

# 影像處理

## (Image Processing)

真理大學 資訊工程系 吳汶涓老師

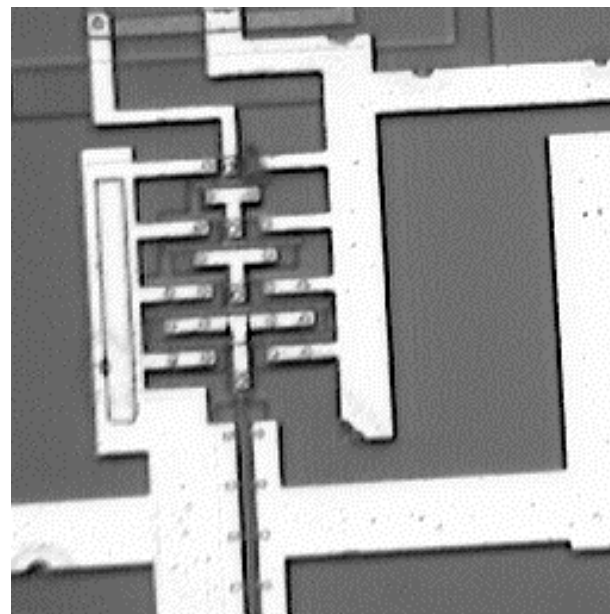
### Course 9

#### 影像分割



# Outline

- 9.1 前言
- 9.2 閾值運算
- 9.3 閾值的應用
- 9.4 設定合適的閾值
- 9.5 可適性閾值運算
- 9.6 邊緣偵測
- 9.7 導數與邊緣
- 9.8 二階導數
- 9.9 Canny邊緣偵測
- 9.10 Hough轉換
- 9.11 在Matlab中執行Hough轉換



## 9.8 二階導數

- 另一類邊緣偵測方法是使用二階導數。
  - Laplacian (水平+垂直方向)

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \Rightarrow \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

- **各向同性濾波器** (isotropic filter)：旋轉不受影響
- 缺點：二階導數濾波器非常容易受到雜訊影響

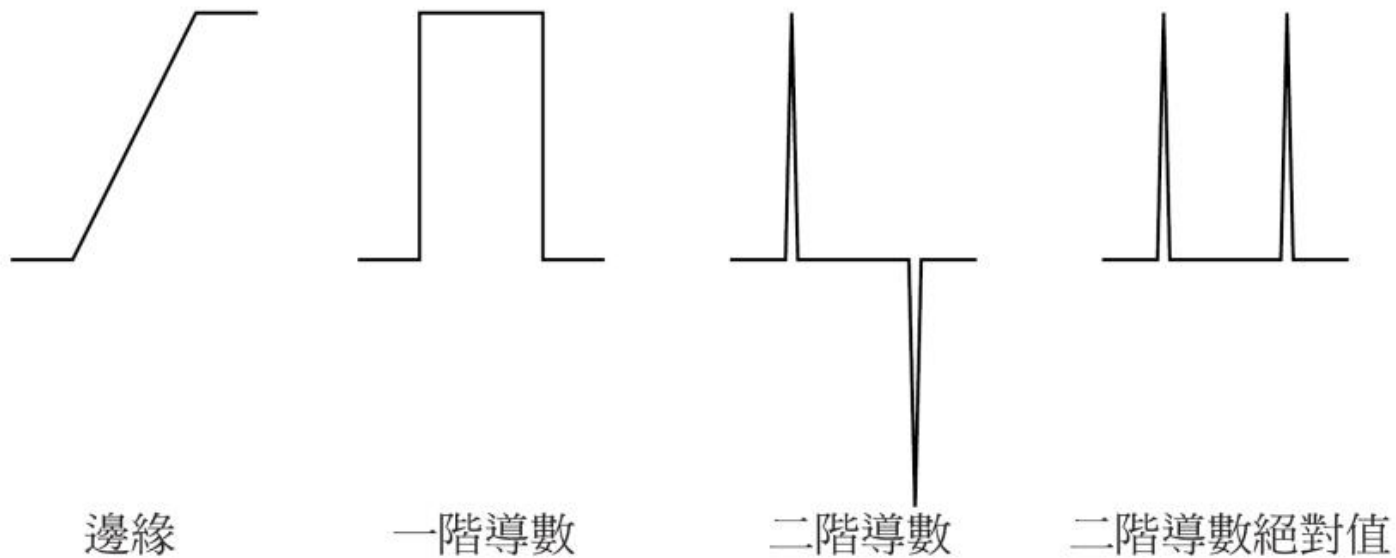
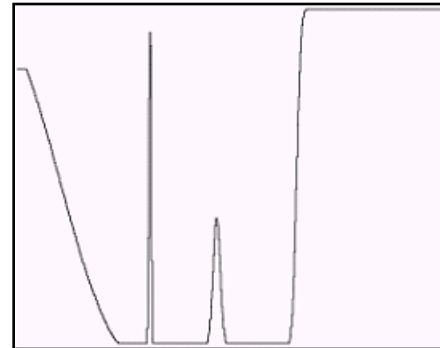
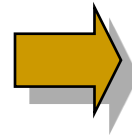
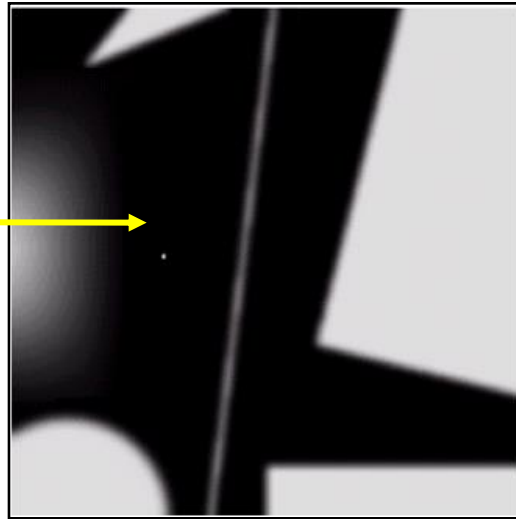
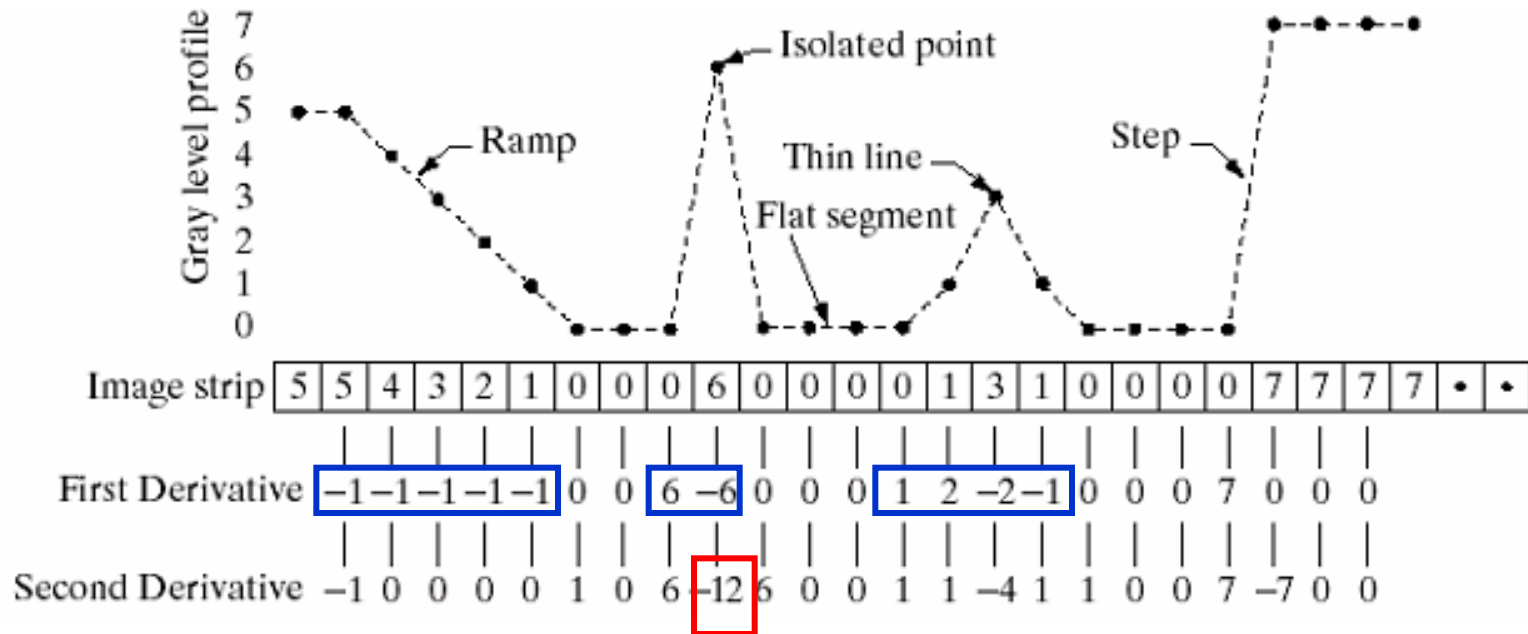


圖 9.21 邊緣函數的二階導數



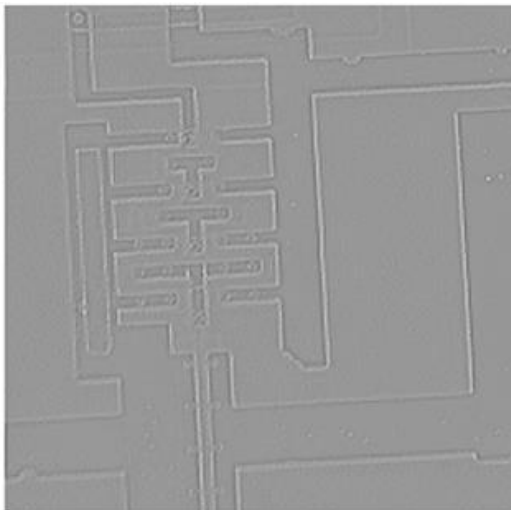
Image



在孤立的雜訊點中，二階梯度會於此處產生較大的強度

```
>> l=fspecial('laplacian',0);  
>> ic_l=filter2(l,ic);  
>> figure,imshow(mat2gray(ic_l))
```

α 值



其他Laplacian遮罩：

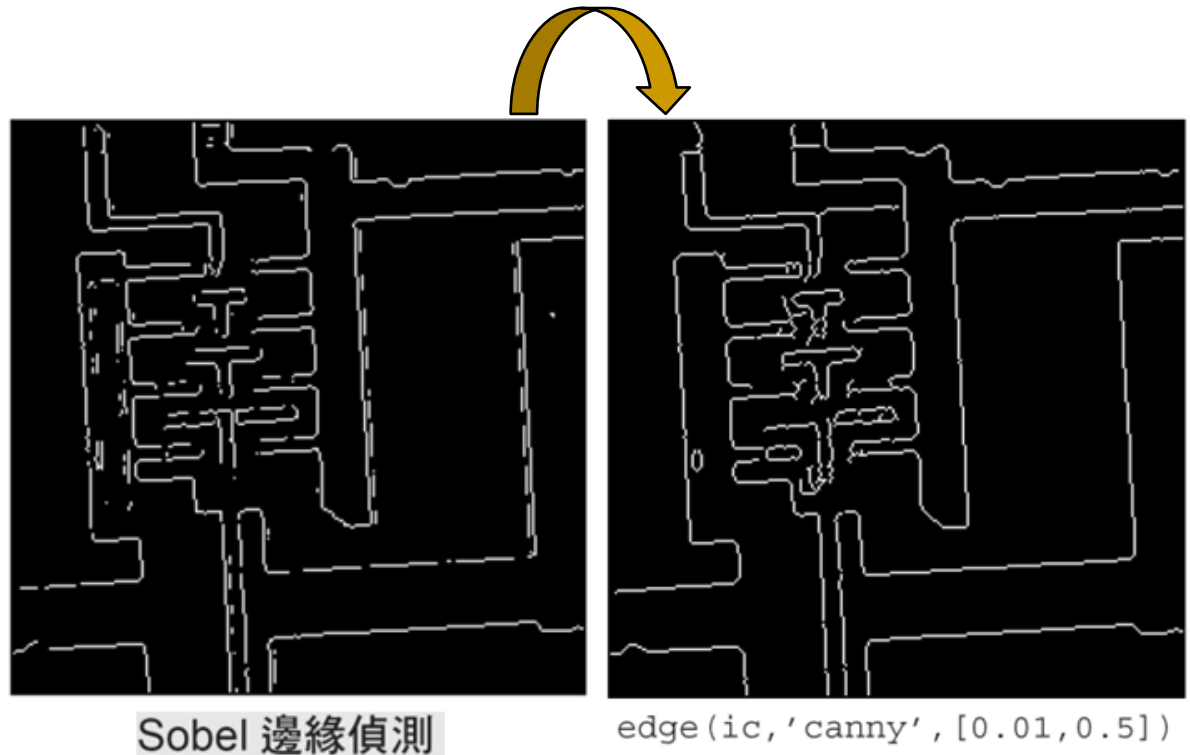
1	1	1
1	-8	1
1	1	1

-2	1	-2
1	4	1
-2	1	-2

圖 9.22 離散 Laplacian 濾波的結果

## 9.9 Canny邊緣偵測

- 1986年由John Canny提出的方法，主要：
  - 降低偵錯錯誤率
  - 邊緣定位
  - 單一反應



1. 影像為  $x$ 。
2. 產生一維高斯濾波器  $g$ 。(高斯濾波器能平滑影像)
3. 產生對應方程式(9.1)的一維濾波器  $dg$ 。

$$\left(-\frac{x}{\sigma^2}\right) e^{-\frac{x^2}{2\sigma^2}} \quad (9.1)$$

4. 將  $g$  與  $dg$  卷积，得到  $gdg$ 。
5. 將  $x$  代入  $gdg$ ，產生  $x_1$ 。
6. 將  $x$  代入  $gdg'$ ，產生  $x_2$ 。
7. 使用下列算式求得邊緣影像：
$$x_e = \sqrt{x_1^2 + x_2^2}$$



- 將邊緣影像 $x_e$ 經過閾值運算，以便去除其他像素，只留下邊緣。
- **非最大值抑制**（non-maximum suppression）

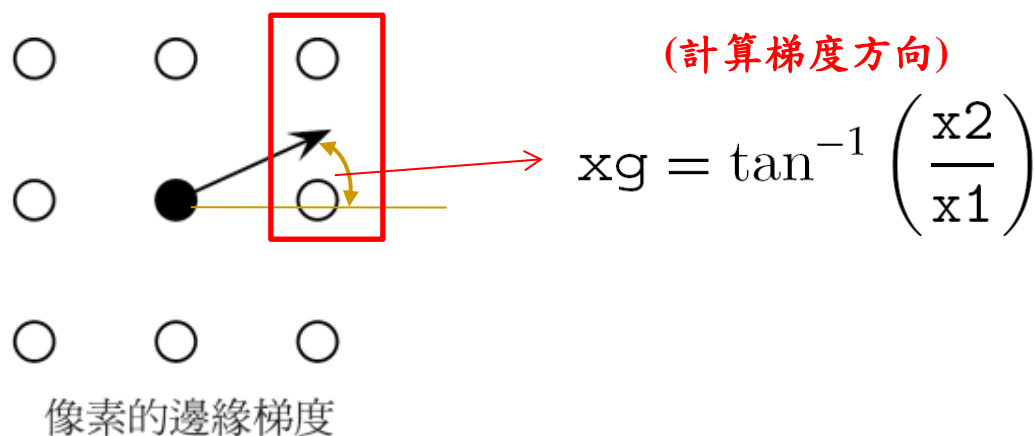


圖 9.26 Canny 邊緣偵測的非最大值抑制

區域	梯度方向	像素位置
1	$0^\circ$	$y \leq 0$ 且 $x > -y$
(1)	$0^\circ$	$y \geq 0$ 且 $x < -y$
2	$45^\circ$	$x > 0$ 且 $x \leq -y$
(2)	$45^\circ$	$x < 0$ 且 $x \geq -y$
3	$90^\circ$	$x \leq 0$ 且 $x > y$
(3)	$90^\circ$	$x \geq 0$ 且 $x < y$
4	$135^\circ$	$y < 0$ 且 $x \leq y$
(4)	$135^\circ$	$y > 0$ 且 $x \geq y$

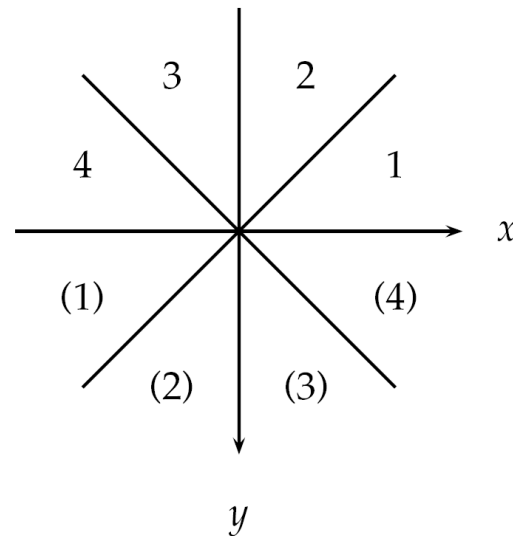


圖 9.27 使用像素位置量化梯度方向

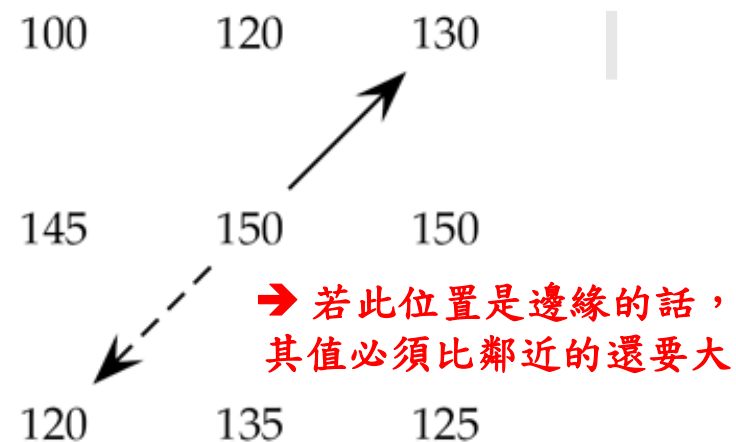


圖 9.28 非最大值抑制量化

□ **磁滯閾值**（hysteresis thresholding）： $t_H$ ,  $t_L$

- 邊緣像素：高於 $t_H$ 的，或是與邊緣像素相鄰且其值界在 $t_H$ 與 $t_L$ 之間

```
>> [icc,t]=edge(ic,'canny');  
>> t  
  
t =  
  
    0.0500    0.1250  
  
>> imshow(icc)
```

或是 自行給閾值:

```
>> icc=edge(ic, 'canny', [0.3, 0.8];  
>> imshow(icc)
```

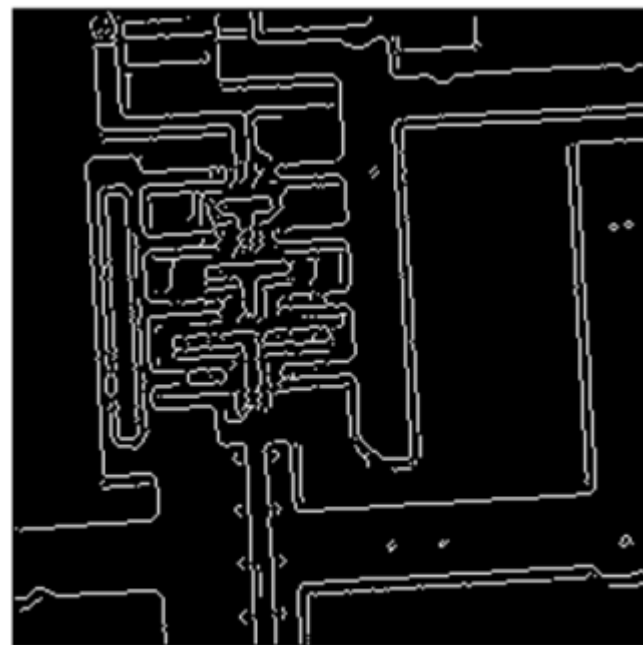
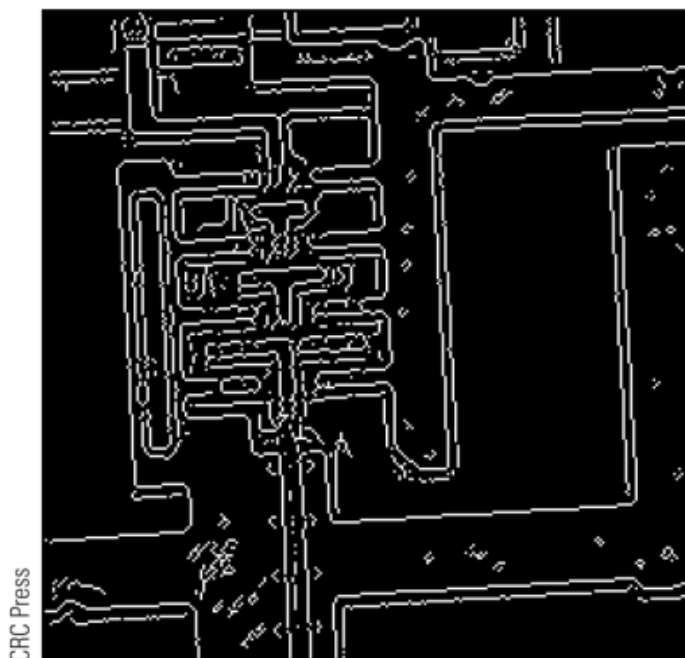
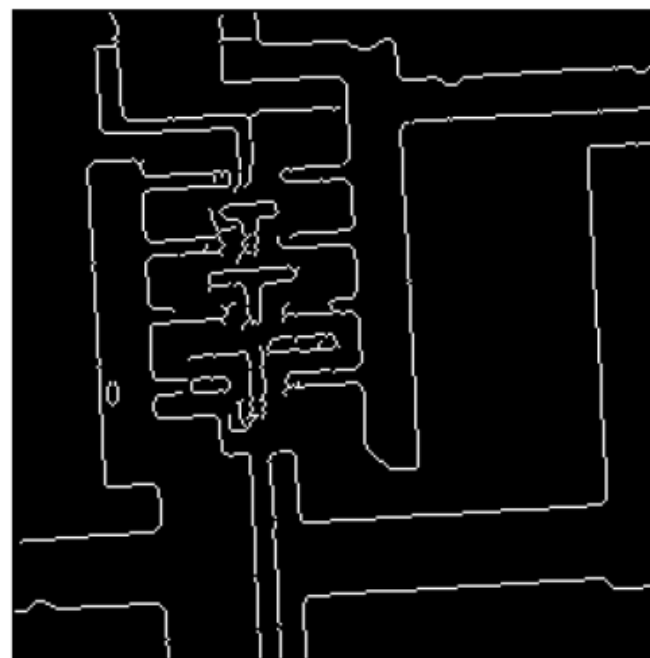


圖 9.29 Canny 邊緣偵測

**Q:** 閾值設越高，會有怎樣的情況呢??



`edge(ic, 'canny', [0, 0.05])`



`edge(ic, 'canny', [0.01, 0.5])`

圖 9.30 不同閾值的 Canny 邊緣偵測