

高等影像處理

作業 5 書面報告

學號： 61247001S

姓名： 蕭融

一、本作業所用之程式語言及編譯器

程式語言：python

編譯工具：pyinstaller

二、程式功能

1. 讀寫影像檔案並對影像進行旋轉與存檔。
2. 本程式可讀入的影像檔格式包含 JPG 檔、BMP 檔，以及 PPM 檔，輸出的影像檔格式為 JPG 檔、BMP 檔，以及 PPM 檔。
3. 畫灰階影像直方圖
4. 添加高斯雜訊或椒鹽雜訊
5. 進行卷積
6. 進行直方圖均衡化

三、程式流程或演算法

(請附程式流程圖或演算法)

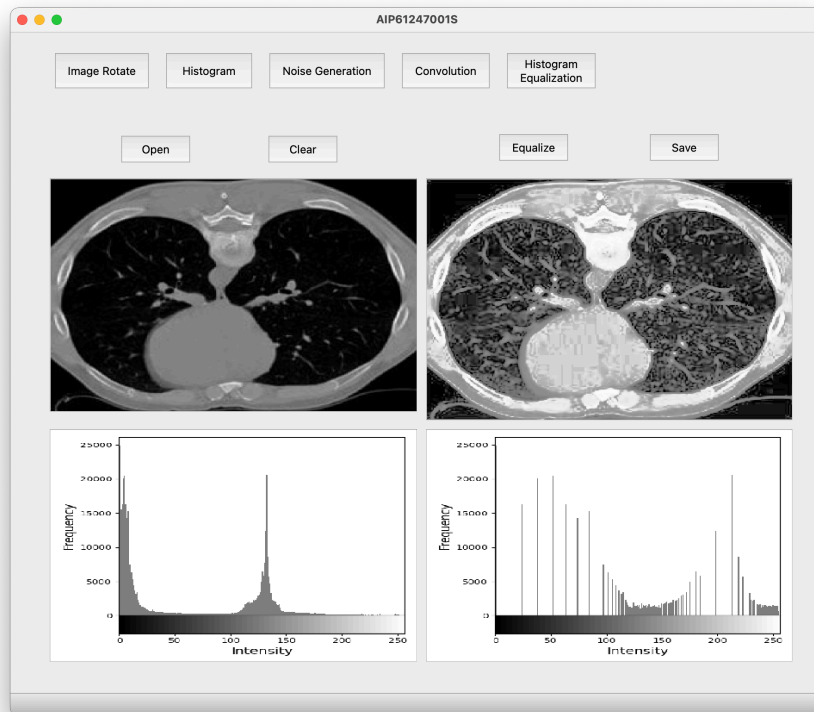
```
def equalization_image(image):
    if image is None:
        return
    G = 256
    height, width, channel = image.shape
    transparent_count = (image[:, :, 3] == 0).sum() if image.shape[2] == 4 else 0

    equal_image = np.zeros_like(image, dtype=np.uint8)
    for c in range(channel):
        if c < 3:
            h, _ = np.histogram(image[:, :, c].ravel(), 256, [0, 256])
            h[0] -= transparent_count
            for i in range(1, G):
                h[i] += h[i-1]
            g_min = min(np.nonzero(h)[0])
            h_min = h[g_min]
            t = np.round((h - h_min) / (height * width - h_min - transparent_count) * (G - 1))

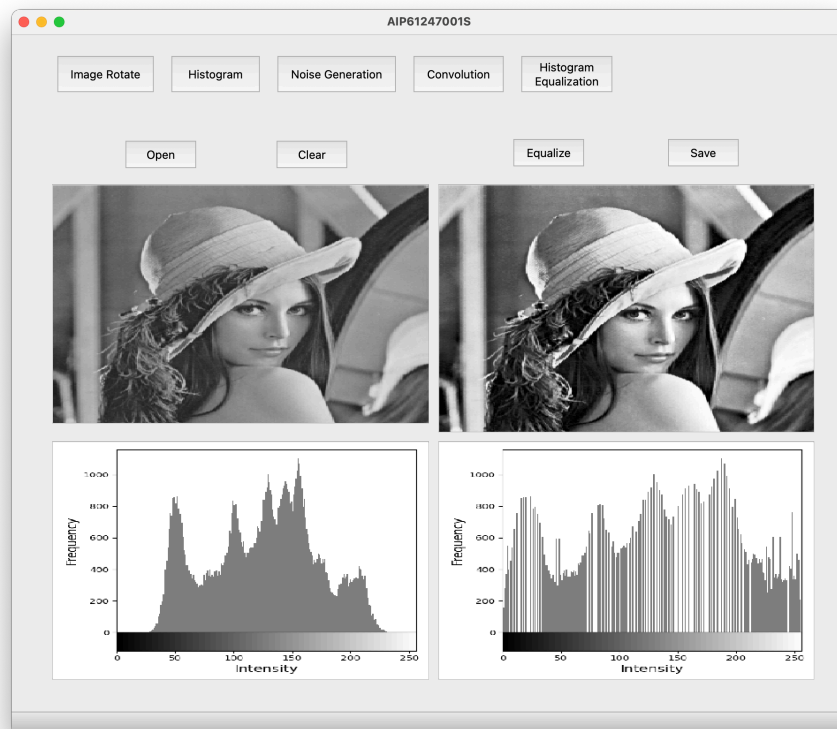
            for i in range(height):
                for j in range(width):
                    if image.shape[2] == 4 and image[i, j, 3] == 0:
                        continue
                    equal_image[i, j, c] = t[image[i, j, c]]
    return equal_image, transparent_count
```

四、測試結果(請附至少四組畫面截圖，二組做影像平滑化，二組做邊緣偵測，並說明附對應的 convolution masks)

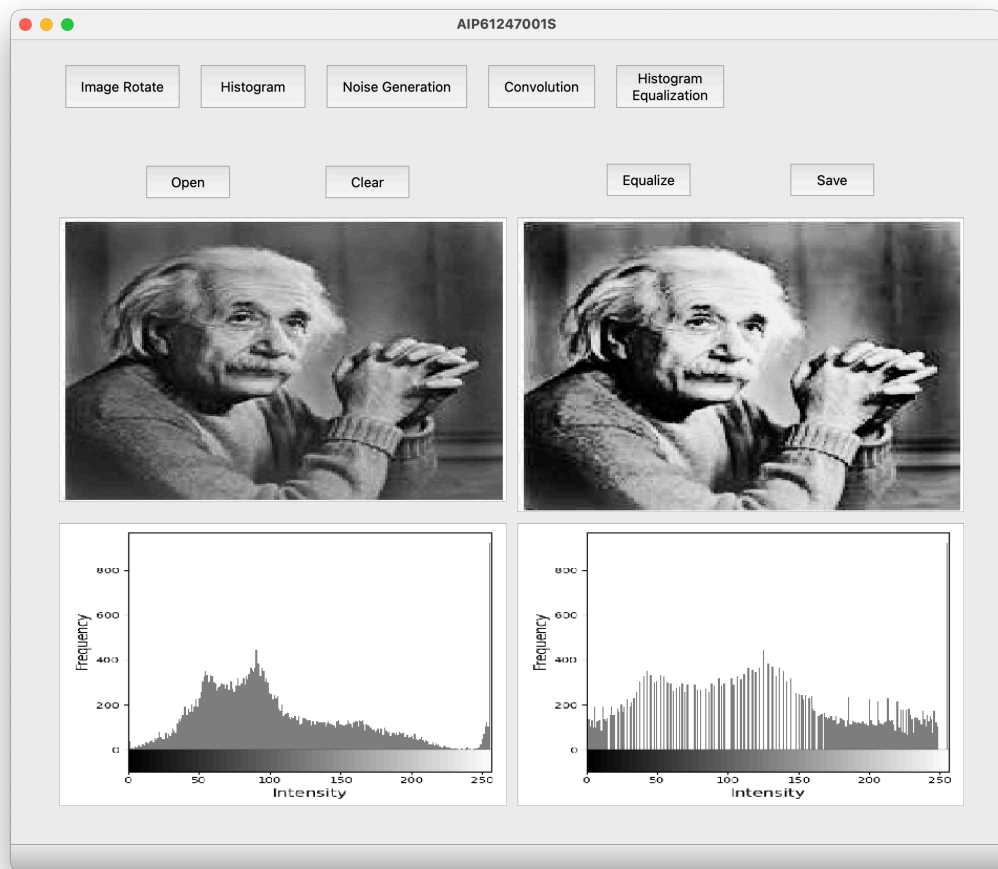
範例一可以看到原本不明顯的深色部分可以看清楚差異。



範例二可以看到原本黑的地方更黑，白的地方更白。



範例三發現黑白更分明。



五、 程式撰寫心得(至少 100 字)

這次的作業也算簡單，利用作業二的直方圖再加以處理，就能得到均衡化的圖了，雖然黑白的圖片很快速就可以完成，但我不確定彩色的圖片是否為正確的，另外還有透明度的像素，會讓整個感覺很複雜，所以我雖然有寫，但好像不太對。

P.S. 開啟/儲存/清除檔案的按鈕在上方工具列的 File 中，可以使用 Ctrl+O 和 Ctrl+S 快捷鍵，點擊左邊空白圖片也可以快速開啟圖片。使用 Ctrl+L 可以快速清除圖片。