

## HOMEWORK #1 Image Enhancement and Noise Removal

系級：資工三

學號：b04902099

姓名：黃嵩仁

### WARM-UP: SIMPLE MANIPULATIONS

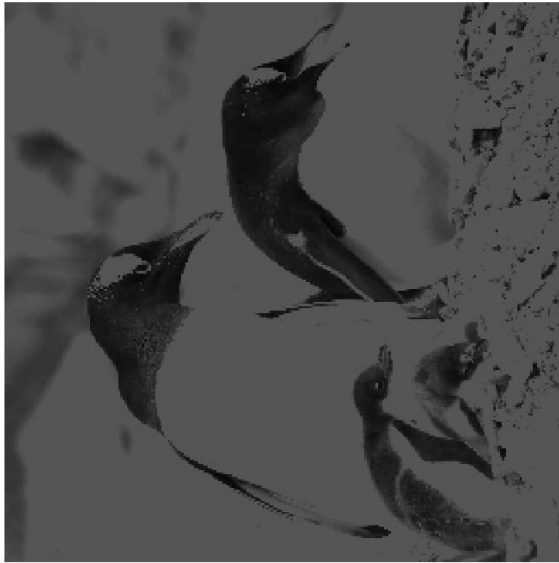
原圖(sample 1)：



Gray-level：



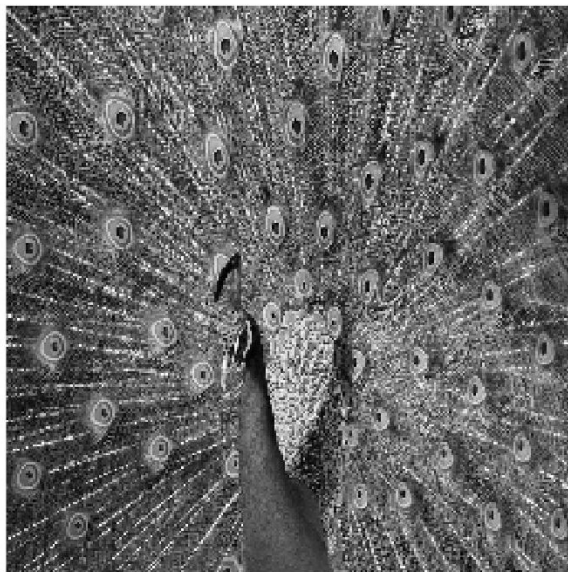
實作方式：將每個位置的  $r, g, b$  值相加後除以 3  
B (diagonal flipping)：



實作方式：將剛才灰階化的 I1，沿著左上-右下的角度翻轉(matlab： $G = G'$ )

### **PROBLEM 1: IMAGE ENHANCEMENT**

原圖(sample 2)：



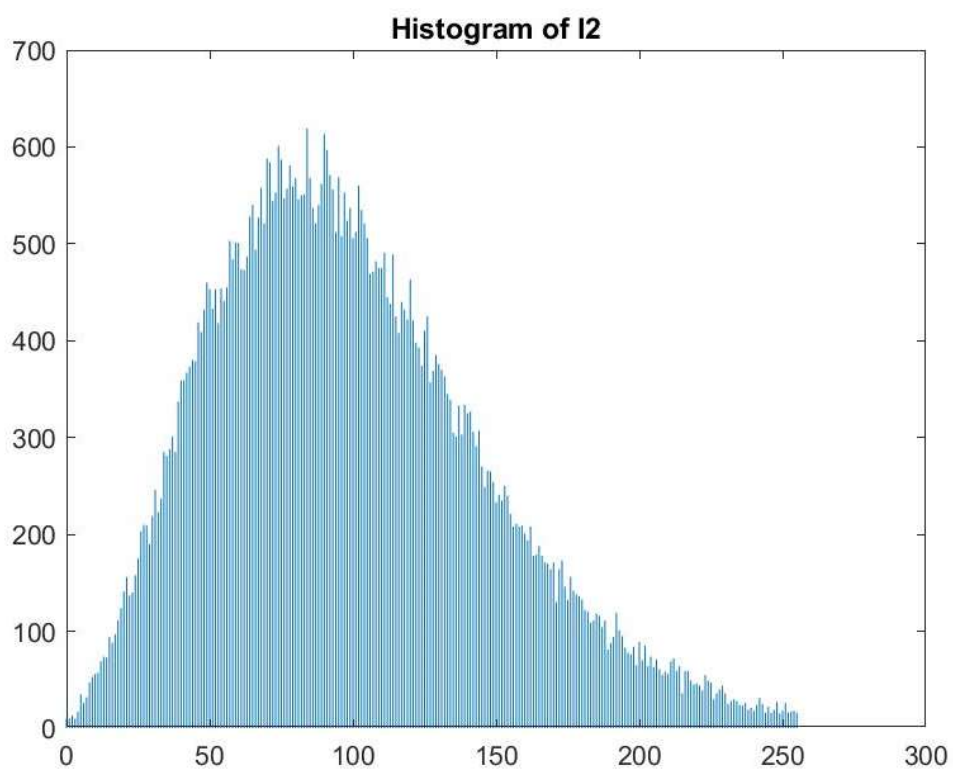
(a)



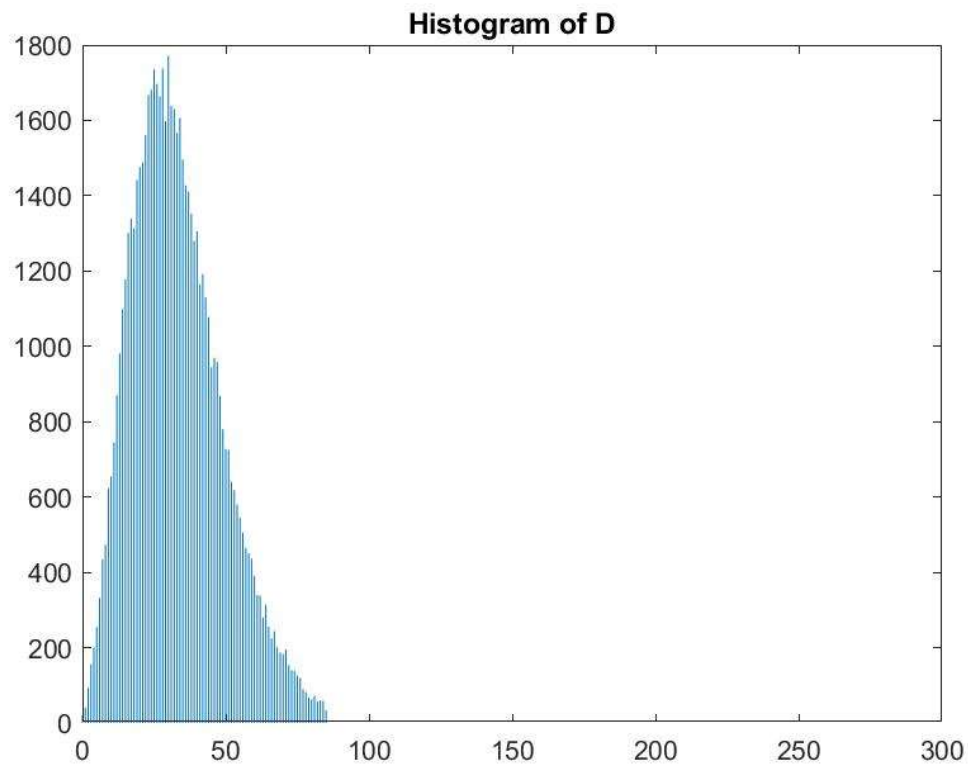
實作方法：將 image array 每個點的值都除以 3。

(b)

Histogram of I2 :

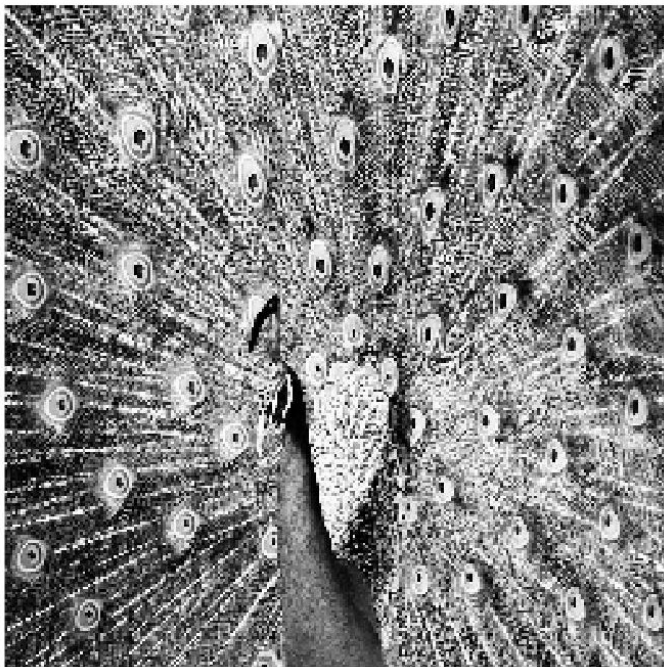


Histogram of D :



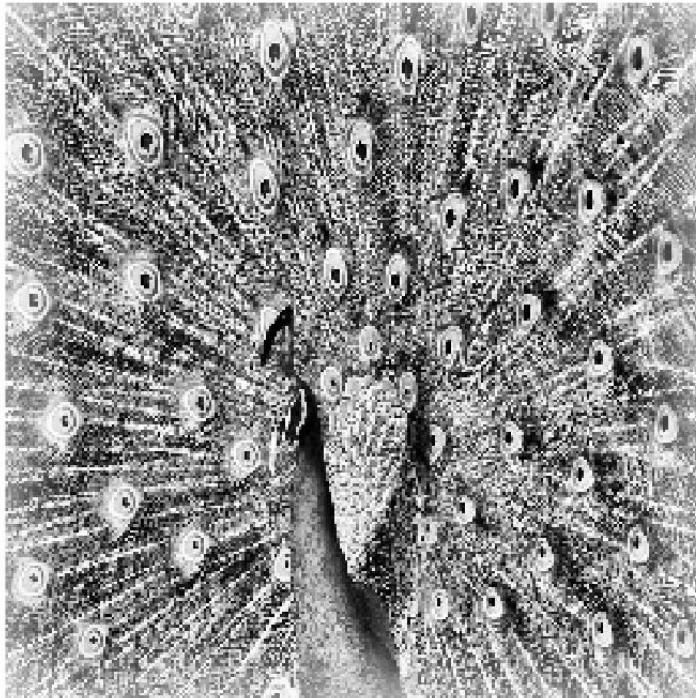
從 histogram 可以明顯看出整個分布都往左壓縮(變黑)，最高點的 intensity 也比原圖還要高上許多。

#### (c) Histogram equalization

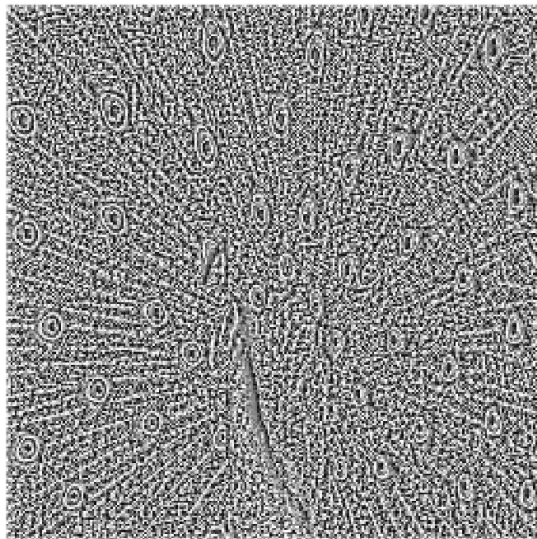


備註：雖然經過 histogram equalization 後較接近原圖的亮度，但也能看見與原圖相比損失很多細節。

#### (d) Local histogram equalization

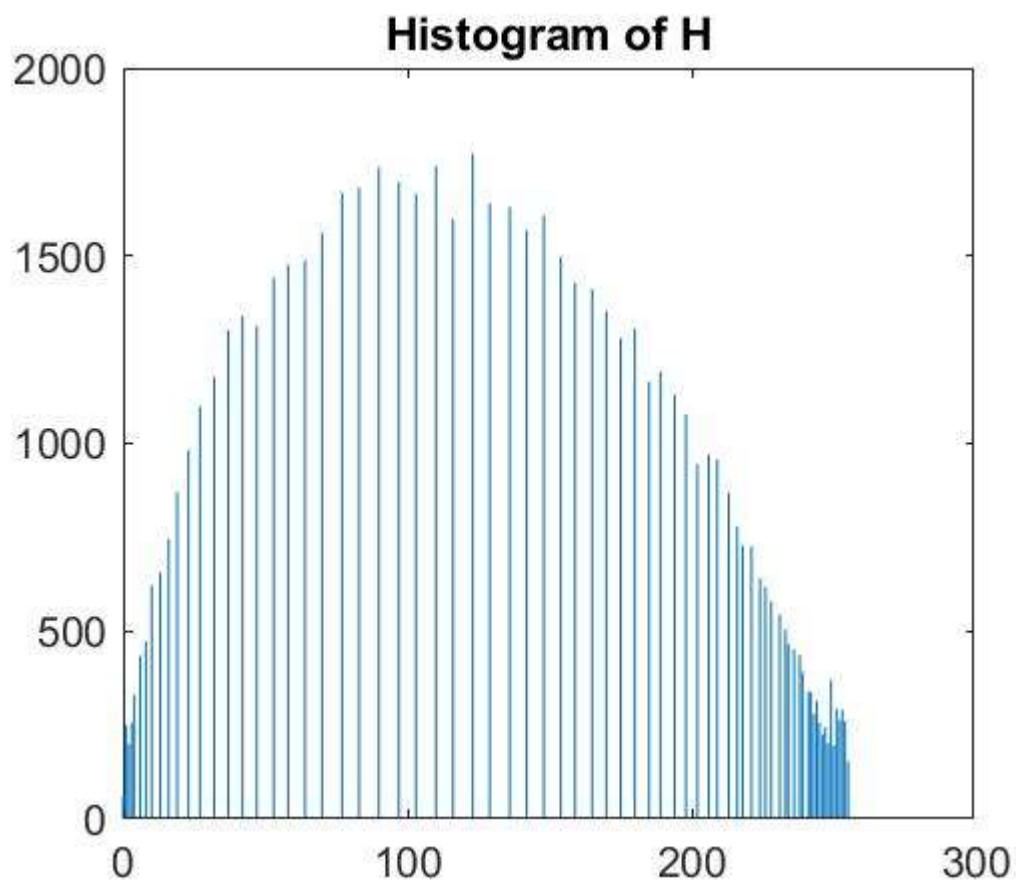


備註：因為我在 local histogram equalization 前，是在照片外圍 padding 0，而不是使用 odd/even copy，因此照片邊緣有偏白的傾向。此外，在嘗試過不同的 window size 後，發現當 window size 太小時還原度極差(如下圖，window = 5)，我最後選擇的 window size = 51。

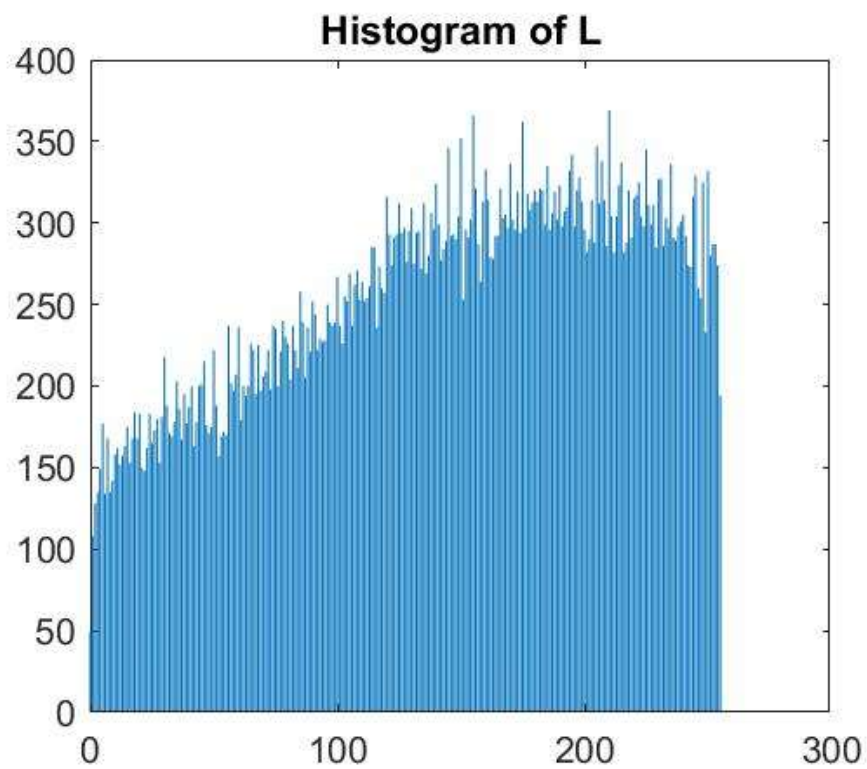


(e)

Histogram of H



Histogram of L



比較兩者後可明顯的發現，H 的 histogram，較稀疏且高，而 L 的 histogram 就



較為平均分散。代表 H 的深淺細節表現可能沒有 L 那麼好(使用較少的深淺度來表現)。

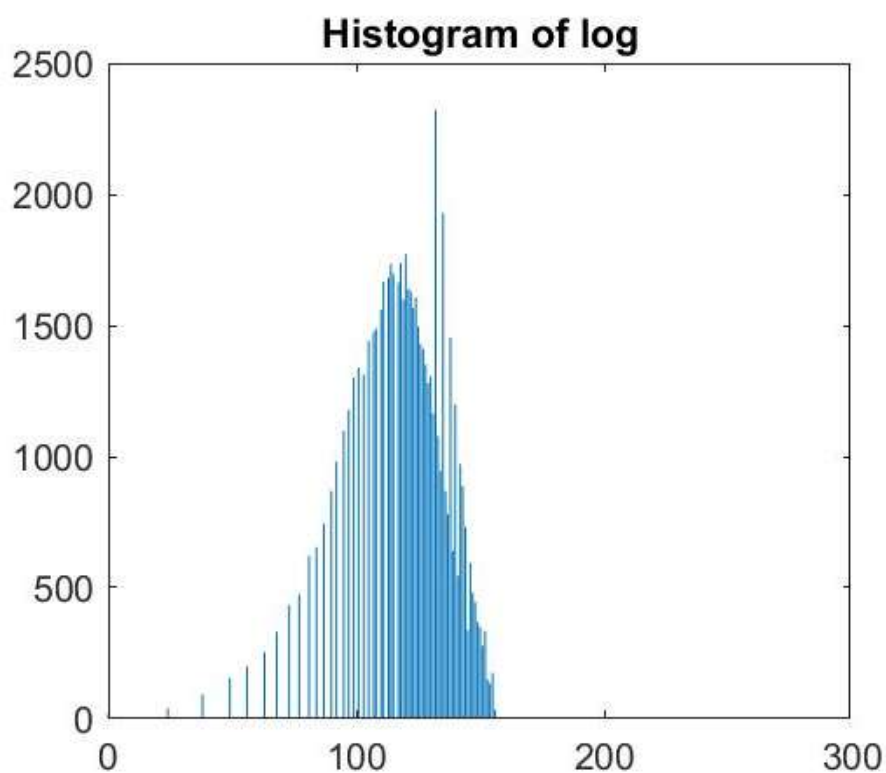
(f)

### (1)Log transform

Formula :  $y = c * \log(x + 1)$ ,  $x = 0 \sim 255$

我所選用的 constant c 值為 35。

註：c 值太小或太大都會影響深淺層次的表現。

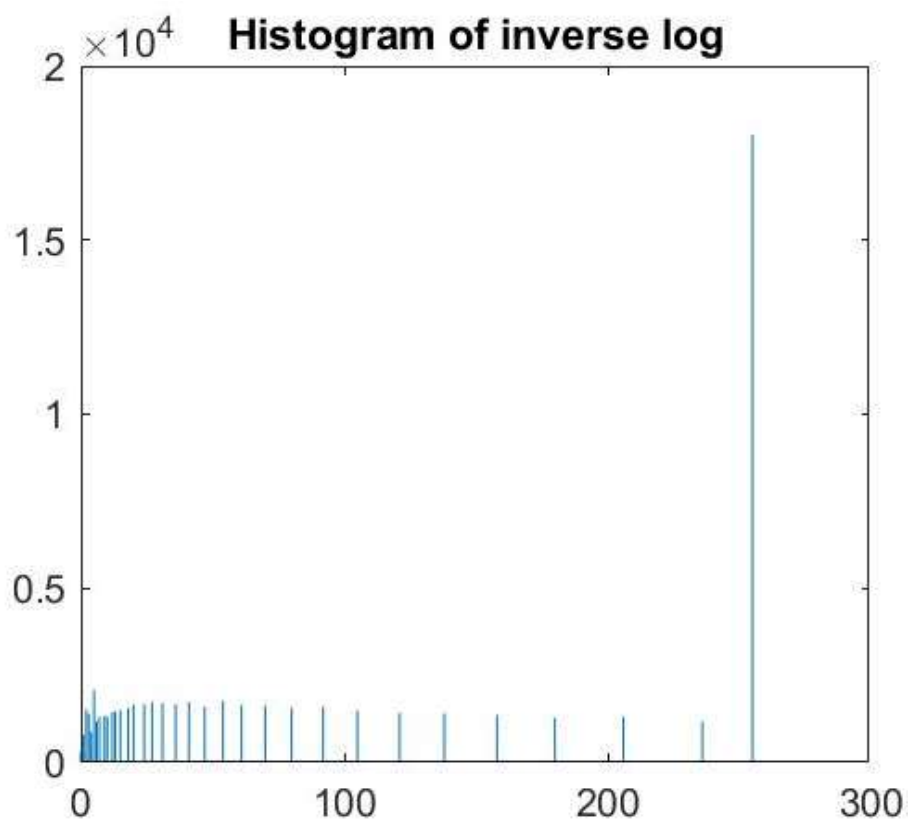
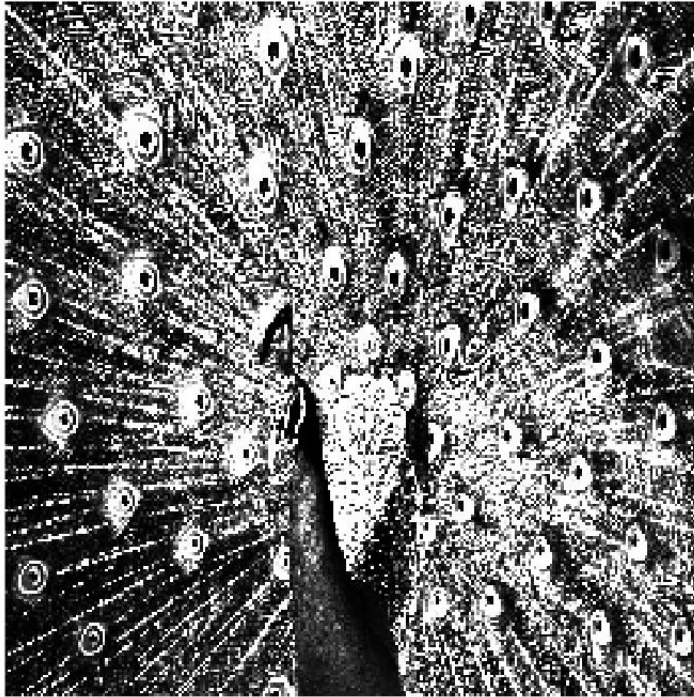


## (2) Inverse log transform

Formula :  $y = e^{\frac{x}{c}} - 1$

我選用的 constant  $c$  值為 7.5

註： $c$  值太小會有曝光的感覺， $c$  值太大則畫面整體過暗



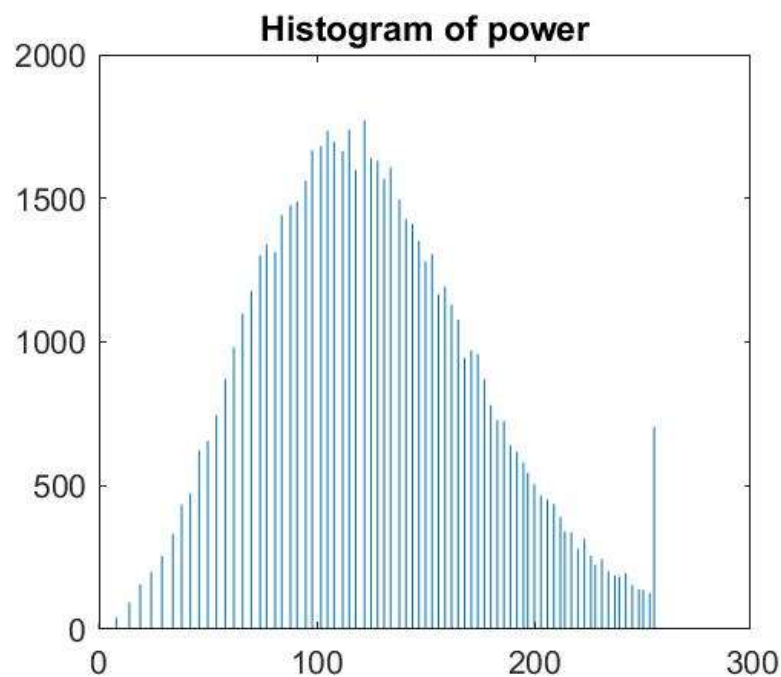
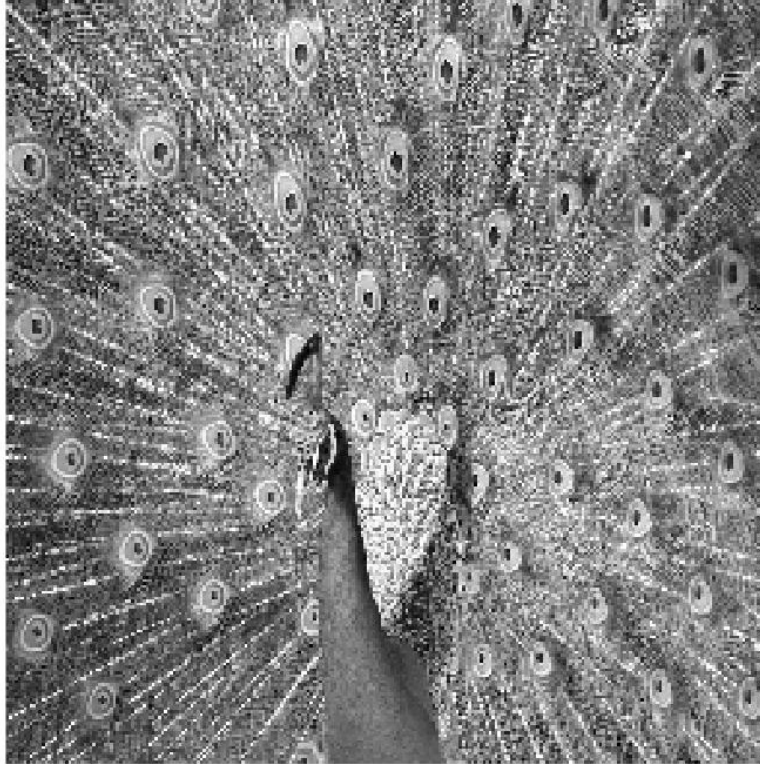


### (3) power-law transform

Formula :  $y = c * x^\gamma$

我選用的  $c$  值為 18， $\gamma$  值為 0.5

註：根據觀察  $c$  值影響亮暗細節，而  $\gamma$  值的影響比較像是曝光量，過小時，整張照片會偏糊且暗， $\gamma$  值過大時，則會有點過曝的感覺(在同一  $c$  值下)。



Conclusion：從三種方法的結果上來看 power-law transform 的效果最好，不只圖像看起來較接近，從 histogram 也可以看出其分布最接近原圖，而 log / inv-log transform 則有過亮 / 暗的問題，細節表現也相對較差。

## PROBLEM 2: NOISE REMOVAL

(I)

原圖



(a)

G1



G2



註：G1 所添加的 Gaussian noise 屬於 $(\mu, \sigma) = (0, 8)$

G2 所添加的 Gaussian noise 屬於 $(\mu, \sigma) = (0, 32)$

比較兩圖可發現，雖然兩個所屬分配的 $\mu$ 值相同，但 $\sigma$ 的差異卻讓兩者雜訊的量(顆粒)相差極大，當 sigma 值增加時，雜訊的量與強度會明顯增加。

(b)

S1



S2



S1、S2 兩圖的主要差異在 threshold 值的設定，threshold 值可看作 salt & pepper noise 的出現頻率，S1 的 threshold = 0.01，S2 的 threshold = 0.05，從圖片可看見當 threshold 變大時，noise 出現的機率也會增加(S2 出現 noise 的機率是 S1 的  $5 \times 2 = 10$  倍)，而兩者的 noise 相較 Gaussian noise 看起來都比較

沒有規律。

(c)  
RG



RS





RG 的部分：因為 Gaussian noise 是 uniform noise，因此我採用 loss-pass filter 來 de-noise。首先，我先採用 odd boundary extension，mask 我採用 3\*3

的大小  $\frac{1}{16} * \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ 。經嘗試後，雖然 5\*5 消除 Gaussian noise 的效果較 3\*3

好(顆粒感較低)，但照片整體較模糊，船尾上的字也更不清晰 (如下圖)，因此最後選擇了 3\*3 的 mask。



RS 的部分：Salt & pepper noise 屬於 impulse noise，我採取 median filtering 來 de-noise。我先採用 odd boundary extension，在使用 3\*3 的 mask。經觀察 3\*3 的 mask 較 5\*5 的好，若採用 5\*5 的 mask，則”筆刷感”較重，較有油畫

的感覺(如下圖)。



(d)

PSNR values :

RG : 29.5644

RS : 29.1348

(II)

原圖：



去除皺紋後



實作方法：

首先，先將原圖做 **odd boundary extension**，接著利用類似 **median filtering** 的概念(**window size = 9\*9**，我將取 **median** 改成取 **max**，發現這樣效果比較好)，去消除大部分皺紋與細紋，再利用 **low pass filtering (window size = 5\*5)**，降低顆粒感。

從結果可看見皺紋僅剩下下方那條魚尾紋，但整張照片卻也以此變得相當模糊，對比度也比原圖差。