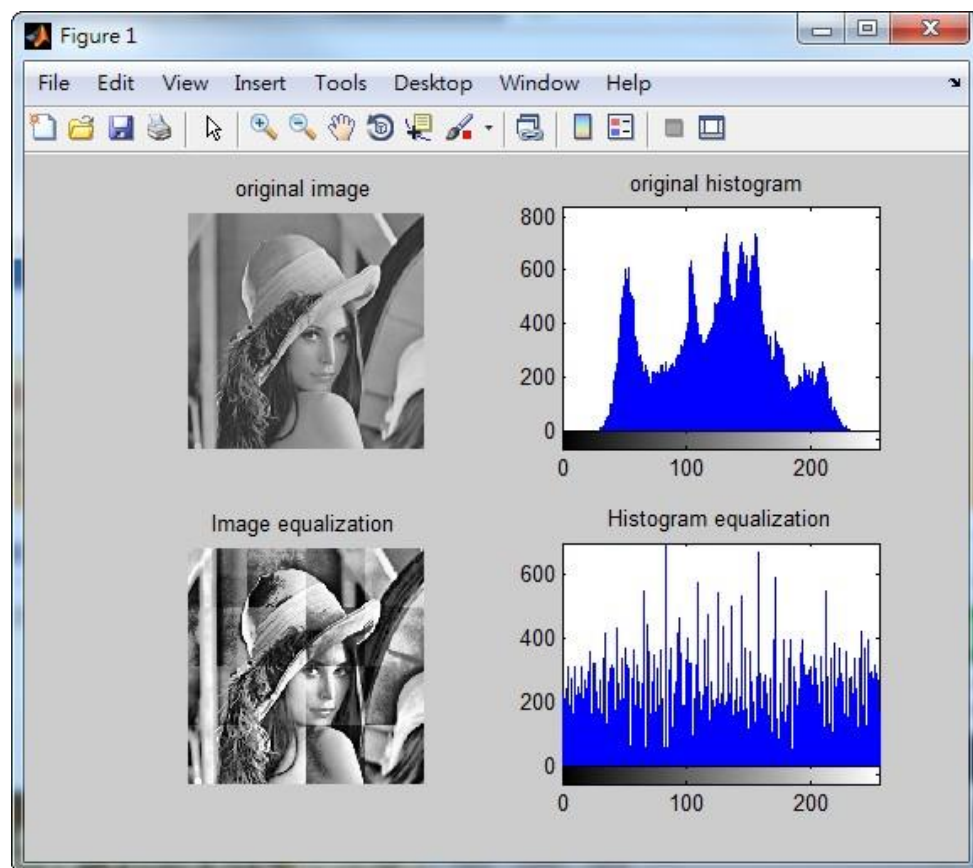
我的方法是利用 mat2cell 把原圖切成 16 塊，要存放輸出圖片的矩陣也是切成 16 塊，跑 4X4 次迴圈，把每一塊都做根 Global 一樣的事情，做完後存到輸出矩陣，最後把每一小塊的輸出矩陣合併，所以最後合併生成出來的圖，會有區塊的痕跡。

- Experimental results

(a) Global Histogram Equalization



## (b) Local Histogram Equalization



## ● Discussions

根據實驗結果可以看得到做完 Histogram Equalization 後，原本的圖片對比有明顯提高，不管是做 globalHE 還是 localHE，原圖的 histogram 都有明顯被拉伸，像素平均分布，而 localHE 拉伸的程度相對來看較好一些。

globalHE 和 localHE 沒有絕對的孰好孰壞，像是 Lena.bmp 做 global 比做 local 的效果好，而 Peppers.bmp 做 local 比做 global 的效果好。所以沒有一定要用哪種方式才會得到好的效果，而是要根據不同圖片的特性去套用不同的 Histogram Equalization 方式，求得最好的效果。

## ● References and Appendix

<http://ppt.cc/ncJO>

<http://ppt.cc/E32G>

<http://ppt.cc/BtPq>

<http://ppt.cc/aMn8>