

影像處理與電腦視覺 作業一

統計所 李權恩 R26094022

本次作業主要以 Python 完成，最後再以 tkinter 套件將程式包裝成一個使用者介面以及 pyinstaller 包裝成.exe 檔。

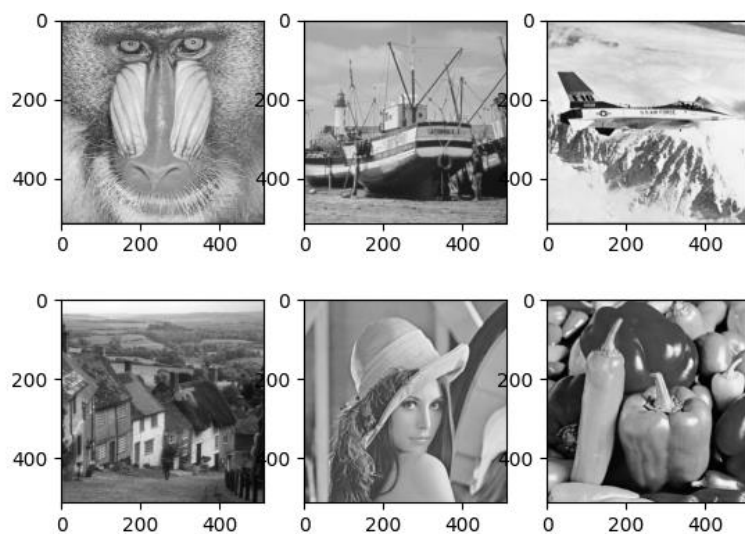
相關程式碼與包裝好的.exe 檔皆附在此：

<https://github.com/Quan-En/DIPCV/tree/main/assignment1>

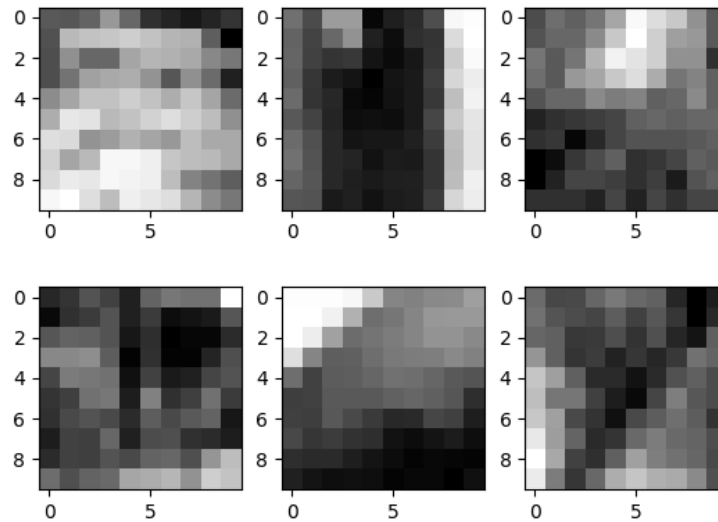
Problem

(a) 檔案讀取：

本次作業共 6 張檔案，其中.bmp 檔與.raw 檔各有 3 張，在.bmp 檔的部分使用 cv2.imread 來讀取圖片，獲得 $[512, 512, 3]$ 的陣列 (array)，3 個維度對應為圖片的高、寬與三原色 (channel)，由於 3 張檔案皆為灰階 (grayscale) 圖片，因此只需取三原色層中的任一層，而在.raw 檔的部分使用 np.fromfile 讀入一個長度為 262144 的向量，透過 reshape 的方式轉換成圖片，最後皆可使用 plt.imshow 來將圖片視覺化，原圖如下：



取出圖片中央 10×10 的子區塊，圖片如下：



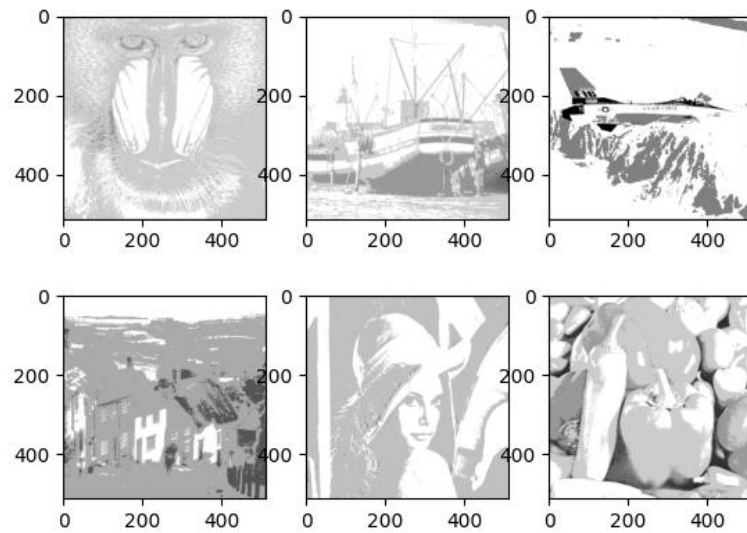
(b) 影像強化(image enhancement)：

本次作業實作了 3 種不同的方法分別為對數轉換 (log-transform)、伽瑪轉換 (gamma-transform)及負影像 (image negative)。在下方各項方法公式說明皆以 x, y 分別代表輸入與輸出值，其中 x 代表原影像的亮度值 (Intensity)， y 為轉換後的亮度值， $L - 1$ 代表最大亮度值。

對數轉換 (log-transform)：對數對於較為陰暗的影像有較敏感的轉換能力，因此當影像有部分地方較為黑暗需強化其訊號可以選用對數轉換將其強化，公式如下：

$$y = c \times \log(1 + x)$$

c 為自訂常數，本次實作令 $c = 50$ 來觀察影像轉換後的變化，圖片顯示如下：

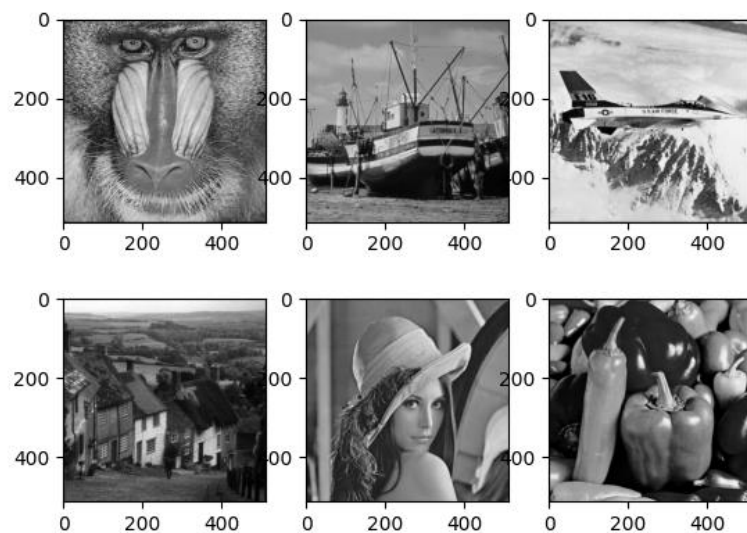


伽瑪轉換 (gamma-transform): 伽瑪轉換主要用於圖像校正，透過調整次方數來調整圖片過黑或過曝，此方法也經常使用在顯示螢幕的校正。

公式如下：

$$y = (L - 1) \times \left(\frac{x}{L - 1} \right)^\gamma$$

需要先將原本的亮度值轉換到 $[0, 1]$ 區間再進行冪次方轉換，此行為的目的在於讓冪次轉換後的亮度值仍在合理的值域，因此將 x 除上 $L - 1$ ，考慮觀察 $\gamma = 1.5$ 時影像轉換的變化，圖片顯示如下：



負影像 (image negative)：

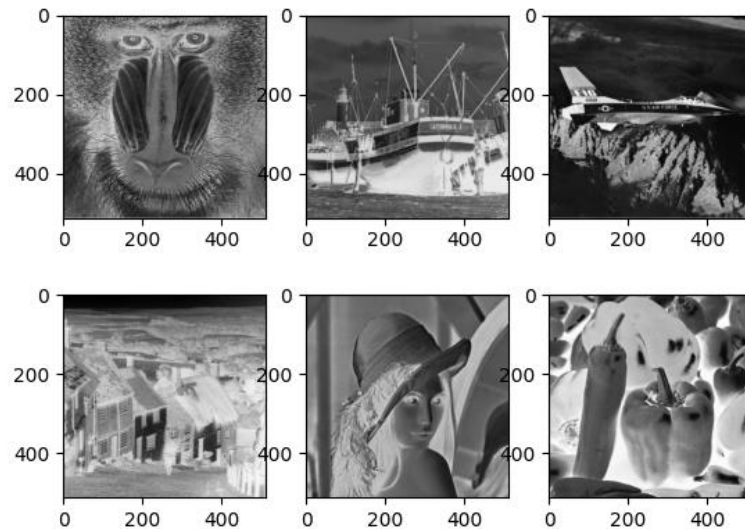
負影像的主要功能便是將圖片的亮暗對換，藉此可將圖像當中的重點轉移到

不同部分。

公式如下：

$$y = (L - 1) - x$$

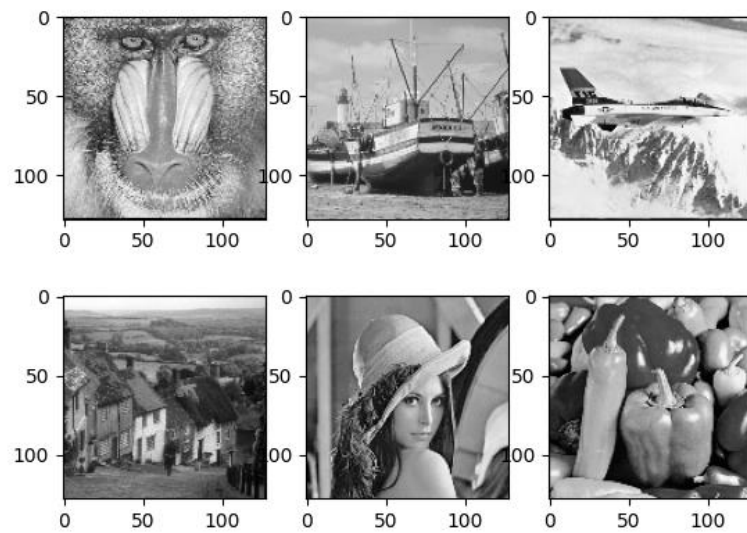
圖片顯示如下：



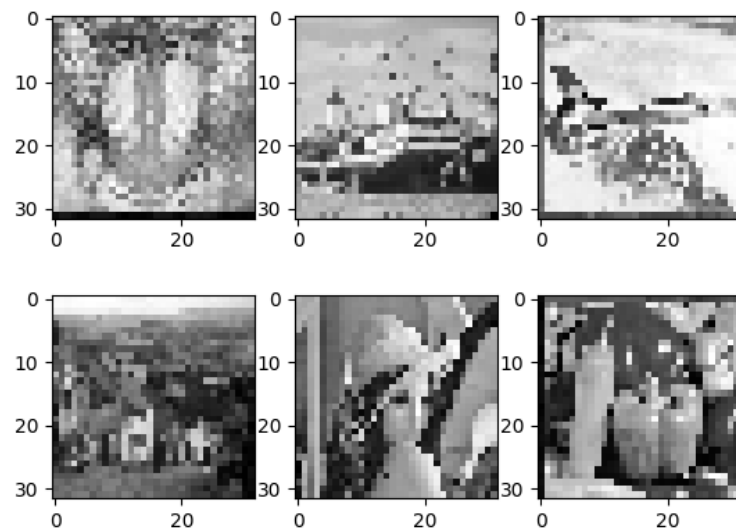
(c) 利用雙線性插值 (Bilinear)以及最近鄰居插值 (Nearest neighbor)兩種方法對圖像進行上採樣 (upsampling)與下採樣 (downsampling)：

雙線性插值 (Bilinear)：雙線性插值是以線性插值拓展到二維空間的方法，在插值上同時考慮 4 個鄰近點並且以各點的距離作為權重作為填值標準，實作上便是將欲轉換的目標大小縮放到原始尺度，找尋各軸上最相鄰的位置，比起最近鄰居插值法考慮到更多資訊，放大圖片所造成的馬賽克狀效果較輕微。

original image (512 × 512) to (128 × 128)：

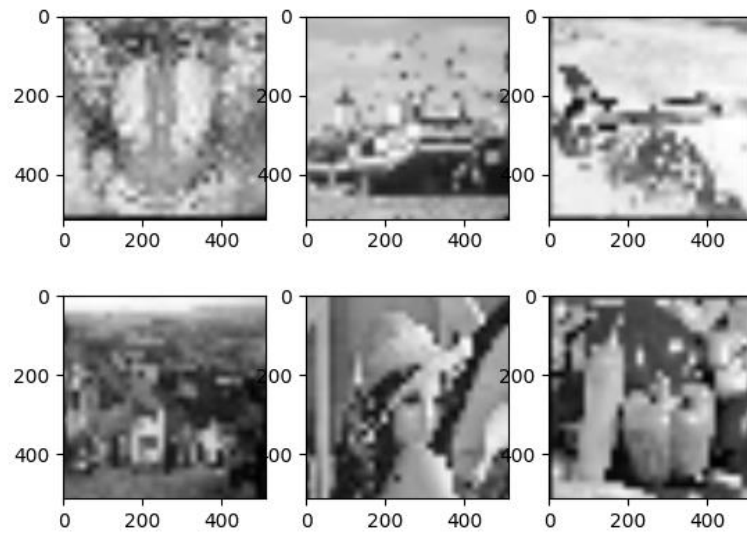


original image (512×512) to (32×32) :

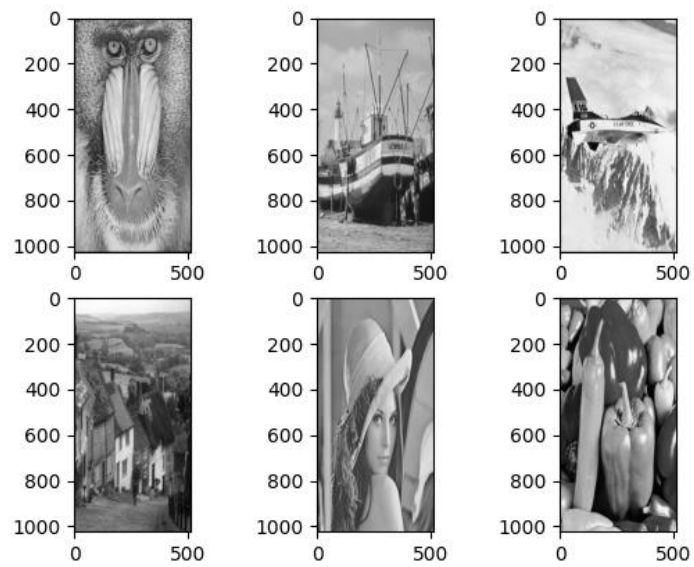


original image (32×32) to (512×512) :

由於沒有 32×32 的原圖，因此本次作業裡先將原圖的 512×512 縮小到 32×32 ，再放大到 512×512 。

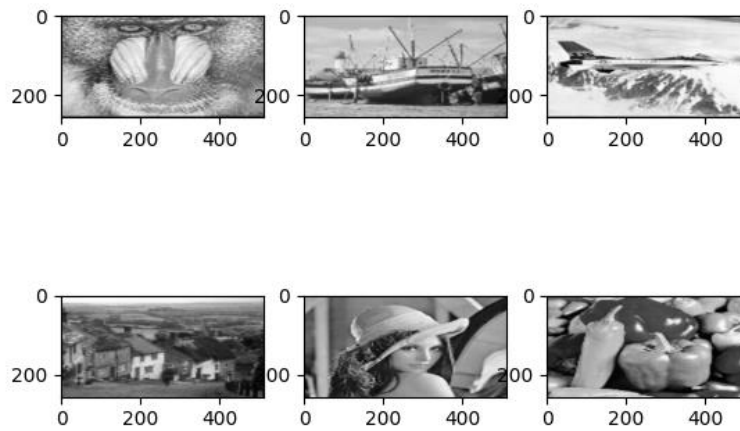


original image (512×512) to (1024×512) :



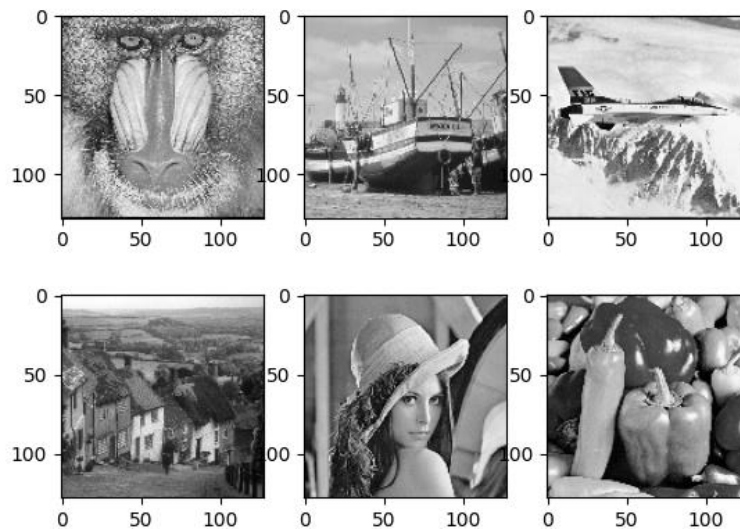
original image (128×128) to (256×512) :

由於沒有 128×128 的原圖，因此本次作業裡先將原圖的 512×512 縮小到 128×128 ，再放大到 256×512 。

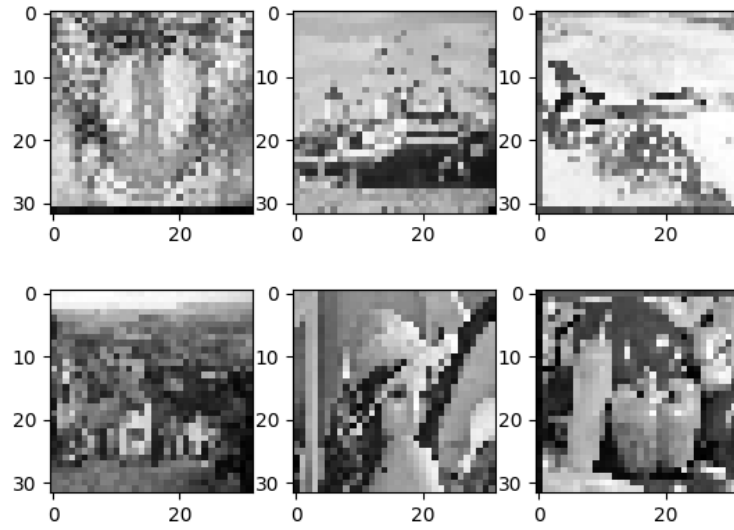


最近鄰居插值 (Nearest neighbor)：最近鄰居插值為一概念單純的方法，主要就是以最相鄰的數值進行填補，實作上便是將欲轉換的目標大小縮放到原始尺度，找最相鄰的位置，但因為只考慮單點，放大圖片時所造成的馬賽克狀效果相當明顯。

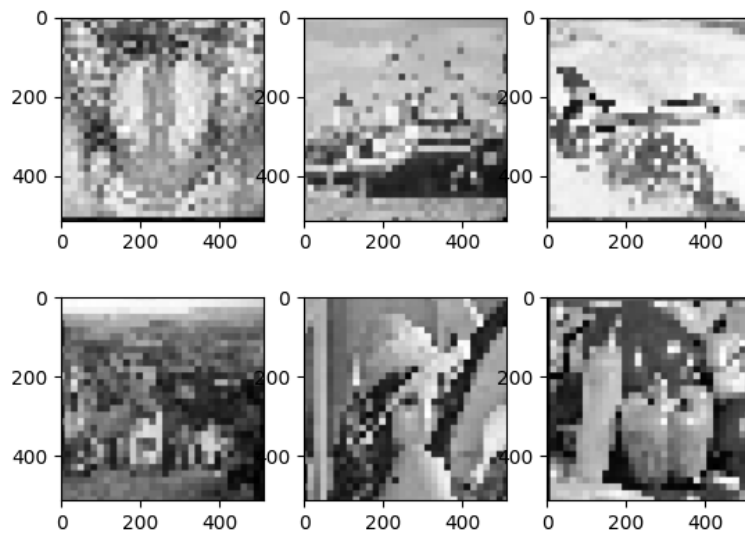
original image (512×512) to (128×128)：



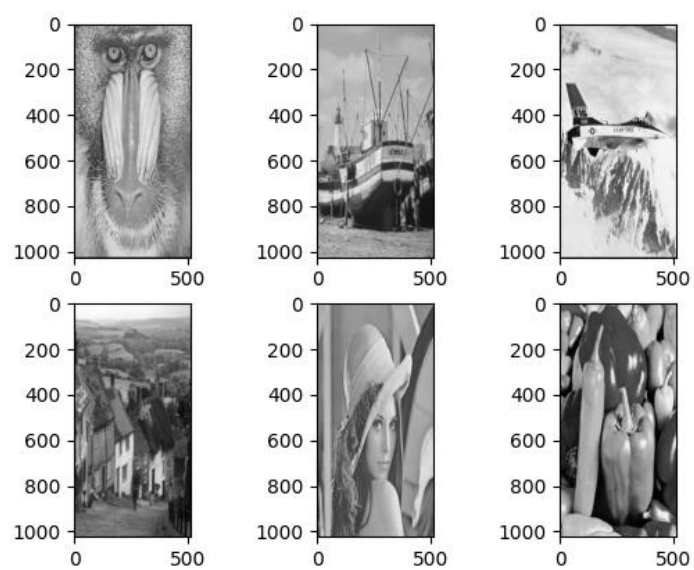
original image (512×512) to (32×32)：



original image (32×32) to (512×512) :



original image (512×512) to (1024×512) :



original image (128x128) to (256x512) :

