

Digital Image Processing (2023) HW2

I. Low-luminosity Enhancement

• Power-Law Transformations

$$O(x, y) = I_{\max} \left(\frac{I(x, y)}{I_{\max}} \right)^\gamma$$

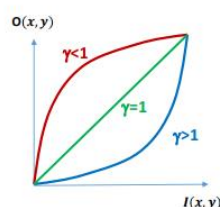
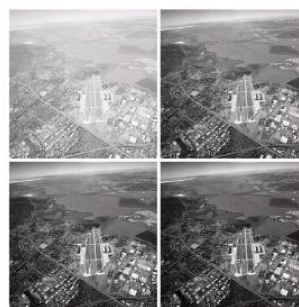


FIGURE 3.9
(a) Aerial image.
(b)–(d) Results of
applying the
transformation in
Eq. (3.25) with
 $\gamma = 1$ and
 $\gamma = 1.4$ and
 5.0 , respectively.
(Original image
for this example
courtesy of
NASA.)




Ref: Gonzalez and Woods, "Digital Image Processing"

圖一、Gamma Ratio 概念呈現

(截圖自課堂講義 Spatial-Domain Operator(2023)-p8)

此題我採用調整 gamma ratio 的方式呈現，如圖一所示，該演算法可以設定使 $0 < \text{gamma} < 1$ ，透過指數調整影像像素點的訊號，而原先數值較低的輸入就會被以非線性放大。

實作上，由於影像是透過 `u_int8(0~255)` 資訊組成，因此可以先做出一個 `0~255` 為 input 的 gamma mapping table，它會依據我設定的 gamma 值 ($\text{gamma} = \frac{1}{\text{program input avg}}$)，mapping table 輸出則依據 $\text{output} = \left(\frac{\text{input}}{255} \right)^\text{gamma} \cdot 255$ 定義。接著，我將 BMP 相片讀取到的 RGB 三通道數值分別通過這個 gamma mapping，找到對應新值後，輸出新圖片。

		
原圖(等同 gamma=1)	Low (gamma = 1/2)	High (gamma = 1/5)

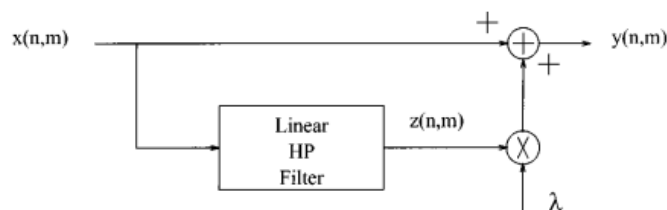
表一、gamma value adjustment 比較圖

上表中，左至右依序呈現將 gamma 值調低的效果，可以發現原本亮度較低的區塊，隨著 gamma 下降，也越趨明亮、可以分辨實際物體。這是因為 BMP RGB 三通道數值越低越接近黑色，而 gamma 設定在小於 0 時，則會將這類靠近 0 的 input 放大更多。也因此相片的亮部並沒有因為暗部條亮，就變得整片都亮到看不見。

其實有關於 Low-luminosity Enhancement 還有其他做法，像是將 RGB 三通道資訊轉換至 HSL、HSV 等其他顏色系統上，透過調整前者的 L channel (Lightness 亮度)，或是後者的 V channel (Value 明度)，向它施加線性轉換或是非

線性轉換(也可以用 Gamma 調整)，甚至對其做 Histogram Equalization，隨後再轉回 RGB 系統上，即可以得到色彩更為均衡的新相片。

II. Sharpness Enhancement



圖二、Sharpen filter 運作原理

(截圖自 <https://blog.csdn.net/a8039974/article/details/120783419>)

當圖片需要銳化時，需要通過一個 high pass filter，接著再與原 input 疊加。而在空間域時，就是透過在圖片上施加 convolution。High pass filter 作用類似一個微分器，目的是使的圖片微小訊號的變異程度放大，而在圖像內的物體邊緣，就是變異程度大的地方，因此這邊的微分訊號會特別明顯。當得到明顯的邊緣後，就可以疊加在原圖上，使得整體相片在細節處可以有明顯的紋理。

實作上，我採用 Laplacian kernel 與 Identity kernel 疊加出來的 sharpen kernel，如下式所示：

$$\begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

由於 convolution 運算是線性運算的一種，因此圖二呈現的流程圖，僅需要直接 apply sharpen kernel 即可。

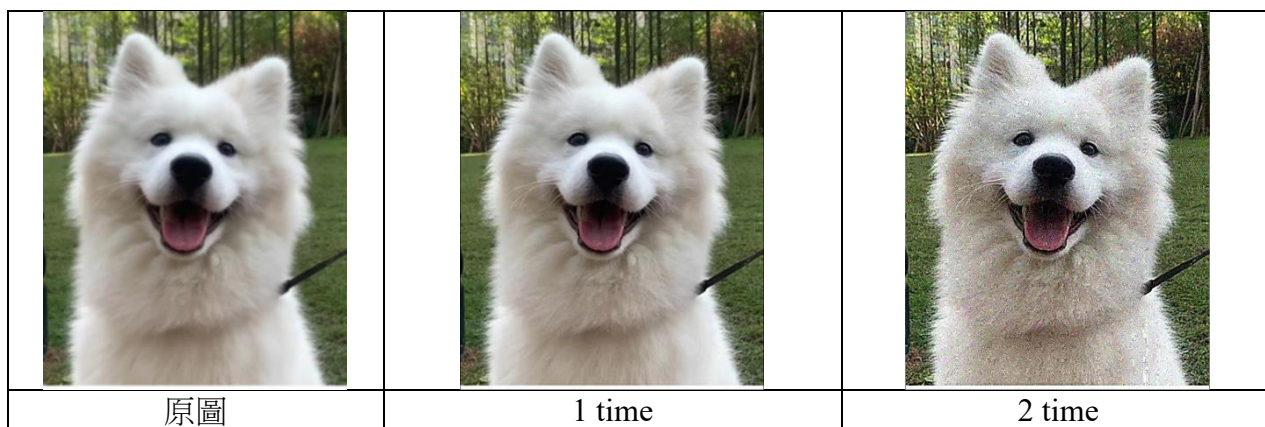


表 2、Sharpen filter 效果比較

上表左至右，分別是原圖、銳化、加強銳化的展示，可以看到狗的毛髮變得清晰分明，甚至出現紋理。需要加深銳化時，可以透過重複施加 sharpen kernel，重複 convolution，即可放大銳化效果。

同樣的也可以在 Laplacian kernel 動手腳，放大每個 element 的數值，像是下式：

$$\text{New Laplacian kernel} = \begin{pmatrix} -2 & -2 & -2 \\ -2 & 16 & -2 \\ -2 & -2 & -2 \end{pmatrix}$$

一樣加上 Identity Kernel 後，會發現銳化效果也同樣放大了。

III. Denoise

範例圖的雜訊很明顯是 salt and pepper noise，因此我透過 Mean filter 來進行過濾，這 filter 被歸類在 Smooth filter，同時也算是一種積分器運算。其演算法是施加一個 3x3 遮罩，掃過相片每個位置，而掃過的區塊會計算平均，該平均值就會是該次遮罩中心點的新像素點資訊。以此方式可以將 local 雜訊去除掉，類似於做一個 FIR filter。但缺點是 mean filter 也會將細節弱化，照片本身的資訊也會跟著遺失、破壞掉。

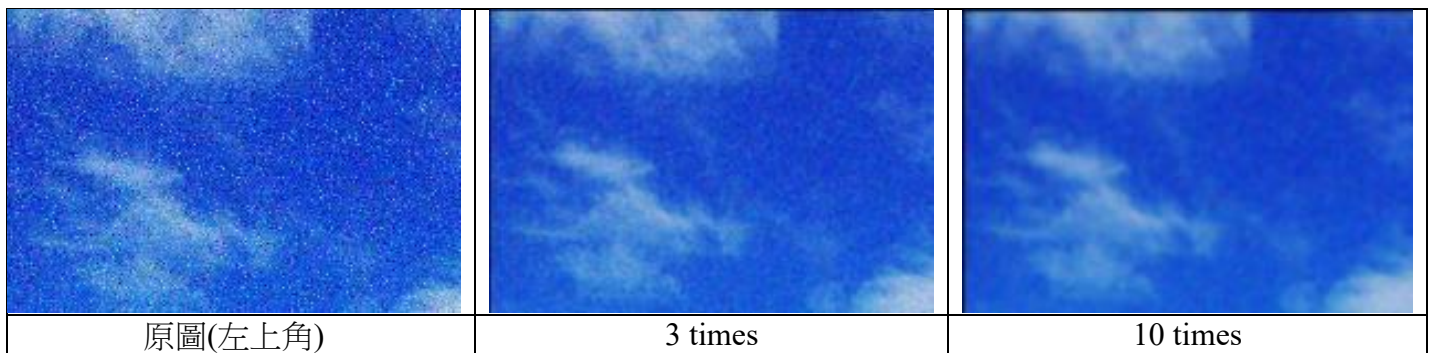


表 3、Denoise by Mean Filter 比較圖

在 Mean filter 上，若要加強 Denoise 效果，可透過重複施加 Mean filter。在上表可見，原先左圖的明顯顆粒雜訊；經過施加 3 次時，顆粒點已經模糊掉了；施加 10 次時，則完全沒有顆粒感。但同時也可以注意到施加次數越多，物圖片中的有效資訊(雲朵物體)的細節也被弱化，也跟著變模糊了。

若想加強 Denoise 效果同時降低資訊損失，則可以改施加 adaptive median filter，其有更細微的參數調整，可防止資訊損失過多。

IV. Reference

1. 課堂講義
2. 銳化濾鏡 <https://jason-chen-1992.weebly.com/home/-unsharp-masking>
3. 圖像銳化 <https://zhuanlan.zhihu.com/p/511643260>
4. sharpening spatial filters https://blog.csdn.net/weixin_44597810/article/details/105140380
5. Image sharpening <https://blog.csdn.net/a8039974/article/details/120783419>