中央大學資工系 103 學年度第一學期
授課教師：陳慶瀚

單元 5、空間濾波

1. 空間迴旋積(Spatial Convolution)

一張影像可以在頻率域(frequency domain)或空間域(spatial domain)進行濾波。本章將介紹各種用在空間域率波的運算子，包括可濾除高頻雜訊的均值濾波器、中值濾波器、高斯濾波器，以及用來增強邊緣特徵的索貝爾濾波器、拉普拉斯濾波器。

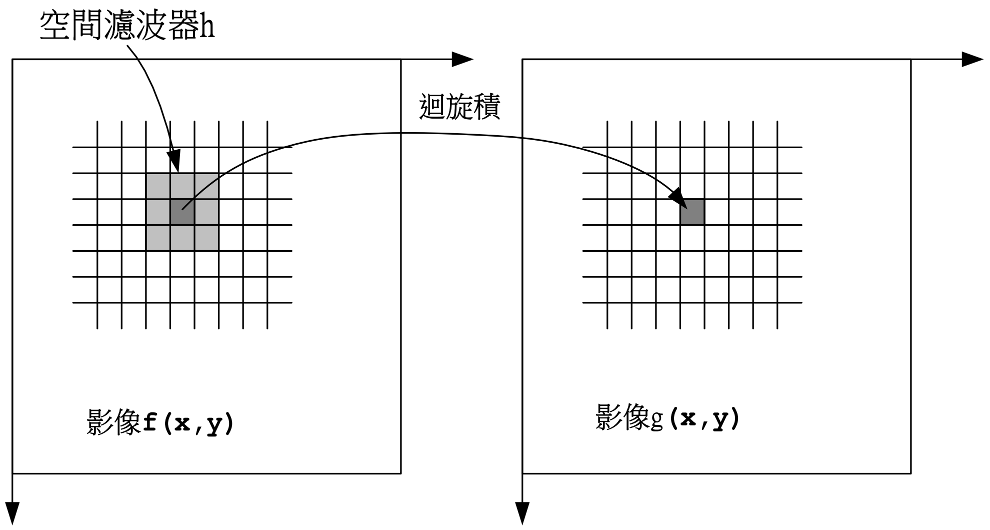
空間濾波器(spatial filter)h(i,j)又稱為 Mask/Kenel/Window，影像 f(x,y)經過空間濾波器的運算，得到濾波後的影像 g(x,y)，

$$g(i,j) = h(i,j) \odot f(i,j)$$

此一運算為迴旋積(convolution)，迴旋積的運算模式為"*shift-multiply-summation"* ,　shift 指由左到右、由上到下移動濾波器 h，針對每一次濾波器視窗所涵蓋的原始影像的區域進行相乘，最後累加所有乘積，得到濾波影像 g 上一個像素的值。假設空間濾波器的大小是 *M*× *M*，則我們可以寫成:

$$g(i,j) = \sum_{m=-\frac{M}{2}}^{\frac{M}{2}} \sum_{n-=\frac{M}{2}}^{\frac{M}{2}} h(m,n)\, f(i-m, j-n)$$

示意圖如下：




```
void convolution(uc2D &ima1, f2D &h, uc2D &ima2)
{
    int i,j,ii,jj;
    int N=h.nr/2;
    int mac;
    for(i=N;i<ima1.nr-N;i++)for(j=N;j<ima1.nc-N;j++)
    {
        mac = 0;
        for(ii=-N;ii<=N;ii++)for(jj=-N;jj<=N;jj++)
        {
            mac = mac + ima1.m[i+ii][j+jj]*h.m[ii+N][jj+N];
        }
        ima2.m[i][j]=mac;
    }
}
```

2. 用於平滑化影像的線性濾波器

A. 均值濾波器(mean filter)

一個 3x3 均值濾波器是最簡單的平滑化濾波器(smoothing filter)，其型態如下

$$\frac{1}{9} \times \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline\end{array}$$

或

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

B.泛用型權值平滑濾波器(generalized weighted smoothing filter)

$$g(x,y)=\frac{\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)f(x+s,y+t)}{\sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)}$$

summation of all coefficient of the mask

一個 3x3 權值平滑濾波器例子

$$\frac{1}{16}\times\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

```
/*-----------------------------------------------------------------*/
//          Convolution of image with mean filter
//
//                 陳慶瀚，2004.10.12
/*-----------------------------------------------------------------*/

#include <fstream.h>
#include <math.h>
#include "array.h"
void mean_filter_design(f2D &h, int winsize);
void convolution(uc2D &ima1, f2D &h, uc2D &ima2);
void main()
{
   ifstream in("120x120.raw",ios::binary);
   ofstream out("bin.raw",ios::binary);
   uc2D ima1,ima2;
   ima1.Initialize(120,120);
   ima2.Initialize(120,120);
   char c;
```

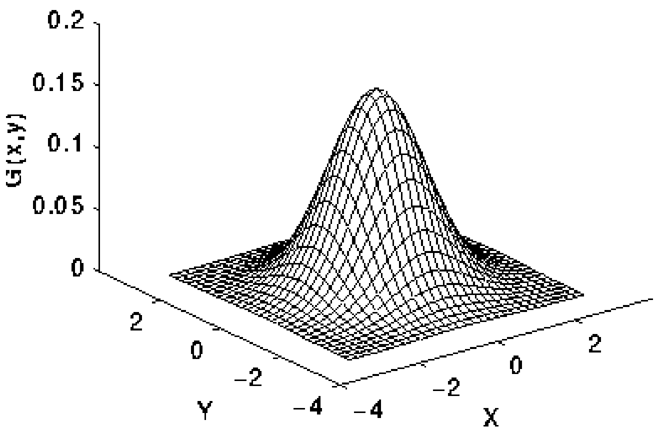
```
  for(int i=0;i<ima2.nr;i++)for(int j=0;j<ima2.nc;j++)
  {
    in.get(c);ima1.m[i][j]=ima2.m[i][j]=c;
  }

  f2D h; //filter
  mean_filter_design(h, 3);
  convolution(ima1, h, ima2);
  for(int i=0;i<ima2.nr;i++)for(int j=0;j<ima2.nc;j++)
    out<<ima2.m[i][j];
}

void mean_filter_design(f2D &h, int winsize)
{
  h.Initialize(3,3); // Size initialize by 3X3
  for(int i=0;i<h.nr;i++)for(int j=0;j<h.nc;j++)
   h.m[i][j]=1.0/9.0; // mean filter's coefficients
}

void convolution(uc2D &ima1, f2D &h, uc2D &ima2)
{
    int i,j,ii,jj;
    int N=h.nr/2;
    int mac;
    for(i=N;i<ima1.nr-N;i++)for(j=N;j<ima1.nc-N;j++)
    {
        mac = 0;
        for(ii=-N;ii<=N;ii++)for(jj=-N;jj<=N;jj++)
        {
            mac = mac + ima1.m[i+ii][j+jj]*h.m[ii+N][jj+N];
        }
        ima2.m[i][j]=mac;
    }
}
```

C. 高斯平滑濾波器(Gaussian smoothing filter)

在一個二維空間，高斯分布的函數定義為：The Gaussian distribution in 1-D has the form:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

假設高斯分布的平均值為 0，而 σ 是高斯分布的標準差，如果 σ=1，我們可以得到以下分布型態圖：



離散型態的高斯濾波器使用整數來逼近高斯函數，例如對於一個 5x5 的高斯濾波器(σ=1)，我們可以得到以下的濾波器係數：

$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

高斯濾波器是一個平滑化濾波器，平滑化程度是由標準差 σ 來控制，σ 值越大，平滑程度越高，相對的，影像越模糊。

作業 1、
設計一個 Gaussian_filter( ) 函式，測試 3 個不同的 σ 值(σ=1, σ=2 和 σ=5)和兩種不同的濾波器大小(filtersize=5 和 filtersize=11)共六種組合，以 mantis500x500.raw 影像為對象，觀察不同濾波效果。

3. 用於平滑化影像的非線性濾波器

建立在濾波器範圍內的像素的統計排序(ranking)來決定濾波輸出，而非使用迴旋積：

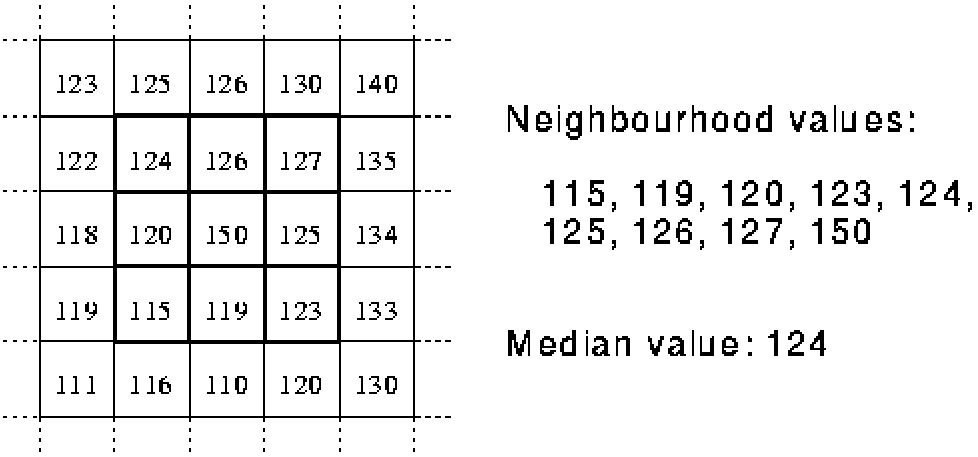
中值濾波器(median filter)：$h = \mathrm{median}\{g_k \mid k=1,2,\ldots, M\times M\}$

最大值濾波器(max filter)：$h = \max\{g_k \mid k=1,2,\ldots, M\times M\}$

最小值濾波器(min filter)：h = min{g_k | k=1,2,…, M× M}

A. 中值濾波器

中值濾波器藉由每一個 pixel 鄰近 pixel 灰階值排序的中間值來取代該 pixel 的灰階值。中值的計算是先將鄰近 pixel(濾波器視窗範圍)灰階值排序，在取出排序居中的值作為濾波器中間位置影像的像素值。例如



以下是使用 bubble sort 原理的中值濾波器的參考程式：

```
//-------------------------------------------------------
// Median filtering
// 2004.10.18 by CHEN Ching-Han
//-------------------------------------------------------

void Median(uc2D &InIm, uc2D &OutIm, int WinSize)
{
   int   r,c, y,x, i,j, n, Area;
   unsigned char  Buf;
   unsigned char  *Lst;

   n = (WinSize-1) >> 1;
   Area = (2*n+1) * (2*n+1);
   Lst = new char[Area];
   Int nr=InIm.nr, nc=InIm.nc;
   for (r=0; r<nr; r++)for (c=0; c<nc; c++)OutIm[r][c] = 0;
   for (r=n; r<nr-n; r++)
   {
     for (c=n; c<nc-n; c++)
     {
       i=0;
       for (y=-n; y<=n; y++)for (x=-n; x<=n; x++)
         {
             Lst [i] = InIm [r+y] [c+x];i++;}
/*---------- bubble sort -------------------*/
       for (i=0; i<Area-1; i++)for (j=Area-1; i<j; j--)
         if (Lst[j-1] > Lst[j])
       {
```

```
            Buf      = Lst[j-1];
            Lst[j-1] = Lst[j];
            Lst[j]   = Buf;
         }
         OutIm [r][c] = Lst [Area/2];  // get madian value
      }
   }
}
```

作業 2：
請分別以 7x7 的 median, max, min 濾波器對 mantis500x500.raw 影像作空間濾波，定性比較三者的差異。

B. 參數可調的中值濾波器—加權中值濾波器(Weighted median filter)

B. Weighted Median Filters

The weighted median (WM) filter was first introduced as a generalization of the standard median filter, where a nonnegative integer weight is assigned to each position in the filter window [13], [51]. In this subsection, we give two alternative definitions of WM filters and discuss the analogy between WM filters and linear FIR filters.

As shown in Fig. 1, the structure of a WM filter is quite similar to that of a linear FIR filter. For real-valued signals, WM filters can be defined in two different but equivalent ways. The first definition can be used in the common case of positive integer weights.

Definition 2.1: For the discrete-time continuous-valued input vector $\underline{X} = [X_1, X_2, \cdots, X_N]$, the output Y of the WM filter of span N associated with the integer weights

$$\underline{W} = [W_1, W_2, \cdots, W_N] \tag{2.10}$$

is given by

$$Y = \mathrm{MED}[W_1 \diamond X_1, W_2 \diamond X_2, \cdots, W_N \diamond X_N] \tag{2.11}$$

where MED[·] denotes the median operation and $\diamond$ denotes duplication, i.e.,

$$K \diamond X = \overbrace{X, \cdots, X}^{K \text{ times}}. \tag{2.12}$$

This filtering procedure can be stated as follows: sort the samples inside the filter window, duplicate each sample X_i to the number of the corresponding weight W_i and choose the median value from the new sequence.

Example 2.1: Consider a length 5 WM filter with integer weights [1, 2, 3, 2, 1]. Now apply the filter to the following sequence so that the window is centered at the sample value 8,

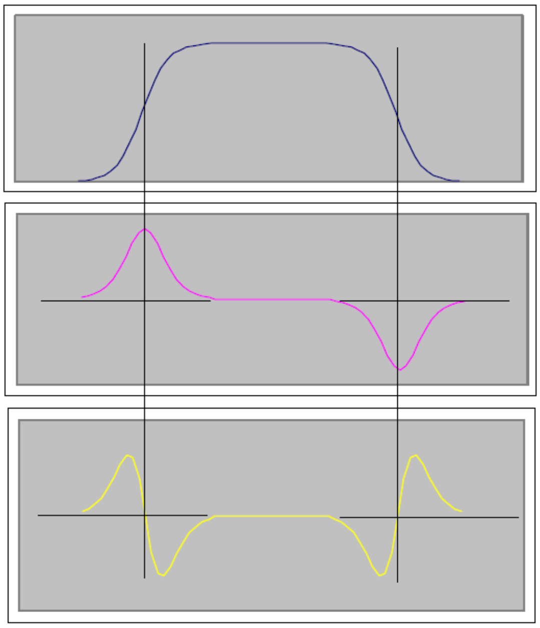
$$\underline{X} = [-1, 5, 8, 11, -2].$$

After sorting and duplication, the samples inside the filter window are 11, 11, 8, 8, 8, 5, 5, -1, -2. The filter output is $Y = 8$, whereas, the 5-point median would have produced the result $Y = 5$.

作業 3：
請參考 median 函式，設計一個 WeightedMedian 函式,並以 3X3 濾波器大小為例，執行加權中值濾波。使用不同的 W 來觀查加權中值濾波對 finger300x300 濾波效果。或假設 W= $\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 2 \\ 3 & 2 & 1 \end{bmatrix}$ 。

進階作業 3(option)：使用隨機最佳化或演化最佳化方法找出一組 W 係數(整數)，使得加權中值濾波器可以執行 Edge enhancement 的低通濾波。

4. 邊緣偵測濾波器(edge detection filter)



Edge profile

First derivative

Second derivative

A. Sobel 濾波器

Sobel 濾波器使用空間一階導數來增強高頻的空間訊號，在影像中，這些高頻訊號通常代表較銳利的物體邊緣或線條特徵。

一維的一階導數：

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

二維的一階導數以梯度(gradient)表示：

$$\frac{\partial f}{\partial x \partial y} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y} = \text{Gx} + \text{Gy}$$

Gx 和 Gy 分別表示 x 和 y 方向的梯度分量，其離散濾波器形式可以表示如下：

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

Gx

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

Gy

梯度的大小(amplitude)是

$$|G| = \sqrt{Gx^2 + Gy^2}$$

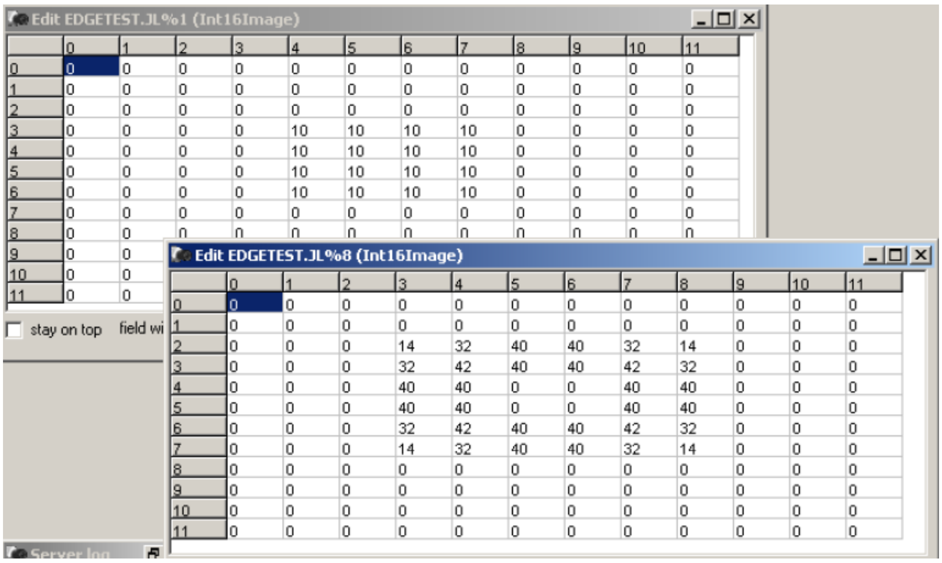
或

$$|G| = |Gx| + |Gy|$$

至於方向(orientation)則為

$$\theta = \arctan(Gy/Gx)$$

作業 4：
設計一個 sobel 濾波器函式，輸入 11x11 的整數值資料檔，輸出 amplitude 資料表和 orientation 資料表，如下圖。另外(option)繪出 orientation 方向圖(4 個方向即可)。


Edit EDGETEST.JL%1 (Int16Image)

	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	10	10	10	10	0	0	0	0
4	0	0	0	0	10	10	10	10	0	0	0	0
5	0	0	0	0	10	10	10	10	0	0	0	0
6	0	0	0	0	10	10	10	10	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0										
10	0	0										
11	0	0										

stay on top field wi

Edit EDGETEST.JL%8 (Int16Image)

	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	14	32	40	40	32	14	0	0	0
3	0	0	0	32	42	40	40	42	32	0	0	0
4	0	0	0	40	40	0	0	40	40	0	0	0
5	0	0	0	40	40	0	0	40	40	0	0	0
6	0	0	0	32	42	40	40	42	32	0	0	0
7	0	0	0	14	32	40	40	32	14	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0

作業 5：
類似 sobel 濾波器的一階導數濾波器還有 Prewitt, Frei-Chen, Sharr 等，其濾波器分別如下：

- **Prewitt:**

$$\begin{matrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{matrix} \qquad \begin{matrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{matrix}$$

- **Frei Chenn:**

$$\begin{matrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{matrix} \qquad \begin{matrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{matrix}$$

implemented with ints and divisor of 100

- **Sharr:**

$$\begin{matrix} -3 & -10 & -3 \\ 0 & 0 & 0 \\ 3 & 10 & 3 \end{matrix} \qquad \begin{matrix} -3 & 0 & 3 \\ -10 & 0 & 10 \\ -3 & 0 & 3 \end{matrix}$$

請設計這 4 個濾波器(包含 Sobel)的程式，並用同一張影像比較 4 個濾波器的執行結果。

作業 6：

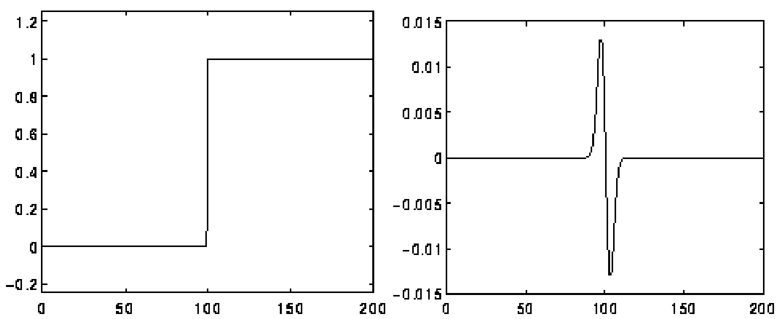
有兩種被稱為羅盤濾波器(compass filter)的一階導數濾波器，就是下圖的 Robinson 濾波器和 Kirsch 濾波器，它們都使用一個濾波器視窗旋轉 8 個角度得到 8 組濾波器係數，這 8 個濾波器視窗分別和影像作迴旋積，8 個迴旋積當中的最大值定義為 edge 的強度(amplitude)，edge 的方向(orientation)則是對應的濾波器視窗角度編號(0~7)，角度 0(即 r0 或 k0)代表垂直 edge，角度 3 代表水平 edge。

$r_0 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$ $r_1 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}$ $r_2 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$ $r_3 = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}$ $r_4 = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$ $r_5 = \begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix}$ $r_6 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$ $r_7 = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$	Robinson 濾波器
$k_0 = \begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix}$ $k_1 = \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix}$ $k_2 = \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}$ $k_3 = \begin{bmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}$ $k_4 = \begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix}$ $k_5 = \begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix}$ $k_6 = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{bmatrix}$ $k_7 = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & 5 \\ -3 & 5 & 5 \end{bmatrix}$	Kirsch 濾波器

請設計這 2 個濾波器的程式，輸入一張影像，輸出 amplitude 影像和 orientation 影像(使用角度編號 0~7，以圖形或資料表呈現均可)。

B.Laplacian 濾波器

拉普拉斯濾波器是一種空間二階導數的運算子，它對於影像中快速變化的區域(包含 edge)具有很大的強化作用，因此常結合 Zero Crossing Detection(如下圖)作為邊緣偵測的工具。



一維的二階導數：

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$

空間(二維)的二階導數：

Laplacian 濾波器 *L(x,y)* 定義為:

$$L(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

逼近的離散 Laplacian 濾波器有以下三種型態：

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

−1	2	−1
2	−4	2
−1	2	−1

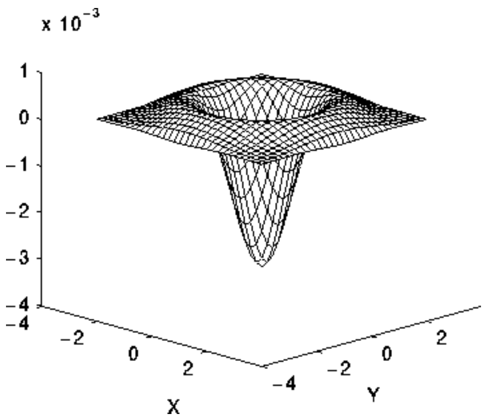
Laplacian 濾波器的使用也採標準的迴旋積方法。由於它對高頻訊號的高度敏感，同時也會強化高頻雜訊，所以應用拉普拉斯濾波器前常先進行平滑化濾波(如高斯平滑濾波)，以便降低雜訊干擾。但高斯平滑濾波+Laplacian 濾波需要雙

重迴旋積運算，十分耗費計算資源，因此多數採用 LoG (Laplacian of Gaussian)濾波器。

LoG (Laplacian of Gaussian)濾波器(zero mean 及標準差 σ) :

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right]e^{-\frac{x^2+y^2}{2\sigma^2}}$$

其函數曲面圖如下：



一個 9x9 離散濾波器係數例子如下(標準差 σ= 1.4)：

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

LoG 注意事項

The LoG operator calculates the second spatial derivative of an image. This means that in areas where the image has a constant intensity (*i.e.* where the intensity gradient is zero), the LoG response will be zero. In the vicinity of a change in intensity, however, the LoG response will be positive on the darker side, and negative on the lighter side. This means that at a reasonably sharp edge between two regions of uniform but different intensities, the LoG response will be:

- zero at a long distance from the edge,
- positive just to one side of the edge,
- negative just to the other side of the edge,
- zero at some point in between, on the edge itself.

作業 7：

設計一個 9x9 LoG (Laplacian of Gaussian)濾波器函式，對 ant(gray)600x400 影像進行 edge 增強濾波，(a)輸出 LoG 濾波後影像(請注意值域的 normalization)；(b)(option)輸出 zero-crossing 影像，也就是找出 LoG 濾波後影像的 zero-crossing 的 pixel 位置標示為 0，其他標示為 255。