

# Edge Detection

408410071 謝明煌

Jan. 5, 2022

## Technical description

請執行 main.m 或 image\_process\_hw4.exe 來執行程式碼。

Sobel 和 LoG 處理都已做成 function 檔的形式方便閱覽及彙整程式

碼，執行 main.m 即會自動呼叫那些 function 來處理影像。

註：這些 function 並非內建的，而是自己寫的。

### 1. Sobel 處理

對應檔案：mySobel.m

(程式碼有用註解標示每個部份，可配合參考)

先分別使用水平、垂直、 $45^\circ$ 、 $-45^\circ$  的 Sobel mask (如下圖) 計算

出四張邊緣圖，然後再對四張圖計算平均值得出最終結果，以確

保各個方向的邊緣都有被標示出來。

-1	-2	-1	-1	0	1	0	1	2	-2	-1	0
0	0	0	-2	0	2	-1	0	1	-1	0	1
1	2	1	-1	0	1	-2	-1	0	0	1	2

### 2. Laplacian of a Gaussian (LoG) 處理

對應檔案：myLoG.m

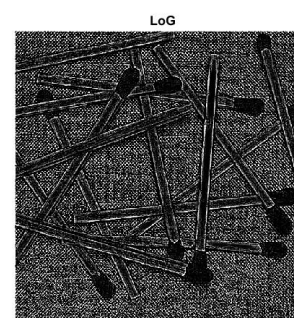
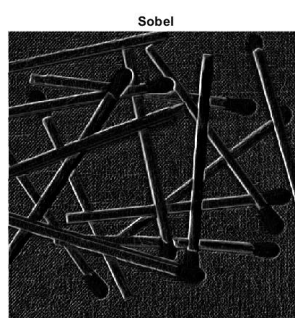
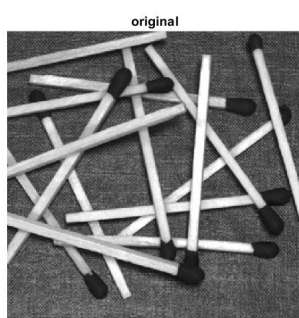
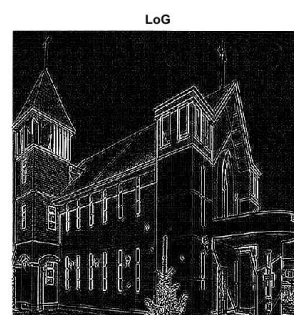
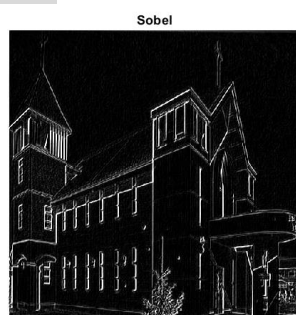
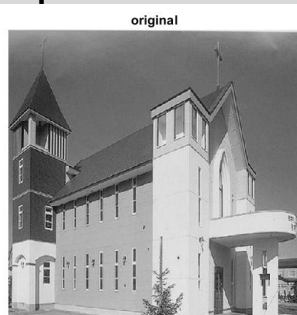
(程式碼有用註解標示每個部份，可配合參考)

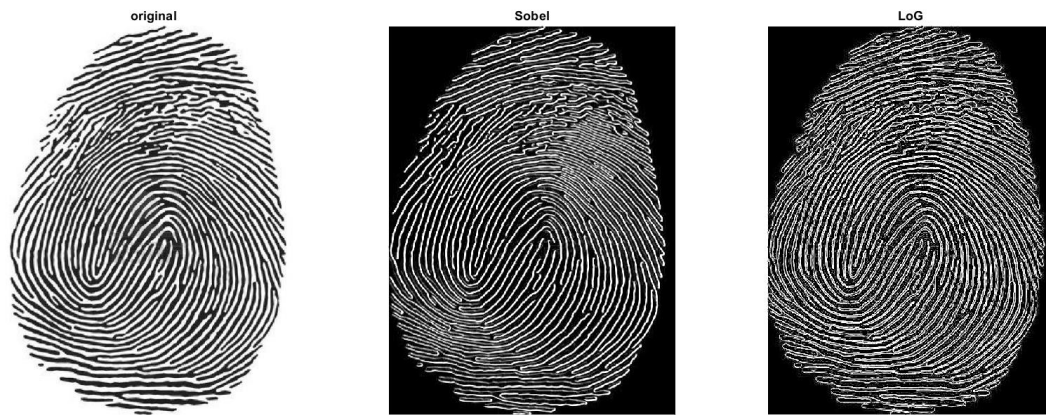
將圖片套用如下圖的 Laplacian of a Gaussian mask 即可。

(註:該 mask 的 size 為  $5 \times 5$ ， $\sigma=0.3$ ，由於真正的數值計算出來會有小數點，同時為了處理離散化(數位化)的資料，因此這個 mask 是近似值，並非真正數值)

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

## Experimental results





## Discussions

從結果可以發現，LoG 比 Sobel 可以抓出更多邊緣，推測是因為 Sobel 僅判斷四個方向的梯度變化，而 LoG 把周圍所有像素都納入考量，所以抓邊緣的效果更好。

然而，我們也可以發現 LoG 即使已經經過 Gaussian filter 平滑化，還是將雜訊及背景紋路也當作邊緣標示出來，推測可能平滑化程度不夠。我們可以透過將  $\sigma$  提高來加強平滑化程度，但如此一來也會使部分真正的邊緣無法被偵測出來，經實驗將  $\sigma$  設為 0.5 會和 Sobel 差不多，因此還是以  $\sigma=0.3$  做 demo 和比較。

## References and Appendix

None