

thresholding

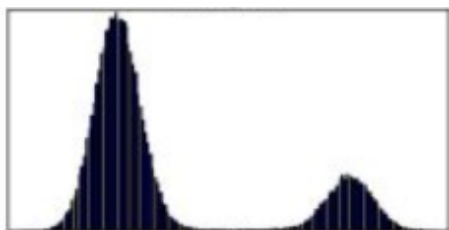
1. Global thresholding: 先將輸入圖像轉為 2 進制圖像，在定義一個閾值，大於他為 1，小於他為 0，也就是說大於他為白色小於他為黑色。

$$g(x, y) = \begin{cases} 1 & \text{if } (x, y) > T \\ 0 & \text{if } (x, y) \leq T \end{cases}$$

假設以 0~7 所示，大於 4 為 1，小於 4 為 0

INPUT					OUTPUT			
3	0	1	5	Binary T=4	0	0	0	1
7	6	0	4		1	1	0	0
2	7	0	6		0	1	0	1
1	3	5	5		0	0	1	1

2. Adaptive thresholding: 適應式閾值，在定義閾值時，透過鄰近平均值、中位數或者最大最小值相加除以 2 來當作閾值。
3. OTSU thresholding: 透過一個演算法得出來的值定義為閾值，第一步為計算強度級的值方圖與機率為 $w_i(0)$ 和 $u_i(0)$ 的初始值再來搜尋可能是閾值的數值進行變異數運算得出兩個最大變異數，所需的閾值就是兩個最大變異數相加除以 2，對於雙峰像較為準確。

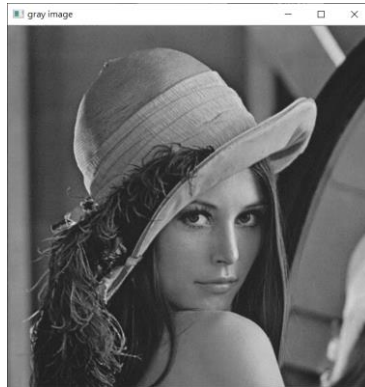


實作

分別對 opencv 範例、藥丸、自己的照片、課本照片進行 thresholding



Step 1 灰階 :在轉成 Thresholding 時先將圖片轉成灰階圖在進行操作。

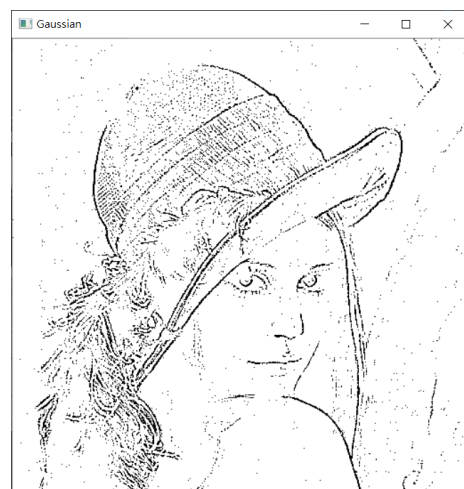


Step 2 thresholding

1. Global thresholding 閾值設為 100 的效果



2. Adaptive thresholding (左方是以平均值下去做閾值的，對於邊緣較為明顯，右方是經過高斯濾波)



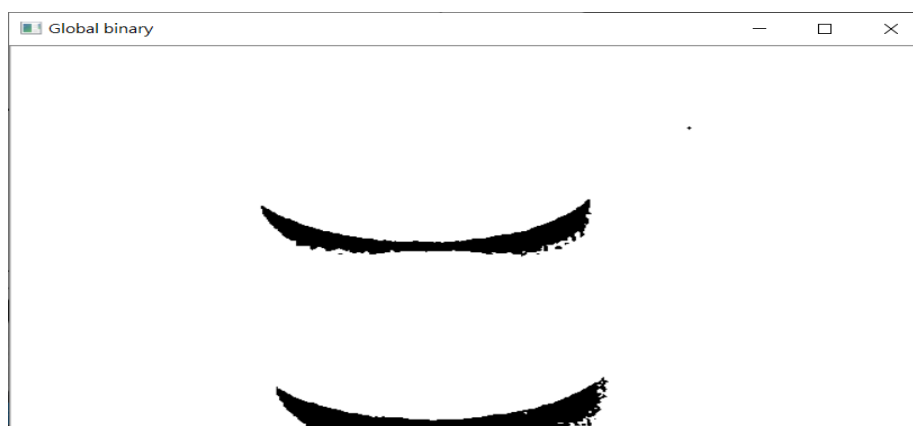
3. OTSU thresholding



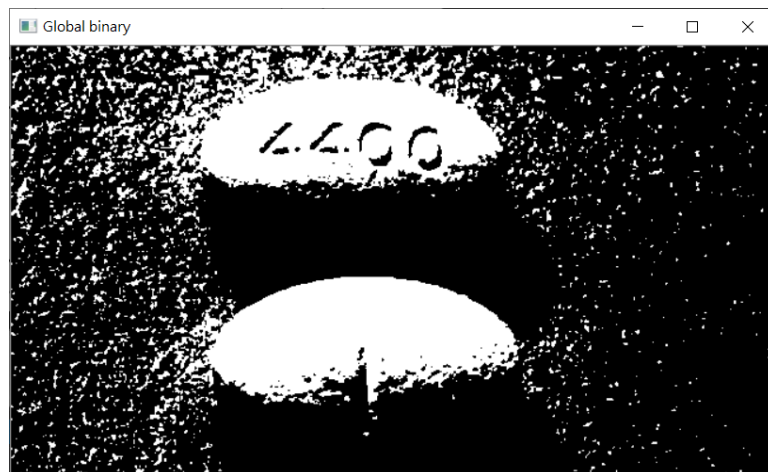
PILL 灰階



1. Global thresholding 閾值設為 100 的效果，看得出來只有陰影低於 100。



以下是閾值設為 240 的效果，才勉強看得出藥丸上的數字



2. Adaptive thresholding:邊緣較為明顯



3. OTSU thresholding:因為藥丸這張圖片大部分比較偏白色，沒有一個很明顯的雙峰直線圖，所以在取 OTSU 實相當不物準。



課本數獨圖灰階



1. Global thresholding: 在閾值設為 100 實看得出來下角因為亮度關係所以低於 100 幾乎都看不到了。



2. Adaptive thresholding: 因為是以鄰近的點來做平均運算，相較 Global thresholding 更為清楚。(右方是經過高斯濾波)



3. OTSU thresholding:



我的照片灰階



1. Global thresholding: 跟上方數獨突有相似問題右側的臉因為光線，在經過 thresholding 呈現較黑。



2. Adaptive thresholding: 一樣是有解決光線問題，且看得出來對於邊緣較為明顯。(右方是經過高斯濾波)



3. OTSU thresholding:

