

學號:B06902136

系級:資工四

姓名:賴冠毓

Problem 1: WARM-UP

(a)

My motivation and approach:

運用彩色轉灰階公式: $R \times 0.299 + G \times 0.587 + B \times 0.114$

Original images:



Output images:



Discussion of results:

看起來確實是灰階的，跟直接以灰階讀進來再輸出的結果看起來一樣，感覺成效很好。

(b)

My motivation and approach:

把圖片左邊數來第 i 行跟右邊數來第 i 行做交換，就像是字串反轉一樣。

Original images:



Output images:



Discussion of results:

看起來確實是水平翻轉，跟直接用 flip 函式的結果長看起來一樣，感覺成效很好。

Problem 2: IMAGE ENHANCEMENT

(a)

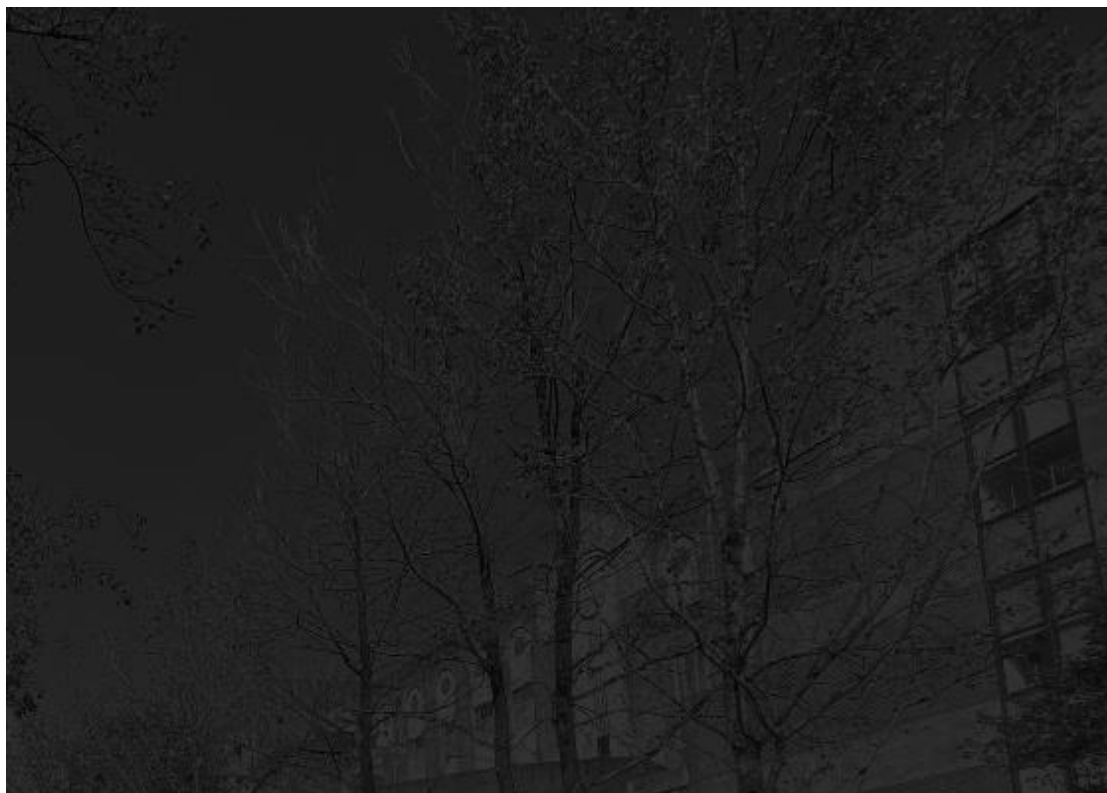
My motivation and approach:

就直接把每個 pixel 的值除以 5。

Original images:



Output images:



Discussion of results:

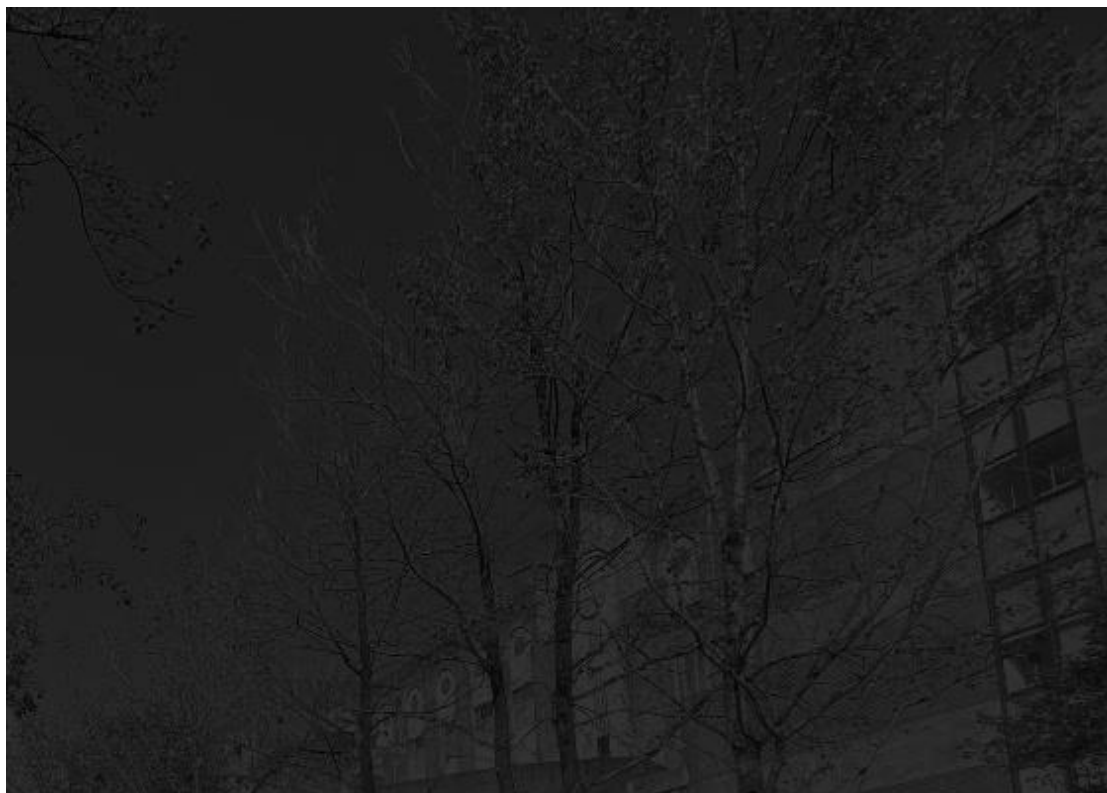
真的變很暗。

(b)

My motivation and approach:

就直接把每個 pixel 的值乘以 5。

Original images:



Output images:



Discussion of results:

看起來有恢復成原本 sample2. jpg 的亮度，感覺成效很好。

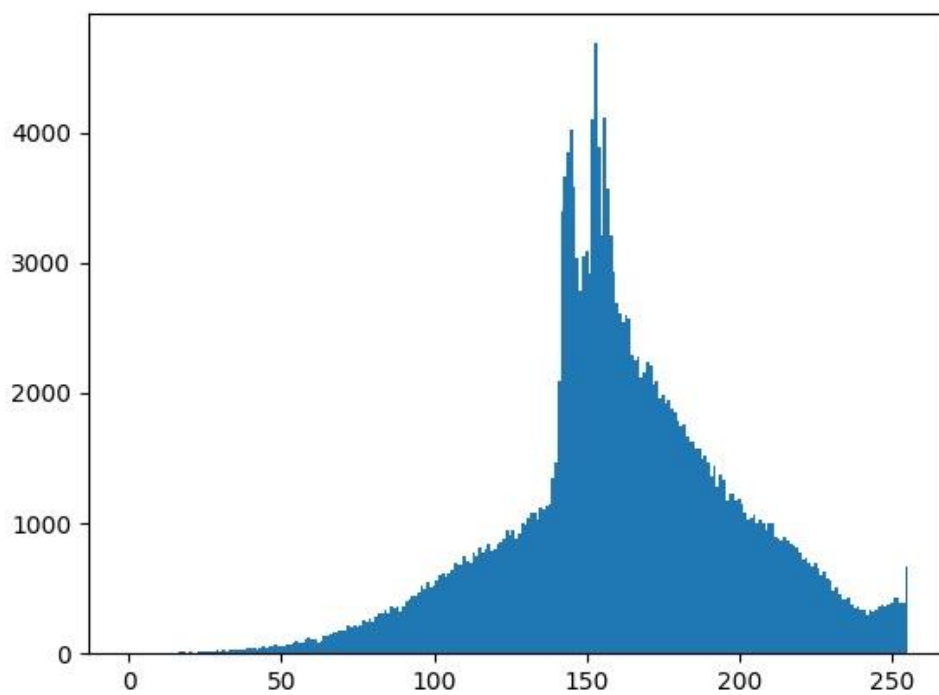
(c)

My motivation and approach:

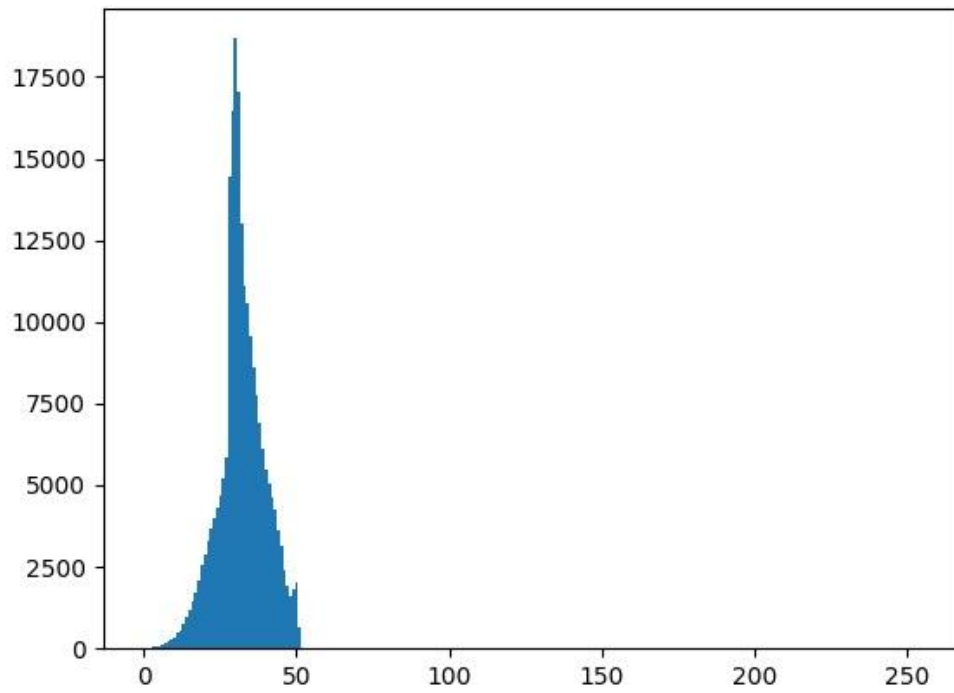
開一個 $[0, 255]$ 的 numpy，去統計每個 intensity value 的 pixel 數。

Output images:

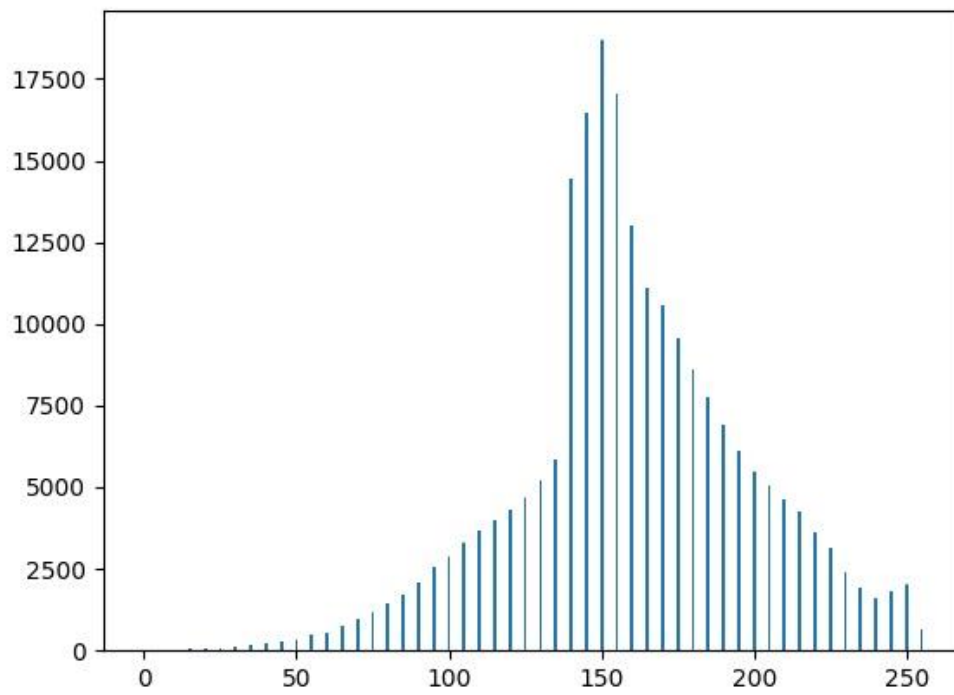
sample2_hist.jpg:



3_result_hist.jpg:



4_result_hist. jpg:



Discussion of results:

先變暗再變亮後的圖片跟原本的看起來一樣，但 histogram 會變得比較分散。
可能是因為在變暗過程中有些 pixel 被除完後變成 0 而損失(過度往左擠壓)，
導致變亮過程中找不回來了，不過整體圖形看起來形狀還是類似。

(d)

My motivation and approach:

先做 histogram 的統計，再來計算 Cumulative Distribution

Function(CDF)，並且將 CDF 分攤掉，給整張圖做 uniformly distributed。

Original images:



Output images:



Discussion of results:

整個圖片的亮度確實有分散掉，原本比較亮的招牌部分反而沒那麼顯眼了。

(e)

My motivation and approach:

先定義一個 boundary extension 函式，並採用 even 模式(擴展出去的點與原本相鄰的 pixel 相同)。再來找到適當的 window size(此處採用 51)，擴展需要的邊界，然後以每個點為中心在 window 進行 equalization。

Original images:



Output images:



Discussion of results:

此處 window size 採用 51，是經過多次測試所採用的數字，若比它小則 pixel 間的微小差異會被放大太多，導致圖片過於模糊效果不夠好；若比它大則招牌

的地方會看不清楚，而其他地方無明顯進步反而徒增時間。另外須注意執行時間可能有點久，要等它一下。

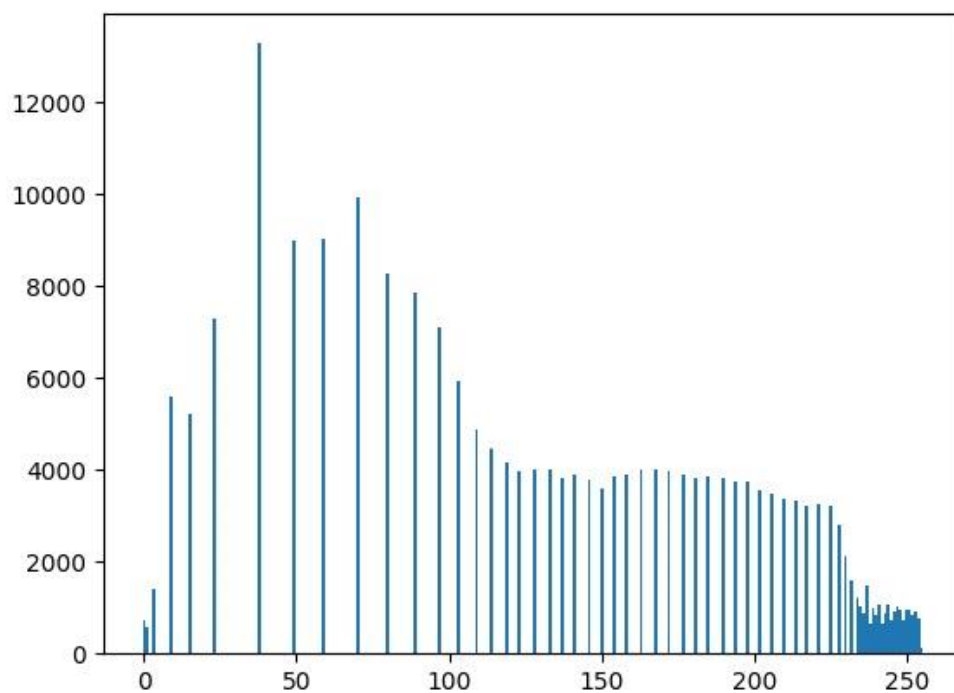
(f)

My motivation and approach:

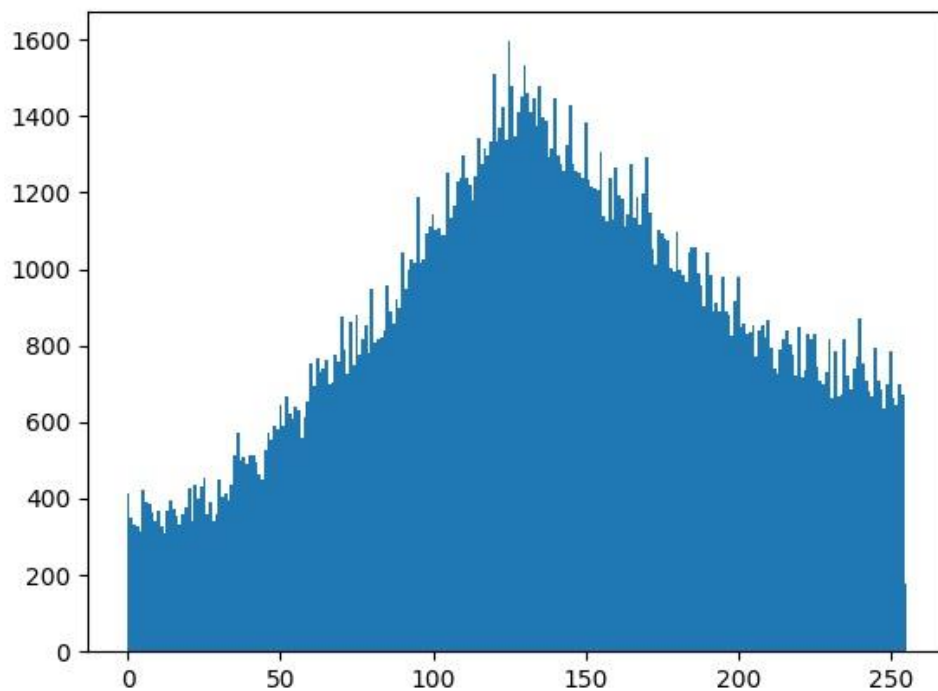
與前面統計 histogram 的方法相同。開一個 $[0, 255]$ 的 numpy，去統計每個 intensity value 的 pixel 數。

Output images:

5_result_hist. jpg:



6_result_hist. jpg:



Discussion of results:

由於 global histogram equalization 是整張圖做分攤，所以 histogram 會比較分散；而 local histogram equalization 則是與附近 window size 內的點進行比較、分攤，window size 越大則 histogram 越密集，各個點之間的差異也會越顯著。

(g)

My motivation and approach:

使用 power-law transform 且次方小於 1 (這裡採用 $\frac{1}{2}$)，讓整張圖的亮度看起來平均一些，彼此差異變小。

Original images:

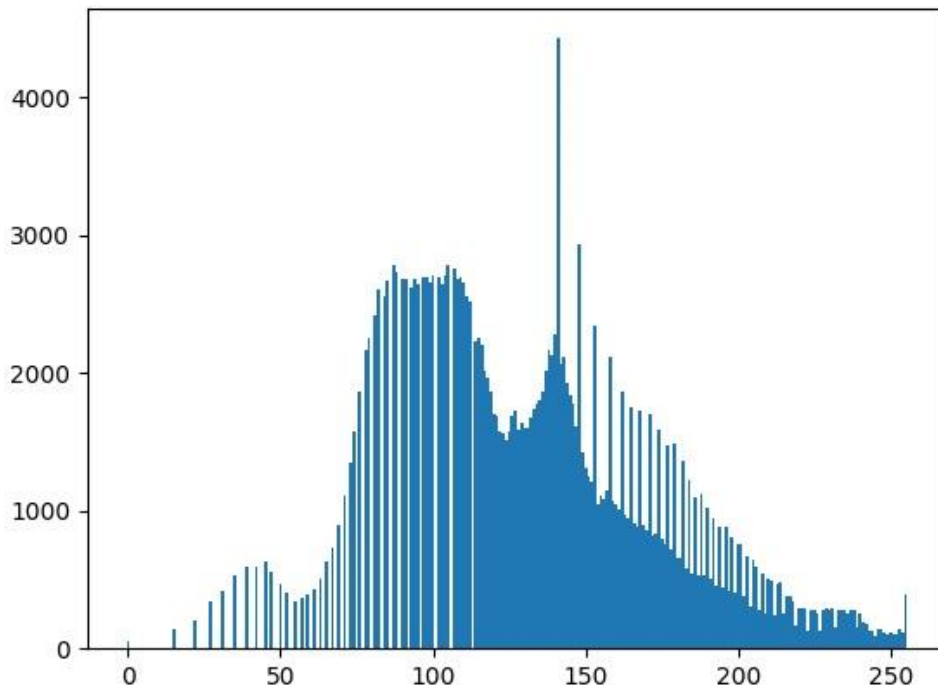


Outputs:

image:



histogram:



Discussion of results:

在 guideline 中提到的 3 種 transform，經過測試後發現 power-law transform 的效果比起 log transform、inverse-log transform 都還要好，因為它可以在把圖變亮的同時保留亮暗對比差異的一些小細節，這樣才能分出太陽跟船。而

power 採用 $\frac{1}{2}$ ，是嘗試多個數字後發現效果最好的，如果採用更小的 power(ex:

$\frac{1}{3}$)，整體會顯得太亮，船跟太陽的對比反而沒有那麼明顯，會分不出來。

Problem 3: NOISE REMOVAL

(a)

My motivation and approach:

sample6. jpg 的 Gaussian noise 是 uniform noise，所以使用 low-pass filter 即可；sample7. jpg 的 salt-and-pepper noise 則是 impulse noise，所以採用 median filter。

low-pass filter: $H = \frac{1}{(b+2)^2} \begin{bmatrix} 1 & b & 1 \\ b & b^2 & b \\ 1 & b & 1 \end{bmatrix}$, $b = 2$ 。然後將 sample6. jpg 做

boundary extension(mask size=3)，最後進行 filtering，每個 pixel 乘以 H 後的值總和再除以 H 所有 element 的總和，即為最後通過 filter 的 pixel 值。

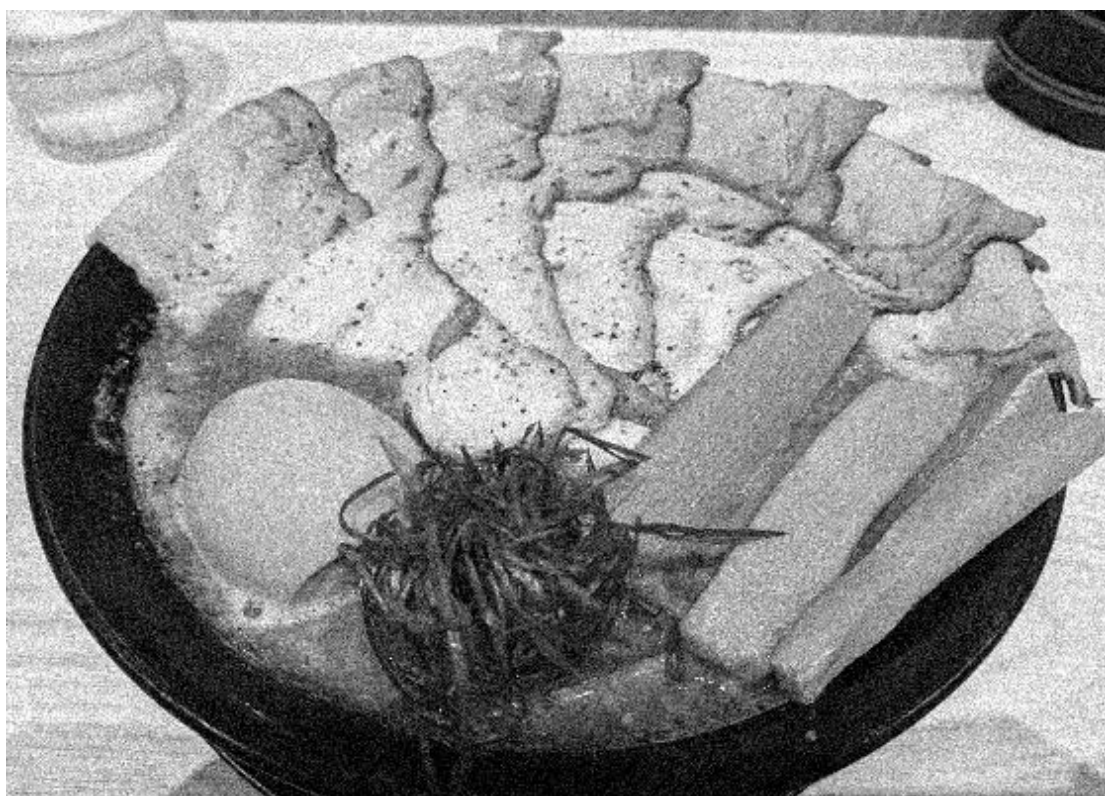
median filter: 方法同上，只是 low-pass filter 為抓取平均值，median filter 則是抓 mask 裡的中位數為最後通過 filter 的 pixel 值。不過這邊由於 sample7.jpg 的雜訊實在太多，需要 filtering 兩次出來的圖才會比較好。

Original images:

sample5.jpg:



sample6.jpg:



sample7. jpg:



Output images:

8_result. jpg:



9_result. jpg:



Discussion of results:

前面已經提過根據 noise 種類不同，所以選擇使用不同的 filter。至於參數的設定則是綜合 output 出來的圖和底下 PSNR 值，經過多次測試發現

mask size = 3、b = 2、median filtering 兩次為最好的結果。以結果的圖來說，我們可以觀察比較光滑的蛋表面，蛋表面沒有雜訊的話其實就表示圖片蠻乾淨的了。

(b)

My motivation and approach:

使用公式：

$$MSE = \frac{1}{w * d} \sum_j \sum_k [F(j, k) - F'(j, k)]^2$$

$$PSNR = 10 \times \log_{10} \left(\frac{255^2}{MSE} \right)$$

Outputs:

PSNR values of 8_result.jpg = 27.378904

PSNR values of 9_result.jpg = 23.509027

Discussion of results:

8_result.jpg 使用 mask size = 3、b = 2，的確產出最高的 PSNR 值；但是 9_result.jpg 使用 median filtering 兩次其實產出的並不是最高的 PSNR 值，只用一次反而 PSNR 值更高，應該是因為使用多次 filtering 會導致圖片變模糊，但是只使用一次 filtering 出來的圖仍有些雜訊沒有過濾乾淨，所以只好犧牲一點 PSNR 值，換取更乾淨的圖片。