Basic Signal and Image Processing Knowledge (3)

基礎信號與影像處理知識(三)

丁建均 教授

國立台灣大學電信工程學研究所

2019年8月15日

Image Processing 基礎知識 (2)

(1) Matlab 寫程式的原則

(2) Image Fusion by the Fourier Transform

(3) Morphology

(4) Equalizer for Images

(5) Principal Component Analysis (PCA) and Singular Value Decomposition (SVD)

(1) Matlab 寫程式的原則

一、各種程式語言寫程式共通的原則

(1) 能夠不在迴圈內做的運算，則移到迴圈外，以節省運算時間

(2) 寫一部分即測試，不要全部寫完再測試 (縮小範圍比較容易 debug)

(3) 先測試簡單的例子，成功後再測試複雜的例子

二、Matlab 寫程式特有的技巧

(1) 迴圈能避免就儘量避免

(2) 儘可能使用 Matrix 及 Vector operation

Example: 由 1 加 到100，用 Matlab 一行就可以了

```
sum([1:100])
```

完全不需迴圈

三、一些重要的 Matlab 指令

(1) function: 放在第一行，可以將整個程式函式化

(2) tic, toc: 計算時間

tic 為開始計時，toc 為顯示時間

(3) find: 找尋一個 vector 當中不等於 0 的entry 的位置

範例：find([1 0 0 1]) = [1, 4]

find(abs([-5:5])<=2) = [4, 5, 6, 7, 8]

(因為 abs([-5:5])<=2 = [0 0 0 1 1 1 1 1 0 0 0])

(4) ′ : Hermitian (transpose + conjugation)， .′ : transpose

(5) imread: 讀圖， image, imshow, imagesc: 將圖顯示出來，

(註： 較老的 Matlab 版本 imread 要和 double 並用

A=double(imread(‘Lena.bmp’));

(6) imwrite: 製做圖檔

(7) xlsread: 由 Excel 檔讀取資料

(8) xlswrite: 將資料寫成 Excel 檔

(9) aviread: 讀取 video 檔

(10) dlmread: 讀取 *.txt 或其他類型檔案的資料

(11) dlmwrite: 將資料寫成 *.txt 或其他類型檔案

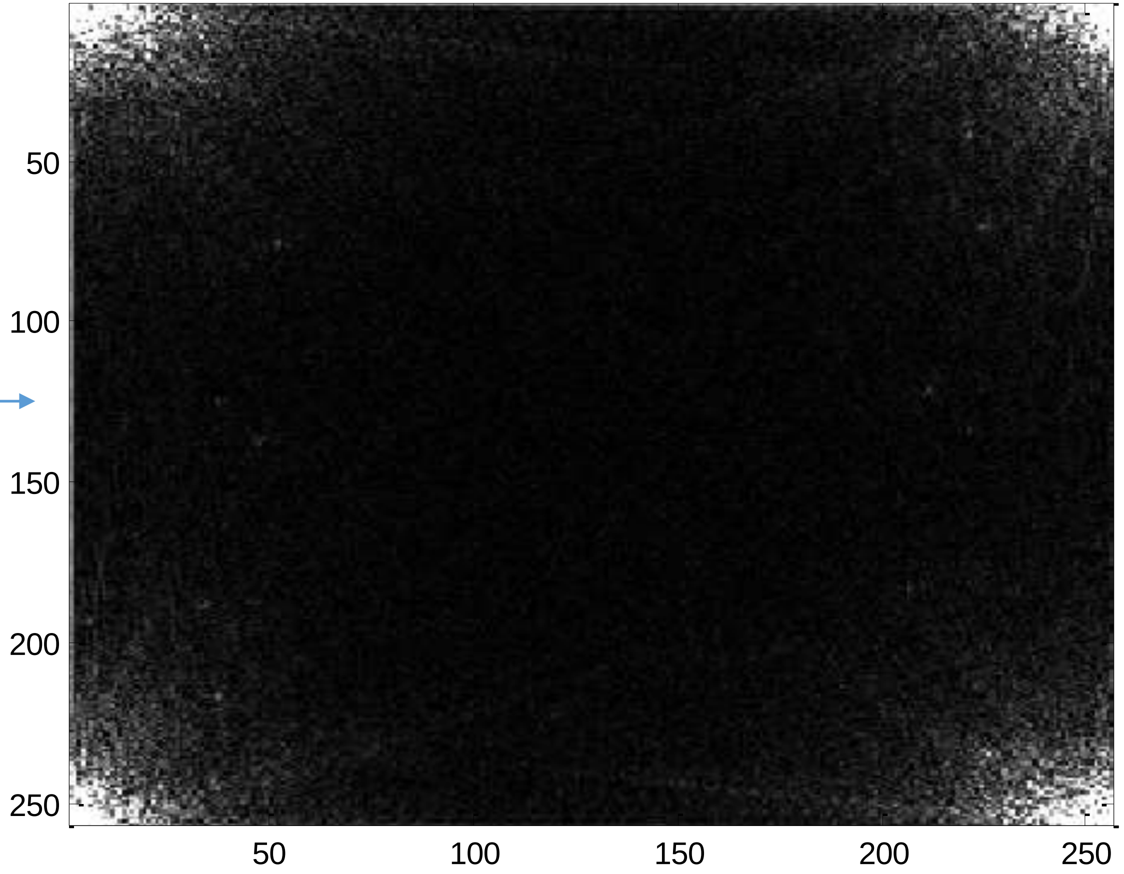
(2) Image Fusion by the Fourier Transform

2-D Discrete Fourier Transform (2-D DFT)

$$X[p,q]=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}x[m,n]e^{-j\frac{2\pi mp}{M}}e^{-j\frac{2\pi nq}{N}}$$

$$x[m,n]=\frac{1}{MN}\sum_{p=0}^{M-1}\sum_{q=0}^{N-1}X[p,q]e^{j\frac{2\pi mp}{M}}e^{j\frac{2\pi nq}{N}}$$

FT →



Using the gray level to show the intensity

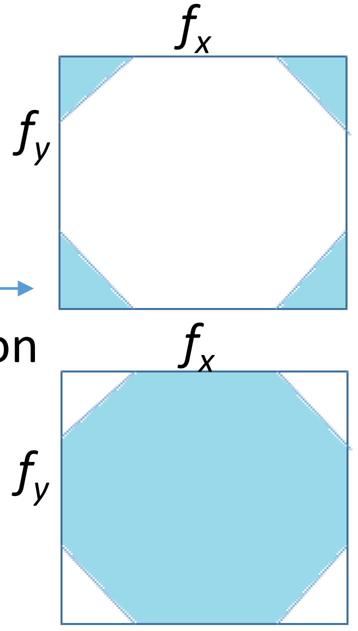
image(…….)
colormap(gray(256))

Low Frequency Part: Signal

High Frequency Part: Edge and Noise

Input Image → 2D Fourier Transform → Spectrum of the Input Image → multiplication →

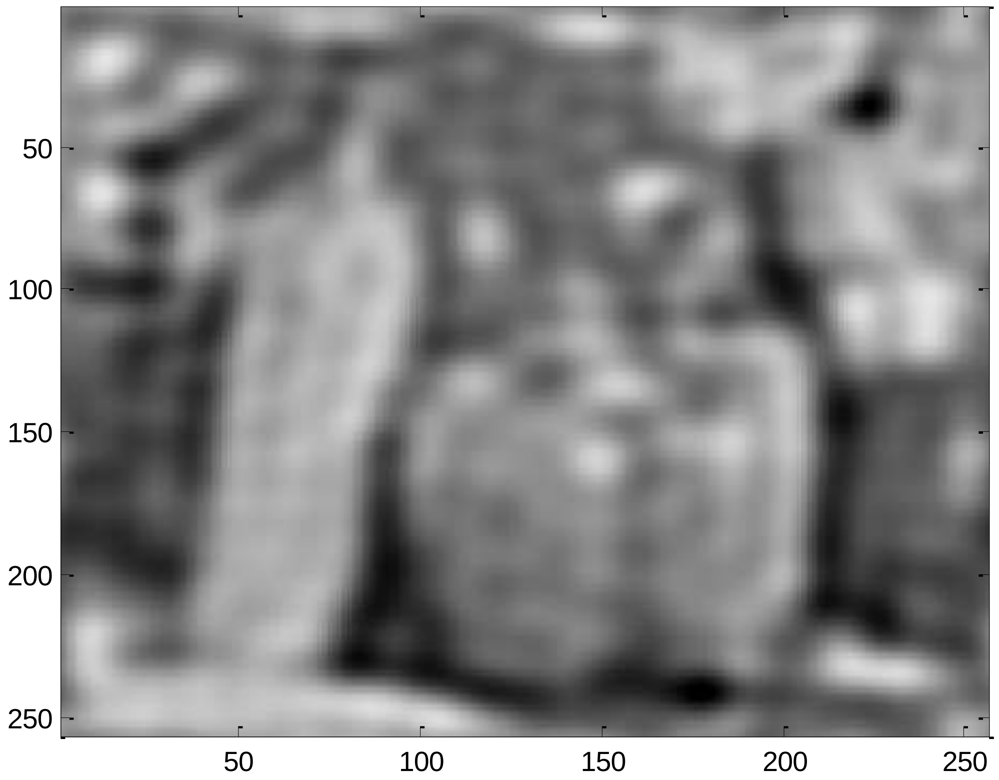
lowpass mask



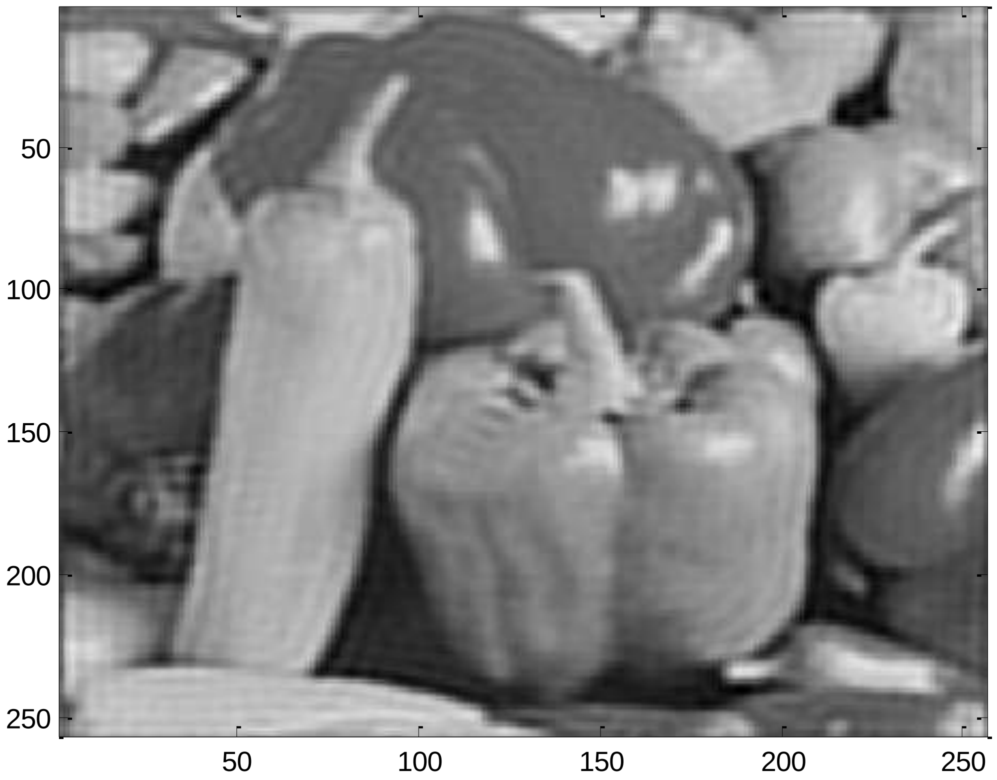
highpass mask

→ Inverse 2D Fourier Transform → Output Image

Lowpass output

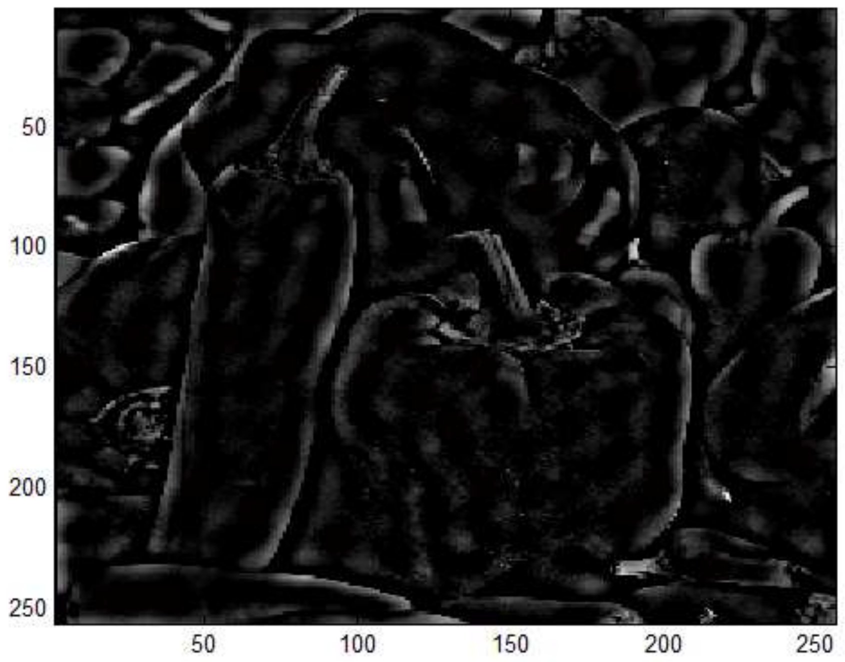


Passband: $|f_x| + |f_y| \leq N/30$

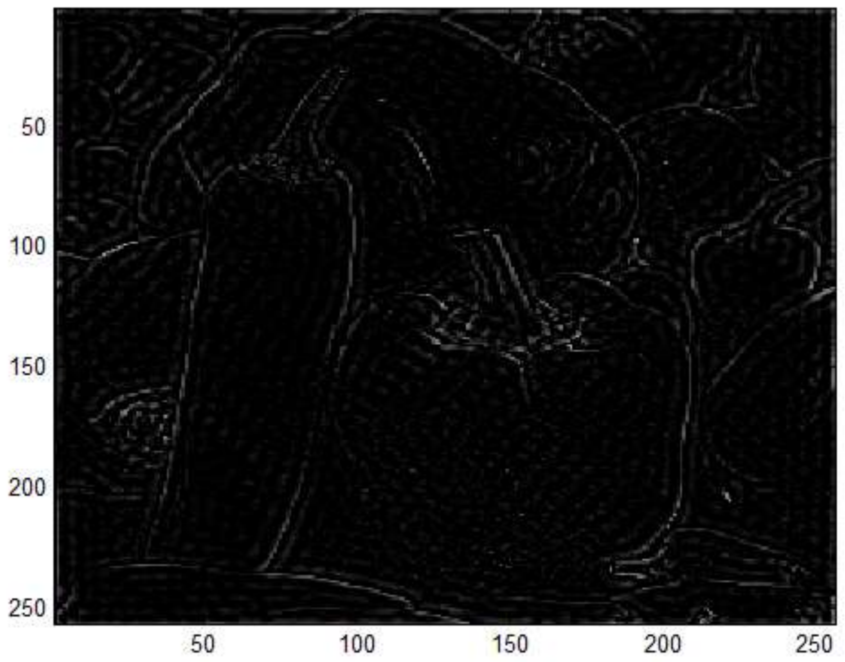


Passband: $|f_x| + |f_y| \leq N/10$

Highpass output

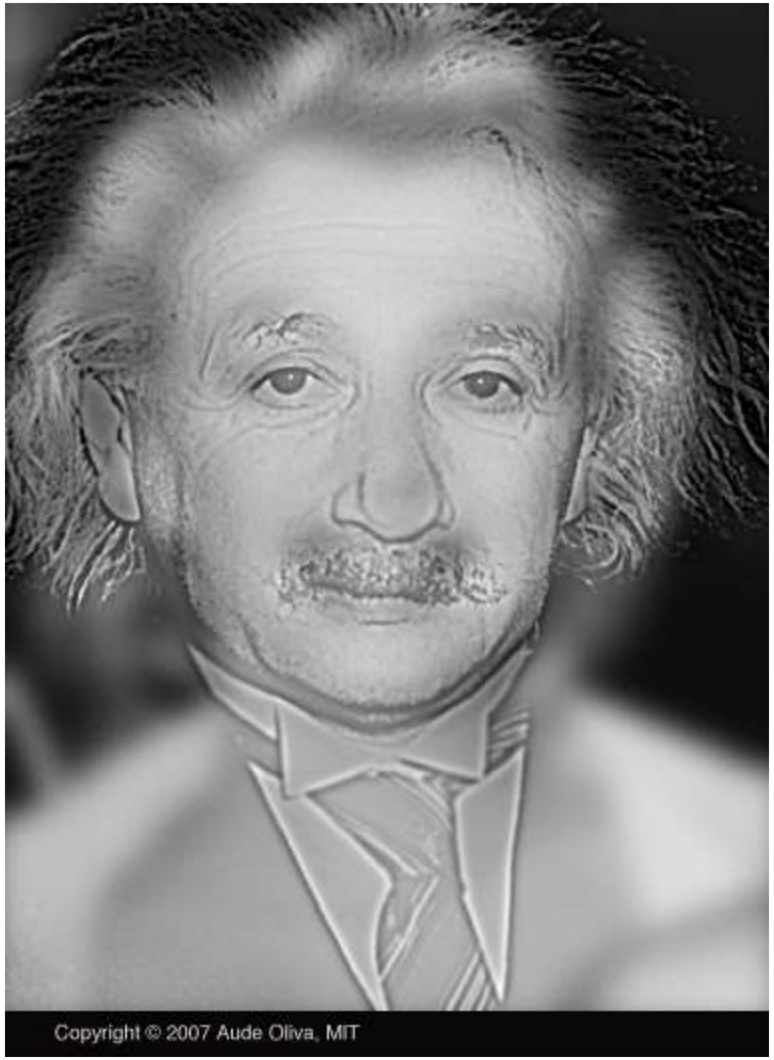


Passband: $|f_x| + |f_y| > N/30$



Passband: $|f_x| + |f_y| > N/10$

A 圖低頻
加 B 圖高頻



http://cvcl.mit.edu/hybrid_gallery/monroe_einstein.html

(3) Morphology

(3-1) Erosion

*	*	*	*	*	*	*
*	*	*	*	~~0~~	*	*
*	*	~~0~~	~~0~~	0	~~0~~	*
*	~~0~~	0	0	0	~~0~~	*
*	~~0~~	0	0	0	~~0~~	*
*	~~0~~	0	0	0	~~0~~	*
*	*	~~0~~	0	~~0~~	*	*
*	*	*	~~0~~	*	*	*
*	*	*	*	*	*	*

*	*	*	*	*	*	*
*	*	*	*	*	*	*
*	*	*	*	0	*	*
*	*	0	0	0	*	*
*	*	0	0	0	*	*
*	*	0	0	0	*	*
*	*	*	0	*	*	*
*	*	*	*	*	*	*
*	*	*	*	*	*	*

$$A[m,n] = A[m,n] \,\&\, A[m-1,n] \,\&\, A[m+1,n] \,\&\, A[m,n-1] \,\&\, A[m,n+1]$$

(3-2) Dilation

*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*
*	*	*	*	O	*	*	*
*	*	*	O	O	O	O	*
*	*	O	O	O	O	*	*
*	*	*	*	O	O	O	*
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*

*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*
*	*	*	*	O	*	*	*
*	*	*	O	O	O	O	*
*	*	O	O	O	O	O	O
*	O	O	O	O	O	O	*
*	*	O	O	O	O	O	O
*	*	*	*	O	O	O	*
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*

$$A[m,n] = A[m,n] \mid A[m-1,n] \mid A[m+1,n] \mid A[m,n-1] \mid A[m,n+1]$$

Erosion for a Non-binary Image

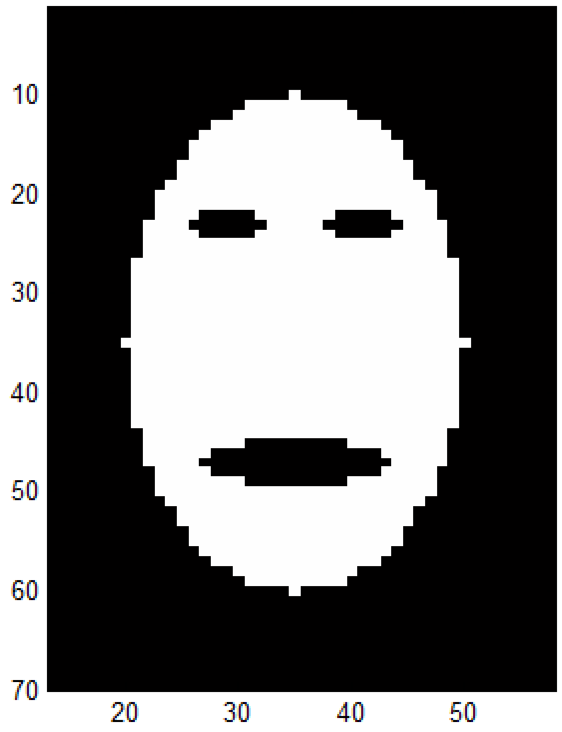
$$A[m,n] = \min\{A[m,n],\, A[m-1,n],\, A[m+1,n],\, A[m,n-1],\, A[m,n+1]\}$$

Dilation for a Non-binary Image

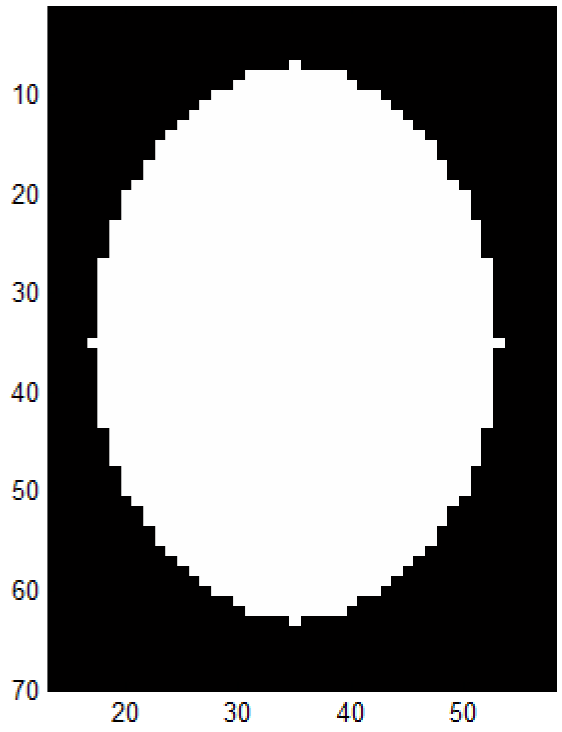
$$A[m,n] = Max\{A[m,n],\, A[m-1,n],\, A[m+1,n],\, A[m,n-1],\, A[m,n+1]\}$$

(3-3) Closing (Hole Filling)

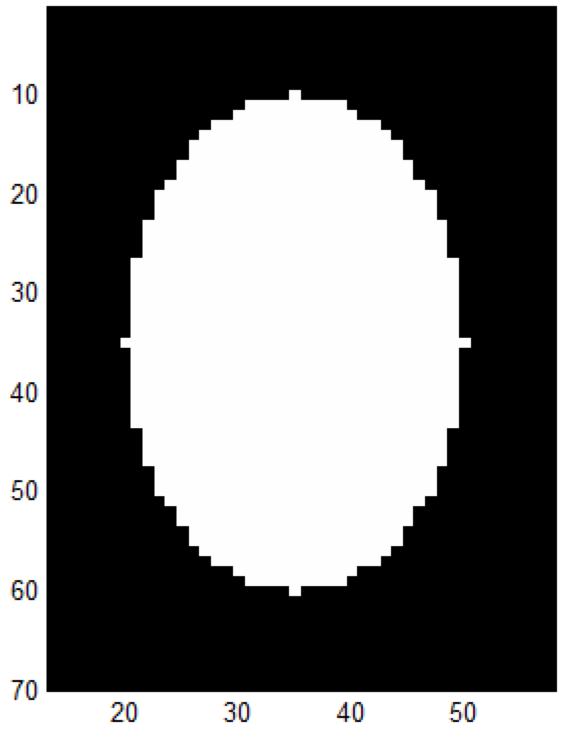
closing = dilation *k* times + erosion k times



dilation 3 times

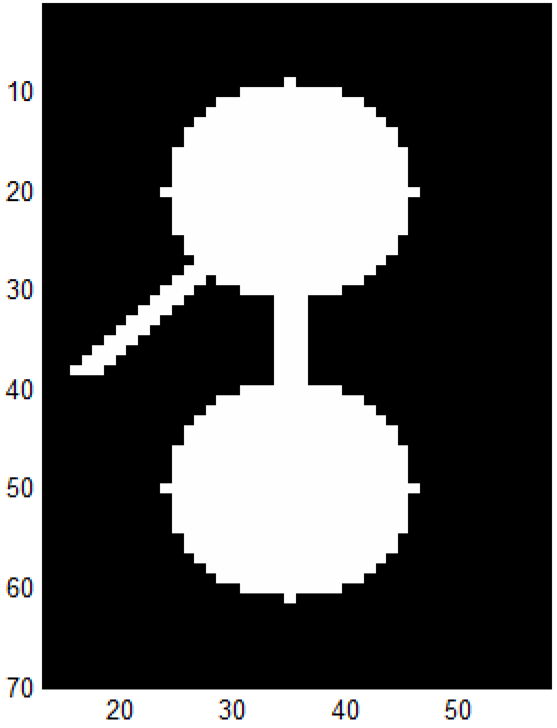


then erosion 3 times

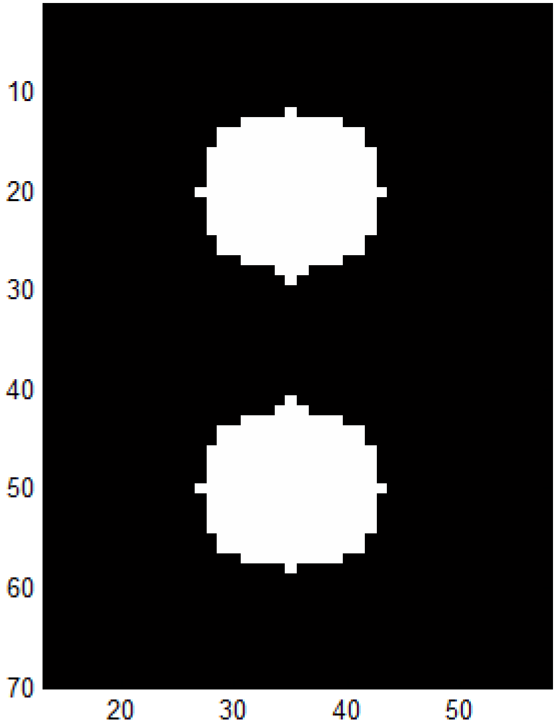


(3-4) Opening

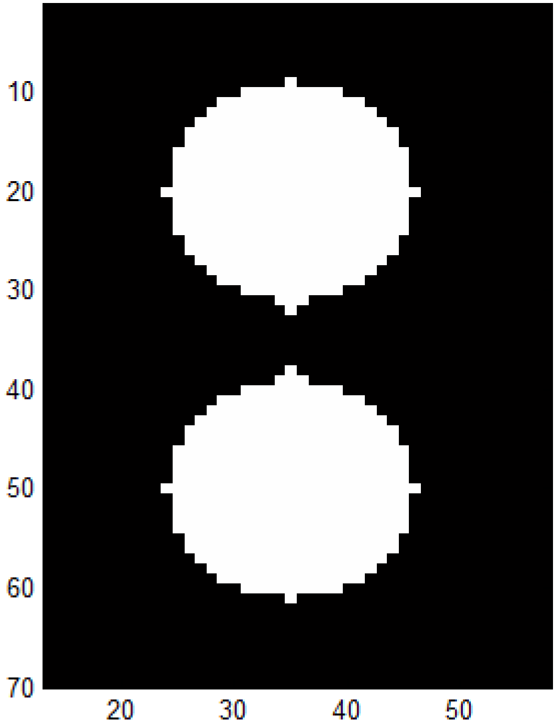
opening = erosion *k* times + dilation k times



erosion 3 times



then dilation 3 times



(4) Equalizer for Images

$$y[m,n] = x[m,n] * k[m,n]$$

Equalizer: $H(F_x, F_y) = \dfrac{1}{K(F_x, F_y)}$

Problem: If the system is interfered by noise $noise[m, n]$

$$y[m,n] = x[m,n] * k[m,n] + noise[m,n]$$

$$Y(F_x, F_y) = X(F_x, F_y) K(F_x, F_y) + Noise(F_x, F_y)$$

$$H(F_x, F_y) Y(F_x, F_y) = X(F_x, F_y) H(F_x, F_y) K(F_x, F_y) + H(F_x, F_y) Noise(F_x, F_y)$$

$$= X(F_x, F_y) + \frac{Noise(F_x, F_y)}{K(F_x, F_y)}$$

If $K(F_x, F_y)$ is near to 0, the effect of the noise is magnified.

Combined with the concept of the Wiener filter, the equalizer is modified as:

$$H(F_x,F_y)=\frac{1}{\dfrac{1}{K^*(F_x,F_y)}\dfrac{E\left(\left|Noise(F_x,F_y)\right|^2\right)}{E\left(\left|X(F_x,F_y)\right|^2\right)}+K(F_x,F_y)}$$

E: mean

$$H(F_x,F_y)=\frac{1}{\dfrac{C}{K^*(F_x,F_y)}+K(F_x,F_y)}$$

C is large when the SNR is small
C is small when the SNR is large

Implementation

$$X(F_x, F_y) = fft2\left(x[m,n]\right)$$

$$K(F_x, F_y) = fft2\left(k_1[m,n]\right)$$

$k_1(m,n) = k(m,n)$ if $0 \le m \le L$, $0 \le n \le L$

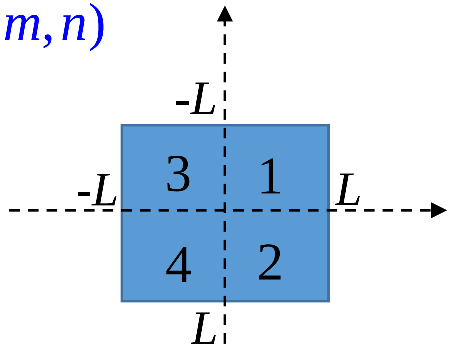
$k_1(M+m,n) = k(m,n)$ if $-L \le m < 0$, $0 \le n \le L$

$k_1(m,N+n) = k(m,n)$ if $0 \le m \le L$, $-L \le n < 0$

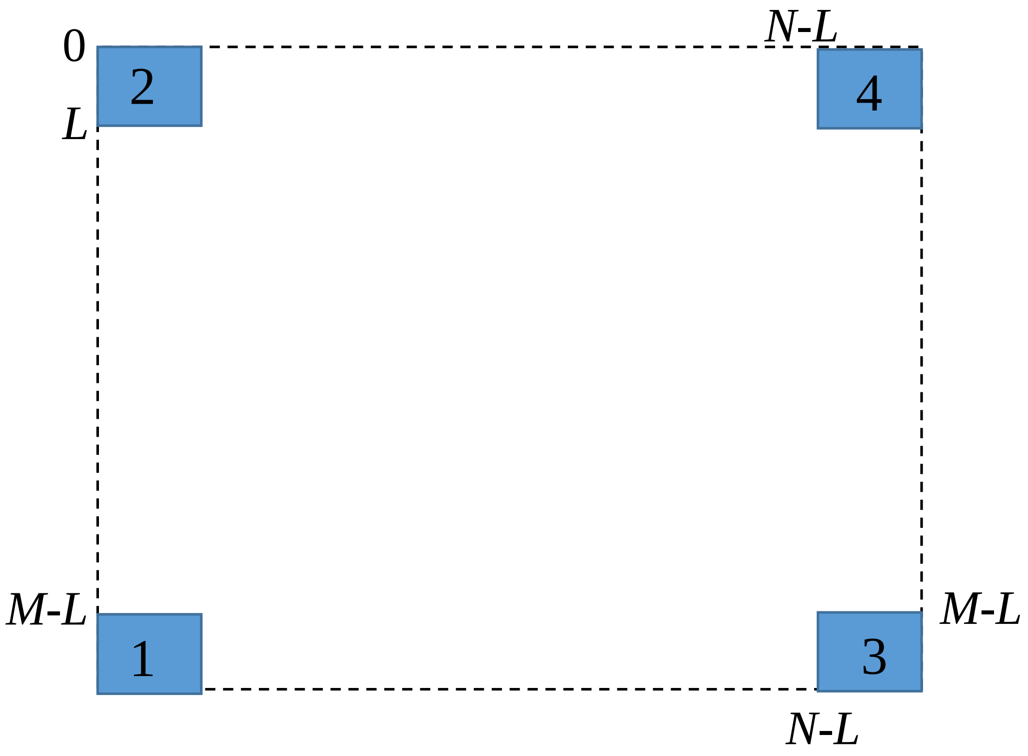
$k_1(M+m,N+n) = k(m,n)$ if $-L \le m < 0$, $-L \le n < 0$

$k_1(M+m,N+n) = 0$ otherwise

$k(m,n)$



$k_1(m,n)$



(5) PCA and SVD

PCA (principal component analysis) 是資料分析和影像處理當中常用到的數學方法，用來分析資料的「主要成分」或是影像中物體的「主軸」。

它其實和各位同學在高中和大一線代所學的回歸線 (regressive line) 很類似。回歸線是用一條一維 (one-dimensional) 的直線來近似二維 (two-dimensional) 的資料，而 PCA 則是用 M-dimensional data 來近似 N-dimensional data ，其中 M 小於等於 N

在講解 PCA 之前，先介紹什麼是 SVD (singular value decomposition)

我們在大一的時候，都已經學到該如何對於 N x N 的矩陣做 eigenvector -eigenvalue decomposition
那麼………

<u>當一個矩陣的 size 為 M x N 且 M 和 N 不相等時，我們該如何對它來做 eigenvector-eigenvalue decomposition?</u>

SVD 的流程：

假設 **A** 是一個 *M* x *N* 的矩陣。

(Step 1) 計算

如果只是要找 principal axis
C 和 U 的計算可省略

$$\mathbf{B} = \mathbf{A}^H\mathbf{A} \qquad \mathbf{C} = \mathbf{A}\mathbf{A}^H$$

注意，**B** 是 *N* x *N* 的矩陣，而 **C** 是 *M* x *M* 的矩陣。上標H代表Hermitian matrix，相當於做共軛轉置。

(Step 2) 接著，對 **B** 和 **C** 做 eigenvector-eigenvalue decomposition

$$\mathbf{B} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1} \qquad \mathbf{C} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$$

其中 **V** 的每一個 column 是 **B** 的 eigenvector (with normalization)， **U** 的每一個 column 是 **C** 的 eigenvector (with normalization) ， **Λ** 和 **D** 都是對角矩陣， **Λ** 和 **D** 對角線上的 entries 是 **B** 和 **C** 的 eigenvalues。並假設 eigenvectors 根據 eigenvalues 的大小排序 (由大到小)

Note: 值得注意的是，由於 $\mathbf{B} = \mathbf{B}^H$ 且 $\mathbf{C} = \mathbf{C}^H$，所以 **B** 和 **C** 的 eigenvectors 皆各自形成一個 orthogonal set。經過適當的 normalization 使得 **U** 和 **V** 的 column 自己和自己的內積為 1 之後， $\mathbf{U}^{-1} = \mathbf{U}^H$ 和 $\mathbf{V}^{-1} = \mathbf{V}^H$ 將滿足。因此，**B** 和 **C** 可以表示成

$$\mathbf{B} = \mathbf{V}\mathbf{D}\mathbf{V}^H \qquad \mathbf{C} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H$$

注意，**V**和**U**是unitary matrix

(Step 3) 計算

$\mathbf{S_1} = \mathbf{U^H A V}$ $\mathbf{S} = |\mathbf{S_1}|$ 取絕對值

S 是一個 $M \times N$ 的矩陣，只有在 **S**[n, n] (n = 1, 2, ..., min(M, N)) 的地方不為 0

(Step 4) 若 $\boldsymbol{S_1}[n, n] < 0$，改變 **U** 第 n 個 column 的正負號

即完成 SVD

$$\mathbf{A} = \mathbf{U S V^H}$$

A 也可以表示為

$$\mathbf{A} = \lambda_1 \mathbf{u_1 v_1^T} + \lambda_2 \mathbf{u_2 v_2^T} + \cdots\cdots + \lambda_k \mathbf{u_k v_k^T}$$

其中 $\lambda_n = \boldsymbol{S}[n, n]$, $k = \min(M, N)$)

註： Matlab 有內建的 svd 指令可以計算 SVD

從 SVD 到 PCA (principal component analysis ，主成份分析)

$$\mathbf{A} = \lambda_1 \mathbf{u}_1 \mathbf{v}_1^{\mathbf{T}} + \lambda_2 \mathbf{u}_2 \mathbf{v}_2^{\mathbf{T}} + \cdots\cdots + \lambda_k \mathbf{u}_\mathbf{k} \mathbf{v}_\mathbf{k}^{\mathbf{T}}$$

$k = \min(M, N))$

若 $\lambda_1 \geqq \lambda_2 \geqq \lambda_3 \geqq \ldots.. \geqq \lambda_k$

$\lambda_1 \mathbf{u}_1 \mathbf{v}_1^{\mathbf{T}}$ 是 A 矩陣的最主要的成份

$\lambda_2 \mathbf{u}_2 \mathbf{v}_2^{\mathbf{T}}$ 是 A 矩陣的第二主要的成份

:

$\lambda_k \mathbf{u}_\mathbf{k} \mathbf{v}_\mathbf{k}^{\mathbf{T}}$ 是 A 矩陣的最不重要的成份

若為了壓縮或是去除雜訊的考量，可以選擇 $h < k$，使得 **A** 可以近似成

$$\mathbf{A} \cong \lambda_1 \mathbf{u}_1 \mathbf{v}_1^{\mathbf{T}} + \lambda_2 \mathbf{u}_2 \mathbf{v}_2^{\mathbf{T}} + \cdots\cdots + \lambda_h \mathbf{u}_\mathbf{h} \mathbf{v}_\mathbf{h}^{\mathbf{T}}$$

PCA 的流程

假設現在有 M 筆資料，每一筆資料 為 N dimension

$\mathbf{g_1} = [f_{1,1}\ \ f_{1,2}, \ldots, f_{1,N}]$

$\mathbf{g_2} = [f_{2,1}\ \ f_{2,2}, \ldots, f_{2,N}]$

$\vdots$

$\mathbf{g_M} = [f_{M,1}\ \ f_{M,2}, \ldots, f_{M,N}]$

(Step 1) 扣掉平均值，形成新的 data

$$\mathbf{d_m} = \left[e_{m,1}\ \ e_{m,2}\ \ \cdots\cdots\ e_{m,N} \right] \qquad m = 1, 2, \ldots, M$$

其中 $e_{m,n} = f_{m,n} - \tilde{f}_n$, $\qquad \tilde{f}_n = \dfrac{1}{M}\sum_{m=1}^{M} f_{m,n}$

(Step 2) 形成 M x N 的矩陣 **A**

A 的第 m 個 row 為 d_m , $\qquad m = 1, 2, \ldots, M$

(Step 3) 對 A 做 SVD 分解

$$\mathbf{A} = \mathbf{USV}^{\mathbf{H}}$$

$$= \lambda_1 \mathbf{u_1 v_1^T} + \lambda_2 \mathbf{u_2 v_2^T} + \cdots\cdots + \lambda_k \mathbf{u_k v_k^T} \qquad k = \min(M, N)$$

$$\lambda_1 \geqq \lambda_2 \geqq \lambda_3 \geqq \ldots\ldots \geqq \lambda_k$$

(Step 4) 將 A 近似成

$$\mathbf{A} \cong \lambda_1 \mathbf{u_1 v_1^T} + \lambda_2 \mathbf{u_2 v_2^T} + \cdots\cdots + \lambda_h \mathbf{u_h v_h^T}$$

則每一筆資料可以近似為

$$g_{\mathbf{m}} \cong \lambda_1 u_1[m]\mathbf{v_1^T} + \lambda_2 u_2[m]\mathbf{v_2^T} + \cdots\cdots + \lambda_h u_h[m]\mathbf{v_h^T} + \begin{bmatrix} \tilde{f}_1 & \tilde{f}_2 & \cdots & \tilde{f}_N \end{bmatrix}$$

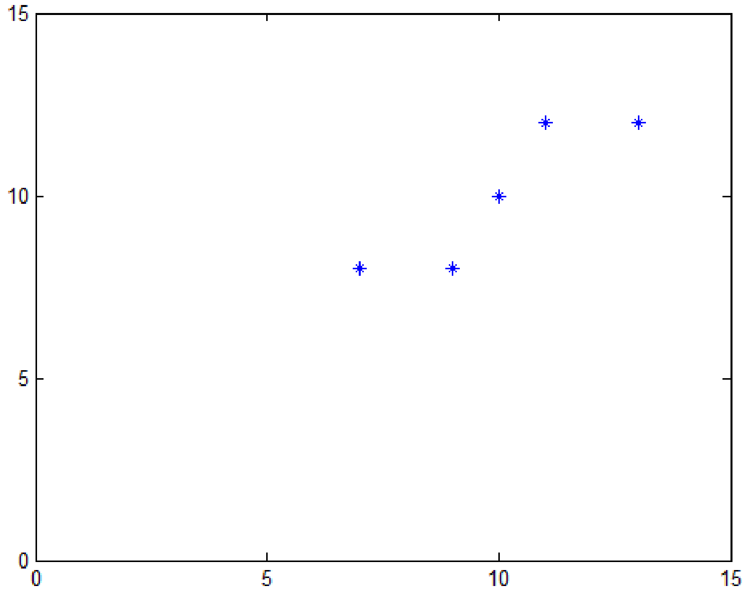
除了平均值 $\begin{bmatrix} \tilde{f}_1 & \tilde{f}_2 & \cdots & \tilde{f}_N \end{bmatrix}$ 之外

$\mathbf{v_1}^{\mathbf{T}}$ 是資料的最主要成分， $\mathbf{v_2}^{\mathbf{T}}$ 是資料的次主要成分，
$\mathbf{v_3}^{\mathbf{T}}$ 是資料的第三主要成分，以此類推

PCA 的例子

假設在一個二維的空間中，有5個點，座標分別是

(7,8), (9,8), (10, 10), (11,12), (13,12)



M = 5, N = 2

試求這五個點的 PCA (即回歸線)

(Step 1) 將這五個座標點減去平均值 (10, 10)

(-3, -2), (-1 -2), (0, 0), (1, 2), (3, 2)

(Step 2) 形成 5x2 的 matrix

$$\mathbf{A} = \begin{bmatrix} -3 & -2 \\ -1 & -2 \\ 0 & 0 \\ 1 & 2 \\ 3 & 2 \end{bmatrix}$$

(Step 3) 計算 SVD

$$\mathbf{A} = \mathbf{USV}^{\mathbf{H}}$$ From page 39

$$\mathbf{U} = \begin{bmatrix} -0.6116 & 0.3549 & 0 & 0.0393 & 0.7060 \\ -0.3549 & -0.6116 & 0 & 0.7060 & -0.0393 \\ 0 & 0 & 1 & 0 & 0 \\ 0.3549 & 0.6116 & 0 & 0.7060 & -0.0393 \\ 0.6116 & -0.3549 & 0 & 0.0393 & 0.7060 \end{bmatrix} \qquad \mathbf{S} = \begin{bmatrix} 5.8416 & 0 \\ 0 & 1.3695 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{V} = \begin{bmatrix} 0.7497 & -0.6618 \\ 0.6618 & 0.7497 \end{bmatrix}$$

主成分

次要成分

$$\mathbf{A} = 5.8416 \begin{bmatrix} -0.6116 \\ -0.3549 \\ 0 \\ 0.3549 \\ 0.6116 \end{bmatrix} \begin{bmatrix} 0.7497 & 0.6618 \end{bmatrix} + 1.3695 \begin{bmatrix} 0.3549 \\ -0.6116 \\ 0 \\ 0.6116 \\ -0.3549 \end{bmatrix} \begin{bmatrix} -0.6618 & 0.7497 \end{bmatrix}$$

(Step 4) 得到主成分 $[0.7497 \quad 0.6618]$

這五個座標點可以近似成

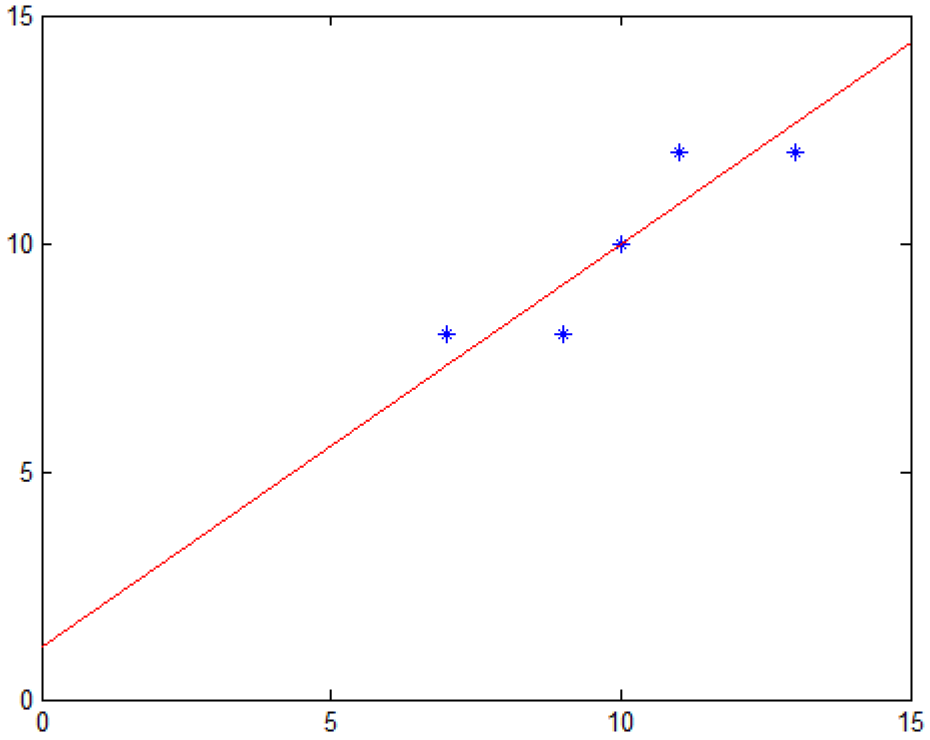
$$5.8416 \cdot u_m [0.7497 \quad 0.6618] + [10 \quad 10] \qquad m = 1, 2, \ldots, 5$$

$u_1 = -0.6116$, $u_2 = -0.3549$, $u_3 = 0$, $u_4 = 0.3549$, $u_5 = 0.6116$

回歸線

$$[10 \quad 10] + c[0.7497 \quad 0.6618]$$

$$c \in (-\infty, \infty)$$



練習

(1) 選兩張大小一樣的圖，用 Fourier Transform 將一張圖的低頻成份和另一張圖的高頻成份複合起來

(2) 寫出對一個圖型做 erosion 三次， dilation 三次，opening (其中 erosion 和 dilation 皆三次)，closing (其中 erosion 和 dilation 皆三次) 的程式

(3) (a) 產生 blurred image

$$y[m, n] = x[m, n] * k[m, n] + \text{noise}[m, n]$$

其中 $k[m, n] = s \exp[-0.1*(m\text{^}2+n\text{^}2)]$, $-10 <= m, n <= 10$

noise 可由 rand 產生

(b) 用 equalizer 來由 $y[m, n]$ 還原 $x[m, n]$

試用二個不同的 C 值，和二個不同的雜訊大小

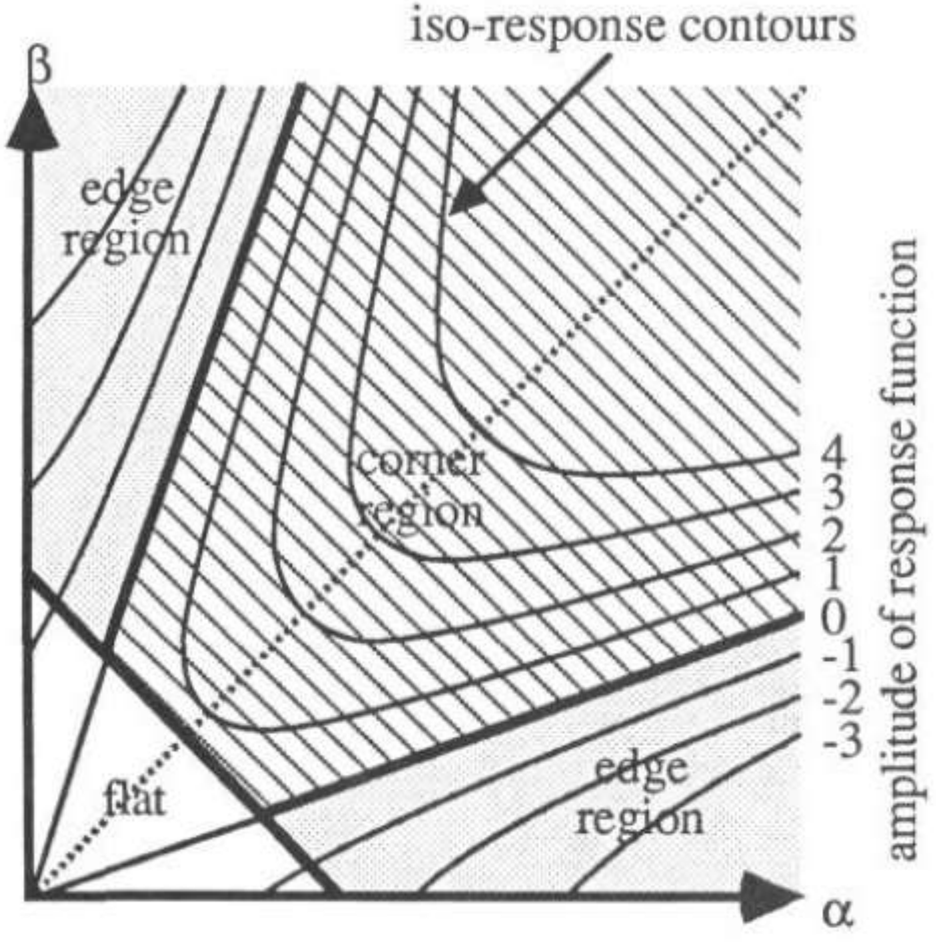
$$s = \frac{1}{\sum_{m=-10}^{10}\sum_{n=-10}^{10} e^{-0.1\left(m^2+n^2\right)}}$$

(4) 對以下的資料做 PCA

(2, -1, 3), (-1, 3, 5), (0, 2, 4), (4, -2, -1), (1, 0, 4), (-2, 5, 5)

(5) 運用 Google 學術搜尋，找出以下的論文並閱讀，並畫出的流程圖

C. Harris and M. Stephens, “A combined corner and edge detector”, Proc. 4th Alvey Vision Conf., 1988, pp. 147-151.



α, β:
the variation along two axes