XIV. Discrete Wavelet Transform

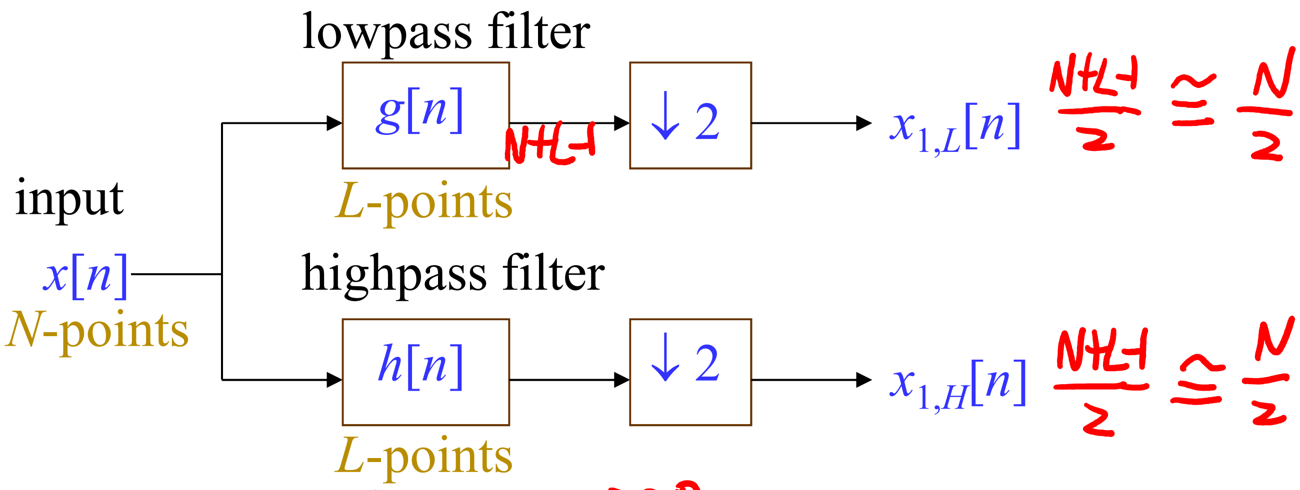
14.1 概念

(1) discrete input to discrete output

(2) 由 continuous wavelet transform with discrete coefficients 演變而來的，
(比較 page 387)
但是大幅簡化了其中的數學

(3) 忽略了 scaling function 和 mother wavelet function 的分析
但是保留了階層式的架構

14.2 1-D Discrete Wavelet Transform

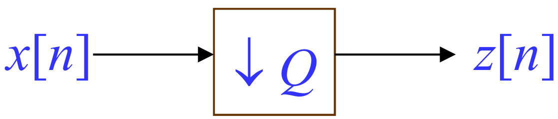


total length
$N+L-1 \cong N$

compared to page 390
$g_{-n} \to g[n]$
$h_{-n} \to h[n]$

$\boxed{\downarrow 2}$: downsampling by the factor of 2

5 times of DFT
$\Theta(N\log_2 N)$
$5N\log_2 N$
$\downarrow$
$3N\log_2 N$



$$z[n] = x[Qn]$$

輸入： $x[n]$ (不需算 $\chi_w(n,m)\big|_{m\to\infty}$，

直接以 $x[n]$ 作為 initial

Low pass filter $g[n]$

角色似 scaling function

(相當於 page 387 的 g_n)

High pass filter $h[n]$

角色似 wavelet function

(相當於 page 387 的 h_n)

1st stage

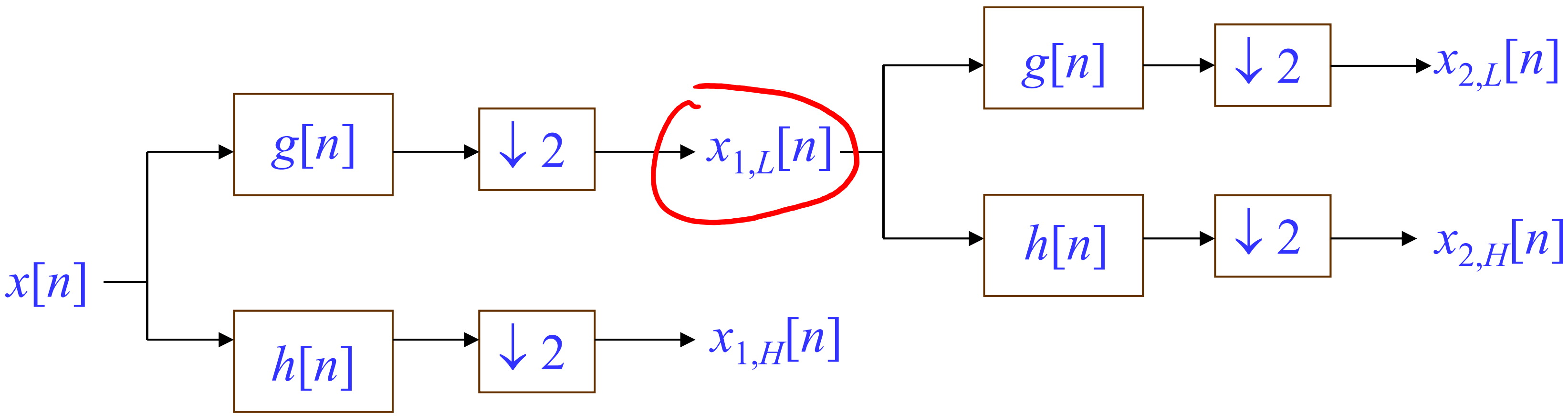
$$x_{1,L}[n]=\sum_{k=0}^{K-1}x[2n-k]g[k]$$

$$x_{1,H}[n]=\sum_{k=0}^{K-1}x[2n-k]h[k]$$

further decomposition (from the $(a-1)^{\text{th}}$ stage to the a^{th} stage)

$$x_{a,L}[n] = \sum_{k=0}^{K-1} x_{a-1,L}[2n-k]\,g[k]$$

$$x_{a,H}[n] = \sum_{k=0}^{K-1} x_{a-1,L}[2n-k]\,h[k]$$



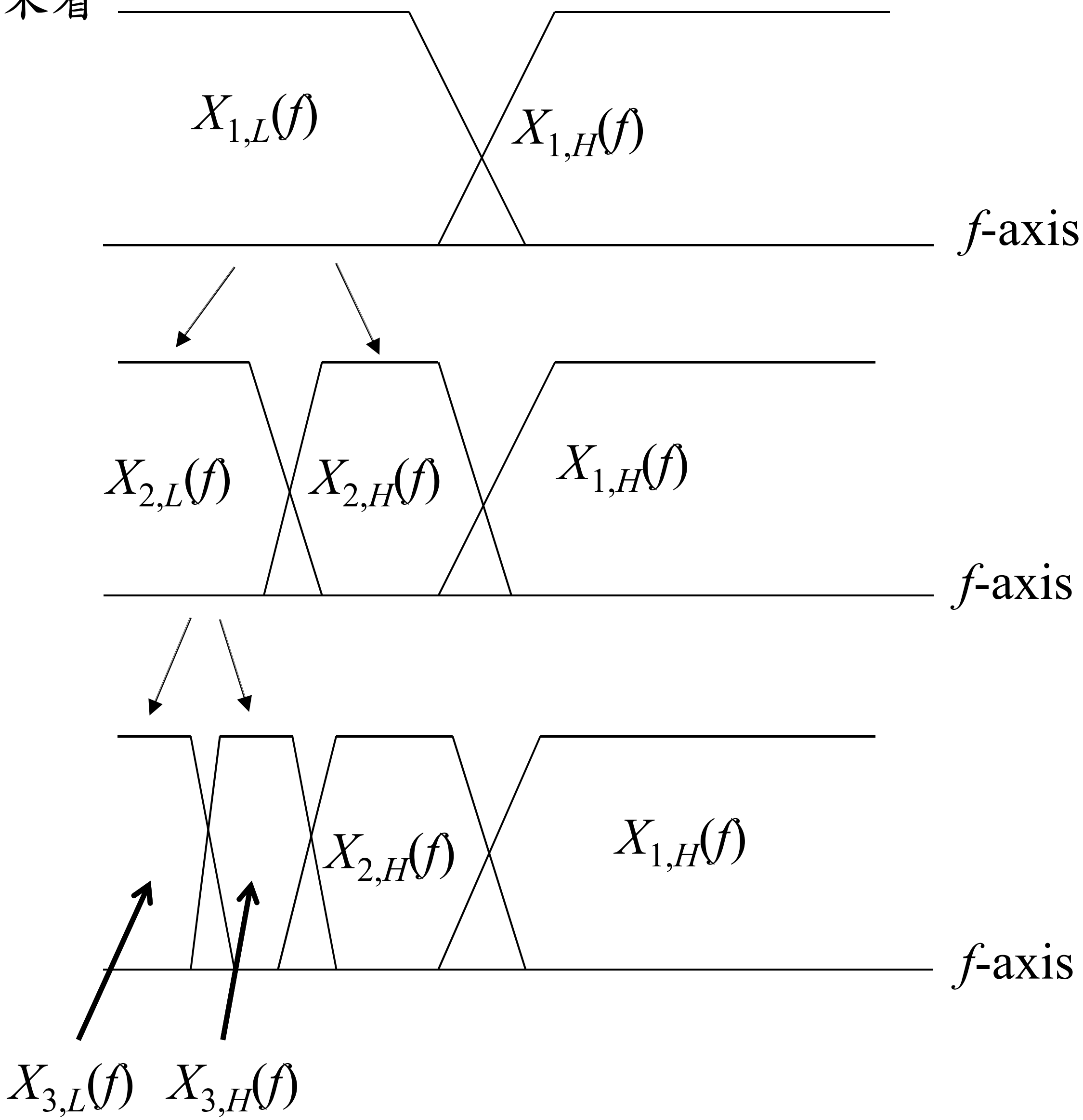
(1) 有的時候，對於 $x_{a,H}[n]$ 也再作細分

(2) 若 input 的 $x[n]$ 的 length 為 N,

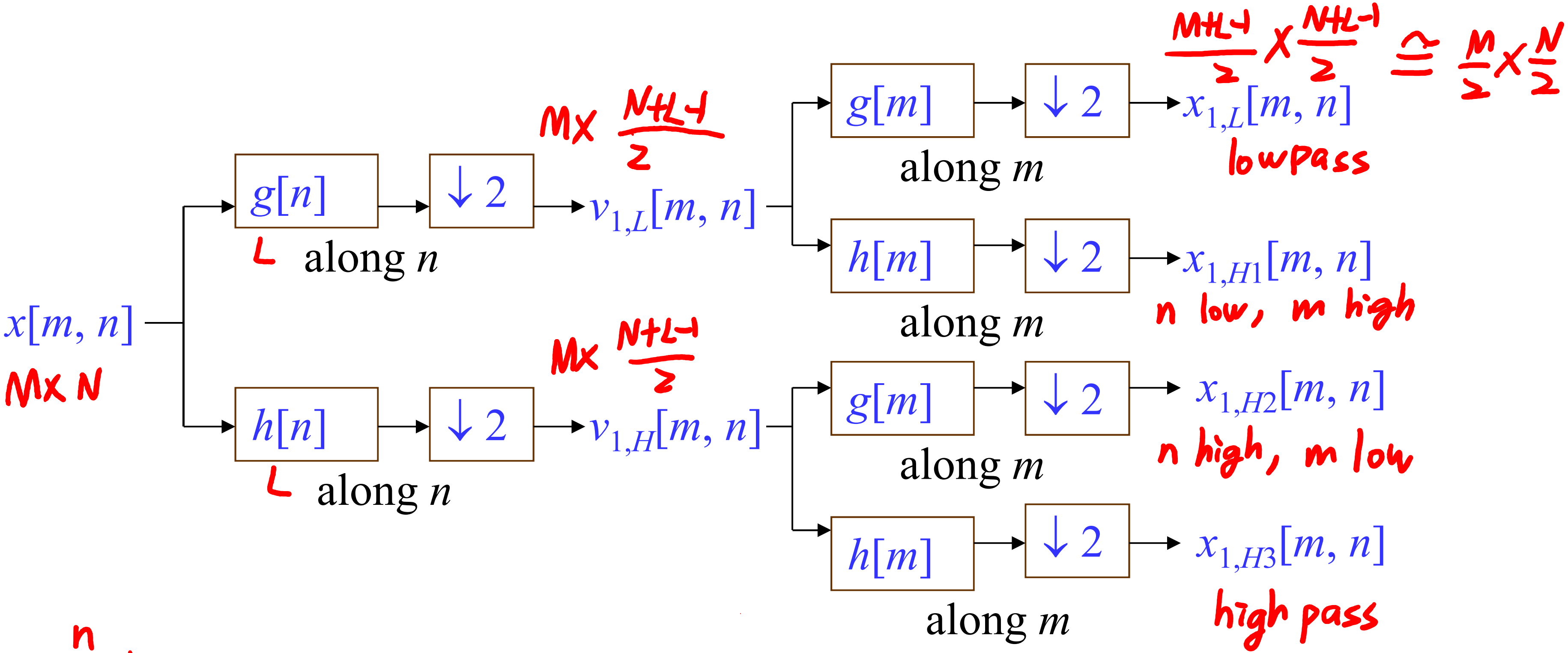
則 a^{th} stage $x_{a,L}[n]$, $x_{a,H}[n]$ 的 length 為 $N/2^a$

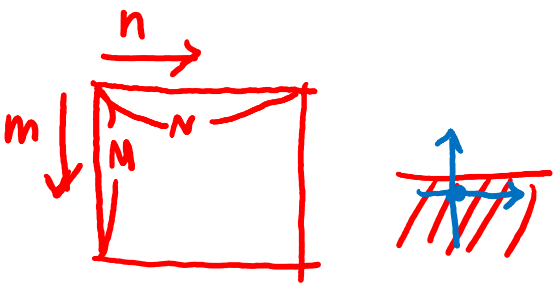
(3) 經過 DWT 之後，全部點數仍接近 N 點

(4) 以頻譜來看


$X_{1,L}(f)$
$X_{1,H}(f)$
f-axis
$X_{2,L}(f)$
$X_{2,H}(f)$
$X_{1,H}(f)$
f-axis
$X_{2,H}(f)$
$X_{1,H}(f)$
f-axis
$X_{3,L}(f)$
$X_{3,H}(f)$

14.3 2-D Discrete Wavelet Transform





輸入： $x[m, n]$

Low pass filter $g[n]$ High pass filter $h[n]$

- along n

$$v_{1,L}[m,n]=\sum_{k=0}^{K-1}x[m,2n-k]g[k]$$

$$v_{1,H}[m,n]=\sum_{k=0}^{K-1}x[m,2n-k]h[k]$$

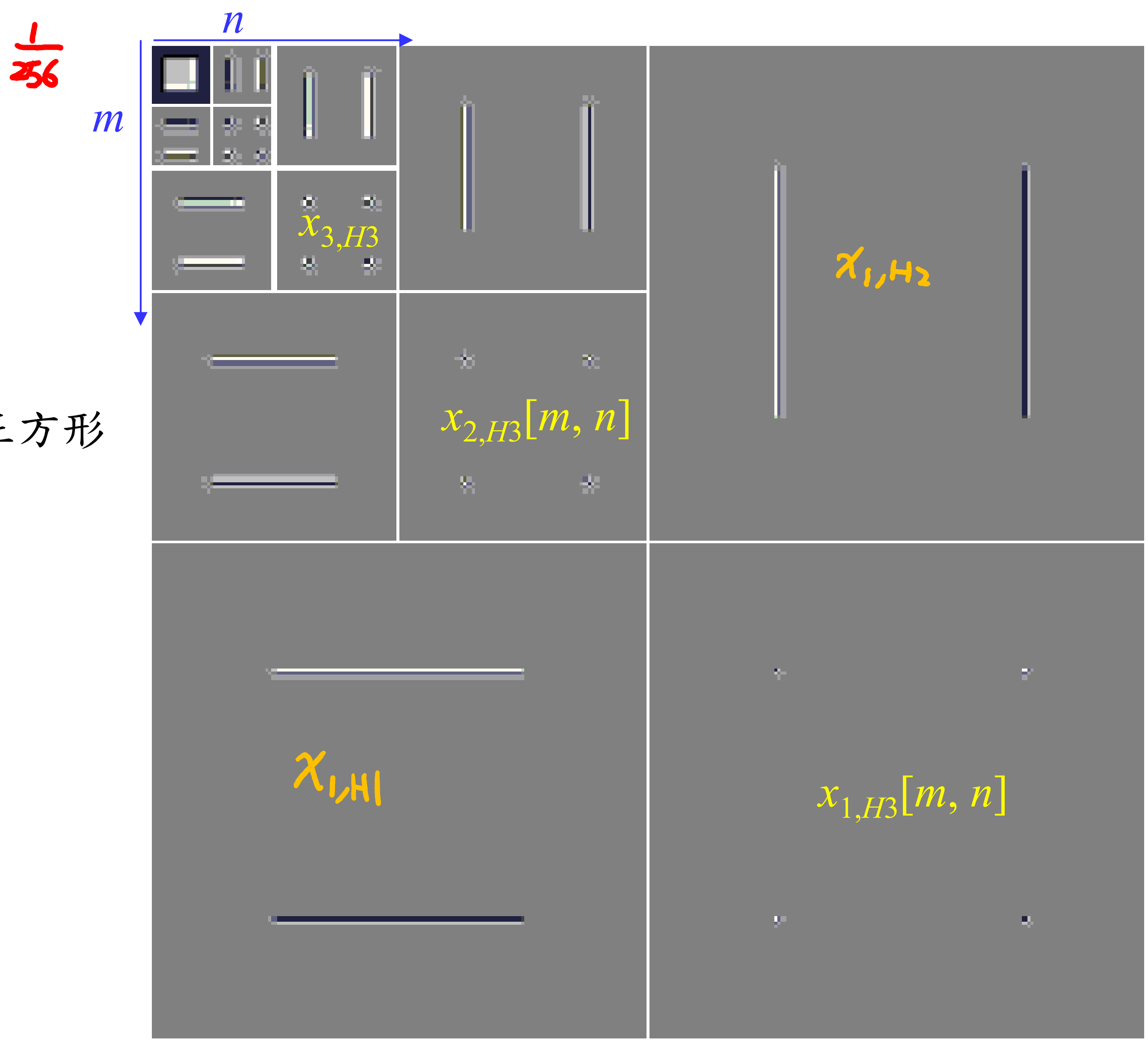
- along m

$$x_{1,L}[m,n]=\sum_{k=0}^{K-1}v_{1,L}[2m-k,n]g[k]$$

$$x_{1,H_2}[m,n]=\sum_{k=0}^{K-1}v_{1,H}[2m-k,n]g[k]$$

$$x_{1,H_1}[m,n]=\sum_{k=0}^{K-1}v_{1,L}[2m-k,n]h[k]$$

$$x_{1,H_3}[m,n]=\sum_{k=0}^{K-1}v_{1,H}[2m-k,n]h[k]$$

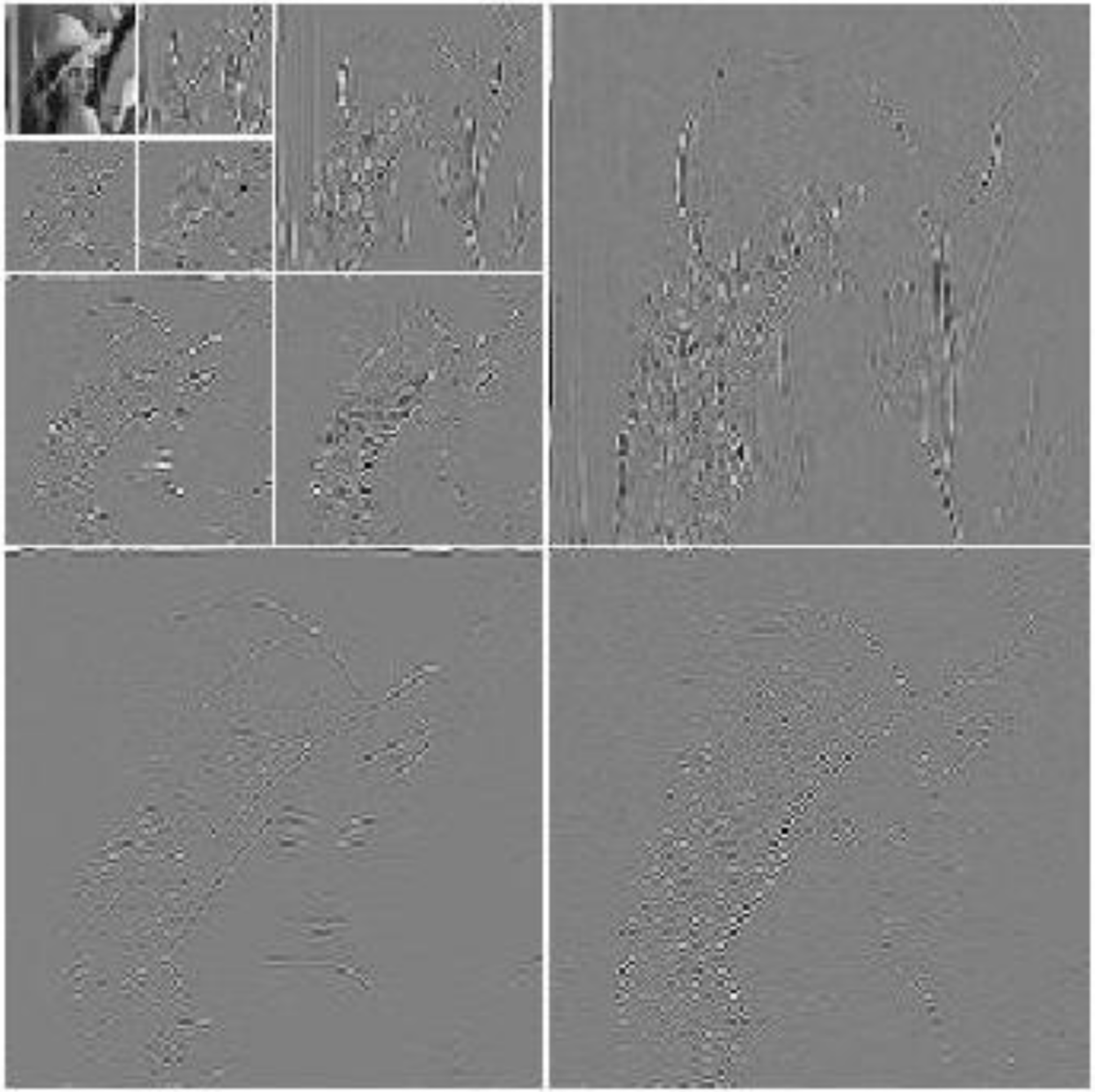


原圖：正方形

from R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Chap. 7, 2nd edition, Prentice Hall, New Jersey, 2002.

$\frac{1}{64}$

原圖：Lena



$x_{1,H2}$

$x_{1,H}$

$x_{1,H3}$

from R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Chap. 7, 2nd edition, Prentice Hall, New Jersey, 2002.

- compression & noise removing

保留 $x_{1,L}[m, n]$，捨棄其他部分

- (directional) edge detection

保留 $x_{1,H1}[m, n]$

捨棄其他部分

或保留 $x_{1,H2}[m, n]$

- $x_{1,H3}[m, n]$ 當中所包含的資訊較少

corner detection?

14.4 Complexity of the DWT

$x[n] * y[n]$, length($x[n]$) $= N$, length($y[n]$) $= L$,

$$\underbrace{IDFT_{N+L-1}}\left[\underbrace{DFT_{N+L-1}\left(x[n]\right)}DFT_{N+L-1}\left(y[n]\right)\right]$$

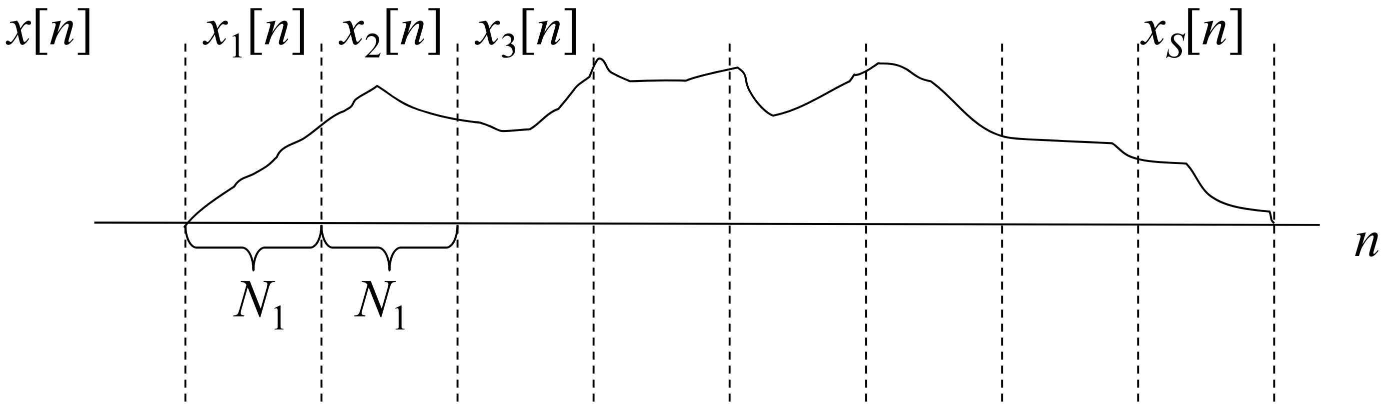
(N+L–1)-point discrete Fourier transform (DFT)

(N+L–1)-point inverse discrete Fourier transform (IDFT)

(1) Complexity of the 1-D DWT (without sectioned convolution)

$$(N+L-1)\log_2(N+L-1) \approx N\log_2 N$$

(2) 當 $N >>> L$ 時，使用 "sectioned convolution" 的技巧



將 $x[n]$ 切成很多段，每段長度為 N_1 $(N > N_1 >> L)$

總共有 $S = N / N_1$ 段

$\because x[n] = x_1[n] + x_2[n] + \cdots\cdots + x_S[n]$

$$x[n] * g[n] = \underset{N_1}{x_1[n]} * \underset{L}{g[n]} + x_2[n] * g[n] + \cdots\cdots + x_S[n] * g[n]$$

output : $(N_1 + L - 1)$-point

$$x[n] * h[n] = x_1[n] * h[n] + x_2[n] * h[n] + \cdots\cdots + x_S[n] * h[n]$$

complexity:

$$
\begin{aligned}
S(N_1+L-1)\log_2(N_1+L-1) &\approx SN_1\log_2(N_1+L-1) \\
&= N\log_2(N_1+L-1) \\
&\approx N\log_2 N_1
\end{aligned}
$$

$S: \frac{N}{N_1}$

N_1 is a fixed constant

- 重要概念：

The complexity of the 1-D DWT is **linear with *N***

$$O(N)$$

when $N >>> L$

(3) Multiple stages 的情形下

- 若 $x_{a,H}[n]$ 不再分解

Complexity 近似於: $$\left(N+\frac{N}{2}+\frac{N}{4}+\frac{N}{8}+\cdots\cdots+2\right)\log_2 N_1$$
$$=(2N-2)\log_2 N_1 \approx 2N\log_2 N_1$$

- 若 $x_{a,H}[n]$ 也細分

Complexity 近似於:

$$\left(N+2\frac{N}{2}+4\frac{N}{4}+8\frac{N}{8}+\cdots\cdots+\frac{N}{2}\cdot 2\right)\log_2 N_1$$
$$=(N\log_2 N)\log_2 N_1$$

(和 DFT 相近)

(4) Complexity of the 2-D DWT on page 431 (without sectioned convolution)

$$M(N+L-1)\log_2(N+L-1)+(N+L-1)(M+L-1)\log_2(M+L-1)$$

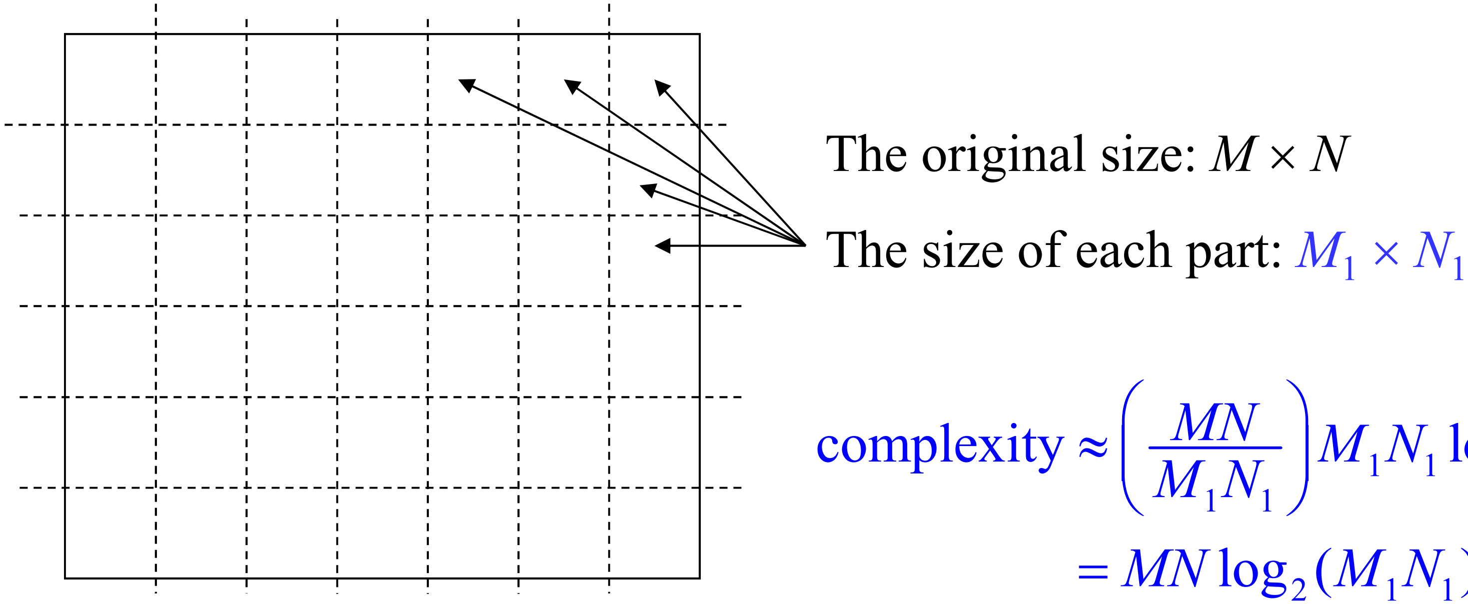
The first part needs M 1-D DWTs and the input for each 1-D DWT has N points

The second part needs N+L−1 1-D DWTs and the input for each 1-D DWT has M points

$$\begin{aligned}\text{complexity} &\approx MN\log_2 N + MN\log_2 M \\ &= MN(\log_2 N + \log_2 M) \\ &= MN\log_2(MN)\end{aligned}$$

(5) Complexity of the 2-D DWT (with Sectioned Convolution)

Image



$$\text{complexity} \approx \left(\frac{MN}{M_1N_1}\right) M_1N_1 \log_2(M_1N_1)$$
$$= MN \log_2(M_1N_1)$$

● **重要概念：**

If the method of the sectioned convolution is applied,

the complexity of the 2-D DWT is **linear with *MN*.**

$$O(MN)$$

(6) Multiple stages, two dimension

$x[m, n]$ 的 size 為 $M \times N$

- 若 $x_{a,H1}[n], x_{a,H2}[n], x_{a,H3}[n]$ 不細分，只細分 $x_{a,L}[n]$

total complexity

$$\left(MN + \frac{MN}{4} + \frac{MN}{16} + \cdots\cdots\right)\log_2(M_1N_1) \approx \frac{4}{3}MN\log_2(M_1N_1)$$

- 若 $x_{a,H1}[n], x_{a,H2}[n], x_{a,H3}[n]$ 也細分

total complexity

$$\left(MN + 4\frac{M}{2}\frac{N}{2} + 16\frac{M}{4}\frac{N}{4} + \cdots\cdots\right)\log_2(M_1N_1)$$

$$= \left[MN\log_2\left(\min(M,N)\right)\right]\log_2(M_1N_1)$$

14.5 Many Operations Also Have Linear Complexities

- 事實上，不只 wavelet 有 linear complexity

  當 input 和 filter 長度或大小相差懸殊時

  1-D convolution 的 complexity 是 linear with N.

  2-D convolution 的 complexity 是 linear with MN.

  (和傳統 $N\log_2 N$, $MN\log_2(MN)$ 的觀念不同）

很重要的概念

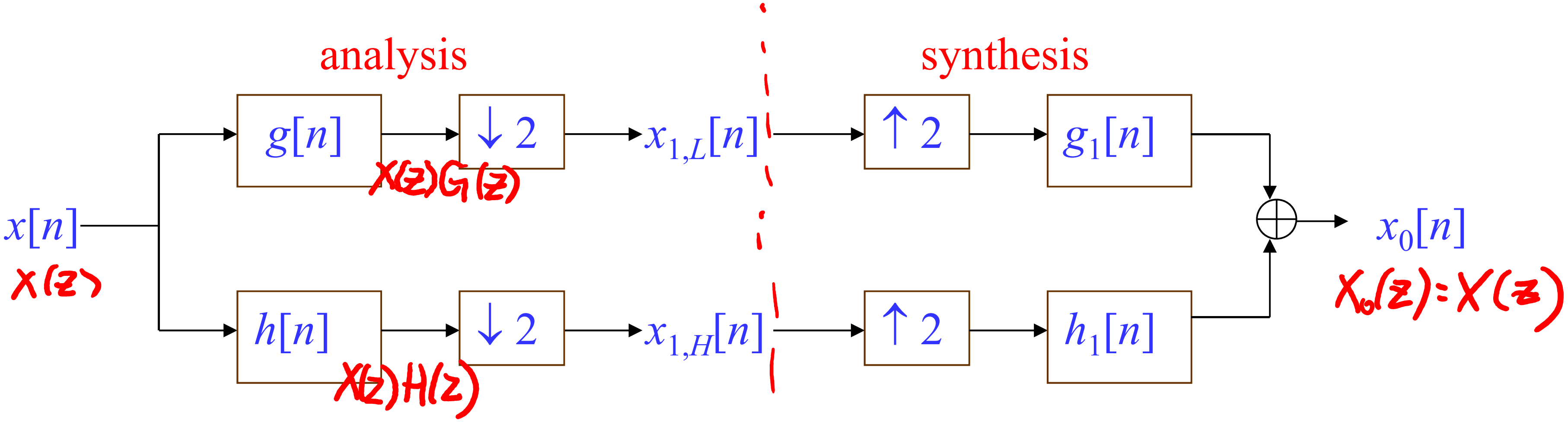
- Note：DCT 的 complexity 也是 linear with MN

(divided into 8×8 blocks)

complexity : $\dfrac{MN}{64}(8\times 8\log_2 8 + 8\times 8\log_2 8) = MN\log_2 64$

14.6 Reconstruction

forward wavelet transform inverse wavelet transform



$g_1[n], h_1[n]$ 要滿足什麼條件，才可以使得 $x_0[n] = x[n]$？

$\boxed{\uparrow 2}$: upsampling by the factor of 2

$a[n] \longrightarrow \boxed{\uparrow Q} \longrightarrow b[n]$

$$b[Qn] = a[n]$$

$$b[Qn + r] = 0 \quad \text{for } r = 1, 2, Q-1$$

Reconstruction Problem

$X(-z)=\sum_n x[n](-z)^{-n}$

$=\sum_n (-1)^n x[n] z^{-n}$

用 Z transform 來分析 $X(z)=\sum_{n=-\infty}^{\infty} x[n] z^{-n}$

- If $a[n]=b[2n]$, $\xrightarrow{\text{Z transform}}$ $\boxed{A(z)=\frac{1}{2}\left[B\left(z^{1/2}\right)+B\left(-z^{1/2}\right)\right]}$

  $\downarrow 2$ (downsampling)

(Proof):

$$B\left(z^{1/2}\right)+B\left(-z^{1/2}\right)=\sum_{n=-\infty}^{\infty} b[n] z^{-n/2}+\sum_{n=-\infty}^{\infty}(-1)^n b[n] z^{-n/2}$$

$$=\sum_{n=-\infty}^{\infty}\left(1+(-1)^n\right) b[n] z^{-n/2}=2 \sum_{n_1=-\infty}^{\infty} b[2n_1] z^{-n_1}=2 \sum_{n_1=-\infty}^{\infty} a[n_1] z^{-n_1}=2A(z)$$

$n=2n_1$

- If $a[2n]=b[n]$, $\Longrightarrow$ $\boxed{A(z)=B\left(z^2\right)}$

  $a[2n+1]=0$

  $\uparrow 2$ (upsampling)

$A(z)=\sum_n a[n] z^{-n}$

$=\sum_{n_1} a[2n_1] z^{-2n_1}=\sum_{n_1} b[n_1](z^2)^{-n_1}$

$$X_{1,L}(z) = \frac{1}{2}\left[X\left(z^{1/2}\right)G\left(z^{1/2}\right) + X\left(-z^{1/2}\right)G\left(-z^{1/2}\right)\right]$$

$$X_{1,H}(z) = \frac{1}{2}\left[X\left(z^{1/2}\right)H\left(z^{1/2}\right) + X\left(-z^{1/2}\right)H\left(-z^{1/2}\right)\right]$$

$$\begin{aligned} X_o(z) &= \frac{1}{2}\left[X(z)G(z) + X(-z)G(-z)\right]G_1(z) \\ &\quad + \frac{1}{2}\left[X(z)H(z) + X(-z)H(-z)\right]H_1(z) \\ &= \underbrace{\frac{1}{2}\left[G(z)G_1(z) + H(z)H_1(z)\right]}_{=1} X(z) \\ &\quad + \underbrace{\frac{1}{2}\left[G(-z)G_1(z) + H(-z)H_1(z)\right]}_{=0} X(-z) \end{aligned}$$

Perfect reconstruction: $X_o(z) = X(z)$

Perfect reconstruction: $X_o(z) = X(z)$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

條件：$\begin{cases} G(z)G_1(z) + H(z)H_1(z) = 2 \\ G(-z)G_1(z) + H(-z)H_1(z) = 0 \end{cases}$

$$\underbrace{\begin{bmatrix} G(z) & H(z) \\ G(-z) & H(-z) \end{bmatrix}}_{H_m(z)}\begin{bmatrix} G_1(z) \\ H_1(z) \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} G_1(z) \\ H_1(z) \end{bmatrix} = \begin{bmatrix} G(z) & H(z) \\ G(-z) & H(-z) \end{bmatrix}^{-1}\begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

$$= \frac{1}{G(z)H(-z)-H(z)G(-z)}\begin{bmatrix} H(-z) & -H(z) \\ -G(-z) & G(z) \end{bmatrix}\begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} G_1(z) \\ H_1(z) \end{bmatrix} = \frac{2}{G(z)H(-z)-H(z)G(-z)}\begin{bmatrix} H(-z) \\ -G(-z) \end{bmatrix}$$

$$\boxed{\begin{bmatrix} G_1(z) \\ H_1(z) \end{bmatrix} = \frac{2}{\det\left(\mathbf{H}_m(z)\right)}\begin{bmatrix} H(-z) \\ -G(-z) \end{bmatrix}}$$

where $\boxed{\det\left(\mathbf{H}_m(z)\right) = G(z)H(-z) - H(z)G(-z)}$

14.7 Reconstruction 的等效條件

$$\begin{bmatrix} G_1(z) \\ H_1(z) \end{bmatrix} = \frac{2}{\det(\mathbf{H}_m(z))} \begin{bmatrix} H(-z) \\ -G(-z) \end{bmatrix}$$

if and only if

$$\boxed{\sum_p g[p] g_1[2n-p] = \delta[n]}$$

$$\boxed{\sum_p h[p] h_1[2n-p] = \delta[n]}$$

$$\boxed{\sum_p g[p] h_1[2n-p] = 0}$$

$$\boxed{\sum_p g_1[p] h[2n-p] = 0}$$

這四個條件被稱作

biorthogonal conditions

(Proof)

Note: (a) $\det\left(\mathbf{H}_m(-z)\right) = -\det\left(\mathbf{H}_m(z)\right)$

(b) 令 $P(z) = G(z)G_1(z) = \dfrac{2G(z)H(-z)}{\det\left(\mathbf{H}_m(z)\right)}$

$$P(-z) = \frac{2G(-z)H(z)}{\det\left(\mathbf{H}_m(-z)\right)} = H(z)\frac{-2G(-z)}{\det\left(\mathbf{H}_m(z)\right)} = H(z)H_1(z)$$

Therefore,

$$H(z)H_1(z) = P(-z) = G(-z)G_1(-z)$$

From $G(z)G_1(z) + H(z)H_1(z) = 2$

$$G(z)G_1(z) + G(-z)G_1(-z) = 2$$

$$\Big\downarrow \text{ inverse } Z \text{ transform}$$

$$\sum_p g[p]g_1[n-p] + (-1)^n \sum_p g[p]g_1[n-p] = 2\delta[n]$$

$$\sum_p g[p]g_1[n-p] + (-1)^n \sum_p g[p]g_1[n-p] = 2\delta[n]$$

$$\downarrow$$

$$\boxed{\sum_p g[p]g_1[2n-p] = \delta[n]}$$ ← orthogonality 條件 1

(c) Similarly, substitute $G(z)G_1(z) = H(-z)H_1(-z)$

into $G(z)G_1(z) + H(z)H_1(z) = 2$

$$H(-z)H_1(-z) + H(z)H_1(z) = 2$$

↓ after the process the same as that of the above

$$\boxed{\sum_p h[p]h_1[2n-p] = \delta[n]}$$ ← orthogonality 條件 2

(d) Since
$$G(z)H_1(z)+G(-z)H_1(-z)$$
$$=-G(z)\frac{G(-z)}{\det(\mathbf{H}_m(z))}-G(-z)\frac{G(z)}{\det(\mathbf{H}_m(-z))}$$
$$=-\frac{G(z)G(-z)}{\det(\mathbf{H}_m(z))}+\frac{G(-z)G(z)}{\det(\mathbf{H}_m(z))}=0$$

$$\sum_p g[p]h_1[n-p]+(-1)^n\sum_p g[p]h_1[n-p]=0$$

$$\boxed{\sum_p g[p]h_1[2n-p]=0}$$ ← orthogonality 條件 3

(e) 同理 $G_1(z)H(z)+G_1(-z)H(-z)=0$

$$\boxed{\sum_p g_1[p]h[2n-p]=0}$$ ← orthogonality 條件 4

14.8 DWT 設計上的條件

- Reconstruction
- Finite length　　為了 implementation 速度的考量

$$\boxed{g[n] \neq 0 \text{ only when } -L \leq n \leq L}$$

$$\boxed{h[n] \neq 0 \text{ only when } -L \leq n \leq L}$$

$h_1[n]$, $g_1[n]$?

令 $\boxed{\det\left(\mathbf{H}_m(z)\right) = \alpha z^k}$　　則根據 page 448,

$$G_1(z) = 2\alpha^{-1} z^{-k} H(-z) \qquad H_1(z) = -2\alpha^{-1} z^{-k} G(-z)$$

複習: $x[n-k] \xrightarrow{Z\ transform} z^{-k} X(z)$

$$g_1[n] = 2\alpha^{-1}(-1)^{n-k} h[n-k] \qquad h_1[n] = -2\alpha^{-1}(-1)^{n-k} g[n-k]$$

- 因為 $\det\left(\mathbf{H}_m(z)\right) = G(z)H(-z) - H(z)G(-z)$

$$\det\left(\mathbf{H}_m(z)\right) = -\det\left(\mathbf{H}_m(-z)\right)$$

k 必需為 odd

- Lowpass-highpass pair

14.9 整理：DWT 的四大條件

$G_1(z) = 2\alpha^{-1} z^{-k} H(-z)$
$H_1(z) = 2\alpha^{-1} z^{-k} (-G(-z))$

(1) $$\begin{bmatrix} G_1(z) \\ H_1(z) \end{bmatrix} = \frac{2}{\det(\mathbf{H}_m(z))} \begin{bmatrix} H(-z) \\ -G(-z) \end{bmatrix}$$

(for reconstruction)

$g_1[n] = 2\alpha^{-1}(-1)^{n-k} h[n-k]$

$$\det(\mathbf{H}_m(z)) = G(z)H(-z) - H(z)G(-z)$$

$x[n-k] \xrightarrow{Z} z^{-k}X(z)$

$h_1[n] = -2\alpha^{-1}(-1)^{n-k} g[n-k]$

(2) $h[n] \neq 0$ only when $0 \leq n \leq L-1$

$g[n] \neq 0$ only when $0 \leq n \leq L-1$

($h[n]$, $g[n]$ have finite lengths)

(3) $\boxed{\det(\mathbf{H}_m(z)) = \alpha z^k}$　　k 必需為 odd

$1+z^{-1}$

($h_1[n]$, $g_1[n]$ have finite lengths)

(4) $h[n]$ 為 highpass filter

$g[n]$ 為 lowpass filter

(lowpass and highpass pair)

第三個條件較難達成，是設計的核心

14.10 Two Types of Perfect Reconstruction Filters

(1) QMF (quadrature mirror filter)

$G(z)$ satisfy $\boxed{G^2(z) - G^2(-z) = 2z^k}$ <u>k is odd</u>

$g[n]$ has finite length

$H(z) = G(-z)$ $\qquad h[n] = (-1)^n g[n]$

$G_1(z) = G(z) z^{-k}$ $\qquad g_1[n] = g[n-k]$

$H_1(z) = -G(-z) z^{-k}$ $\qquad h_1[n] = (-1)^{n-k+1} g[n-k]$

$\det(\mathbf{H}_m(z)) = G(z)H(-z) - H(z)G(-z) = ?$ $\quad 2z^k$

$= G(z)G(z) - G(-z)G(-z)$

(2) Orthonormal

$$G(z) = G_1(z^{-1})$$ $g[n] = g_1[-n]$

$$H(z) = -z^k G_1(-z)$$ k is odd $h[n] = (-1)^n g_1[n+k]$

$(-1)^{n+k+1}$

$G_1(z)$ satisfy $G_1(z)G_1(z^{-1}) + G_1(-z)G_1(-z^{-1}) = 2$

$g_1[n]$ has finite length

$G_1(-z) \xrightarrow{z^{-1}} (-1)^n g_1[n]$

$$H_1(z) = -z^{-k} G_1(-z^{-1}) = H(z^{-1})$$ $h_1[n] = (-1)^n g_1[-n+k]$

$h_1[n] = h[-n]$

$$\det(\mathbf{H}_m(z)) = G(z)H(-z) - H(z)G(-z)$$

$$= G_1(z^{-1})z^k G_1(z) + G_1(-z^{-1})z^k G_1(-z) = 2z^k$$

大部分的 wavelet 屬於 orthonormal wavelet

文獻上，有時會出現另一種 perfect reconstruction filter, 稱作 CQF (conjugate quadrature filter)

然而，CQF 本質上和 orthonormal filter 相同

14.11 Several Types of Discrete Wavelets

- discrete Haar wavelet (最簡單的) = 2-point Daubechies wavelet

$g[-1] = g[0] = 1$ $\qquad g[n] = 0$ otherwise

$h[-1] = -1, \quad h[0] = 1$ $\qquad h[n] = 0$ otherwise

$g_1[0] = g_1[1] = 1$ $\qquad g_1[n] = 0$ otherwise

$h_1[0] = 1, \quad h_1[1] = -1$ $\qquad h_1[n] = 0$ otherwise

是一種 orthonormal filter

- discrete Daubechies wavelet (8-point case)

vanish moment = 4

$$g[n]=[-0.0106 \quad 0.0329 \quad 0.0308 \quad -0.1870 \quad -0.0280 \quad 0.6309 \quad 0.7148 \quad 0.2304]$$

$n = 0 \sim 7$ $\qquad g[n]=0$ otherwise

$$h[n]=[0.2304 \quad -0.7148 \quad 0.6309 \quad 0.0280 \quad -0.1870 \quad -0.0308 \quad 0.0329 \quad 0.0106]$$

$n = 0 \sim 7$ $\qquad h[n]=0$ otherwise

$k = -7$ $\qquad h[n] = (-1)^n g_1[n-7]$

$$g_1[n]=[0.2304 \quad 0.7148 \quad 0.6309 \quad -0.0280 \quad -0.1870 \quad 0.0308 \quad 0.0329 \quad -0.0106]$$

$n = -7 \sim 0$ $\qquad g_1[n]=0$ otherwise $\qquad g_1[n] = g[-n]$

$$h_1[n]=[0.0106 \quad 0.0329 \quad -0.0308 \quad -0.1870 \quad 0.0280 \quad 0.6309 \quad -0.7148 \quad 0.2304]$$

$n = -7 \sim 0$ $\qquad h_1[n]=0$ otherwise

$h_1[n] = h[-n]$

- discrete Daubechies wavelet (4-point case)

$$g[n] = [-0.1294 \quad 0.2241 \quad 0.8365 \quad 0.4830]$$

- discrete Daubechies wavelet (6-point case)

$$g[n] = [0.0352 \quad -0.0854 \quad -0.1350 \quad 0.4599 \quad 0.8069 \quad 0.3327]$$

$h[n]$: ? $(k=-5)$ $g_1[n]$: ? $h_1[n]$: ?

- discrete Daubechies wavelet (10-point case)

$$g[n] = [0.0033 \quad -0.0126 \quad -0.0062 \quad 0.0776 \quad -0.0322 \quad -0.2423$$
$$0.1384 \quad 0.7243 \quad 0.6038 \quad 0.1601]$$

- discrete Daubechies wavelet (12-point case)

$$g[n] = [-0.0011 \quad 0.0048 \quad 0.0006 \quad -0.0316 \quad 0.0275 \quad 0.0975$$
$$-0.1298 \quad -0.2263 \quad 0.3153 \quad 0.7511 \quad 0.4946 \quad 0.1115]$$

symlet (6-point case) symmetric wavelet

v.m: 3

$$g[n] = [0.0352 \quad -0.0854 \quad -0.1350 \quad 0.4599 \quad 0.8069 \quad 0.3327]$$

symlet (8-point case) v.m: 4

$$g[n] = [-0.0758 \quad -0.0296 \quad 0.4976 \quad \underline{0.8037} \quad 0.2979 \quad -0.0992 \quad -0.0126 \quad 0.0322]$$

symlet (10-point case) v.m: 5

$$g[n] = [0.0273 \quad 0.0295 \quad -0.0391 \quad 0.1994 \quad \underline{0.7234} \quad 0.6340 \quad 0.0166 \quad -0.1753 \quad -0.0211 \quad 0.0195]$$

Daubechies wavelets and symlets are defined for <u>N is a multiple of 2</u>

6p-point coiflet has the vanish moment of p

coiflet (6-point case)

V.M= 1

$$g[n]=[-0.0157 \quad -0.0727 \quad 0.3849 \quad 0.8526 \quad 0.3379 \quad -0.0727]$$

coiflet (12-point case) V.M=2

$$g[n]=[0.0232 \quad -0.0586 \quad -0.0953 \quad 0.5460 \quad 1.1494 \quad 0.5897$$
$$-0.1082 \quad -0.0841 \quad 0.0335 \quad 0.0079 \quad -0.0026 \quad -0.0010]$$

Coiflets are defined for <u>N is a multiple of 6</u>

The Daubechies wavelet, the symlet, and the coiflet are all orthonormal filters.

The Daubechies wavelet, the symlet, and the coiflet are all derived from the "continuous wavelet with discrete coefficients" case.

Physical meanings:

- Daubechies wavelet

  The ? point Daubechies wavelet has the vanish moment of p.

- Symlet

  The vanish moment is the same as that of the Daubechies wavelet, but the filter is more symmetric.

- ~~Coilfet~~ Coiflet

  The ? point coiflet has the vanish moment of p.

  The scaling function also has the vanish moment.

$$\int_{-\infty}^{\infty} \phi(t)\,dt \neq 0 \qquad \int_{-\infty}^{\infty} t^k \phi(t)\,dt = 0 \quad \text{for } 1 \leqq k \leqq p$$

14.12 產生 Discrete Daubechies Wavelet 的流程

Step 1 $P(y)=\sum_{k=0}^{p-1} C_k^{p-1+k} y^k$

Q: 如何用 Matlab 寫出 C_n^m

(When p = 2, $P(y) = 2y + 1$)

Step 2 $P_1(z)=P\left(\frac{2-z-z^{-1}}{4}\right)$

Hint: $\left((2-z-z^{-1})/4\right)^k$ 在 Matlab 當中，可以用 [-.25, .5, -.25] 自己和自己 convolution k-1 次算出來

(When p = 2, $P_1(z) = 2 - 0.5z - 0.5z^{-1}$)

Step 3 算出 $z^kP_1(z)$ 的根 (i.e., $z^kP_1(z) = 0$ 的地方)

Q: 在 Matlab 當中應該用什麼指令

(When p = 2, roots = 3.7321, 0.2679)

Step 4　算出

$$P_2(z) = (z - z_1)(z - z_2)\cdots(z - z_{p-1})$$

$z_1, z_2, \ldots, z_{p\text{-}1}$ 為 $z^k P_1(z)$ 當中，絕對值小於 1 的 roots

Step 5　算出

$$G_0(z) = (1+z)^p P_2(z)$$

$$g_0[n] = Z^{-1}\{G_0(z)\}$$

注意：Z transform 的定義為 $G_0(z) = \sum_n g_0[n] z^{-n}$

所以 coefficients 要做 reverse

(When p = 2, $g_0[n]$ = [1　1.7321　0.4641　-0.2679])

n = -3 ~ 0

Step 6 Normalization

$$g_1[n] = \frac{g_0[n]}{\|g_0\|}$$

(When p = 2, $g_1[n]$ = [0.4830 0.8365 0.2241 -0.1294])

n = -3 ~ 0

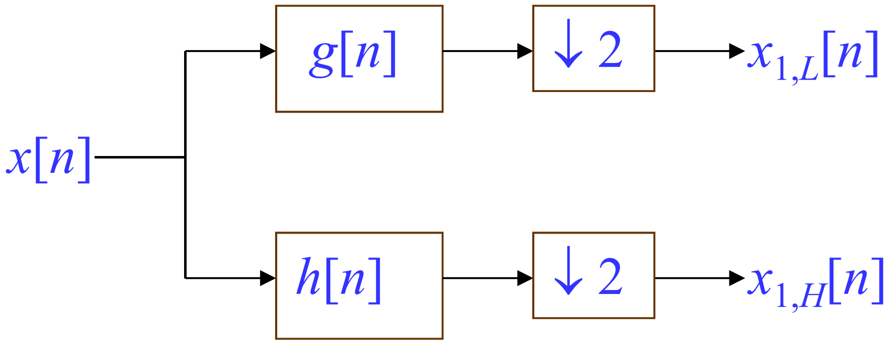
Step 7 Time reverse

$$g[n] = g_1[-n] \qquad h[n] = (-1)^n g[2p-1-n]$$

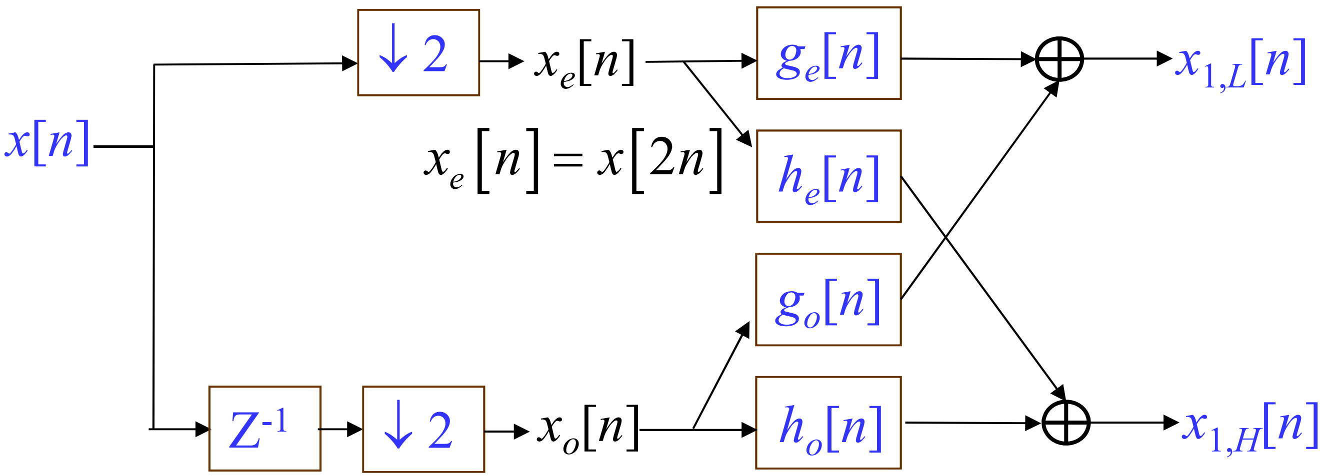
Then, the (2p)-point discrete Daubechies wavelet transform can be obtained

14.13 2x2 Structure Form and the Lifting Scheme

The analysis part



can be changed into the following 2x2 structure



$$x_o[n] = x[2n-1]$$

where

$g_e[n] = g[2n]$

$g_o[n] = g[2n+1]$

$h_e[n] = h[2n]$

$h_o[n] = h[2n+1]$

Z^{-1} means delayed by 1

(Proof): From page 427,

$$x_{1,L}[n] = \sum_{k=0}^{K-1} x[2n-k]\,g[k]$$

$$x_{1,L}[n] = \sum_{k=0}^{K/2-1} x[2n-2k]\,g[2k] + \sum_{k=0}^{K/2-1} x[2n-2k-1]\,g[2k+1]$$

$$= \sum_{k=0}^{K/2-1} x_e[n-k]\,g_e[k] + \sum_{k=0}^{K/2-1} x_o[n-k]\,g_o[k]$$

where

$$x_e[n] = x[2n], \qquad x_o[n] = x[2n-1]$$

$$x[n] \longrightarrow \boxed{Z^{-1}} \longrightarrow \boxed{\downarrow 2} \longrightarrow x[2n\text{-}1]$$

Similarly,

$$x_{1,H}[n] = \sum_{k=0}^{K/2-1} x_e[n-k]\,h_e[k] + \sum_{k=0}^{K/2-1} x_o[n-k]\,h_o[k]$$

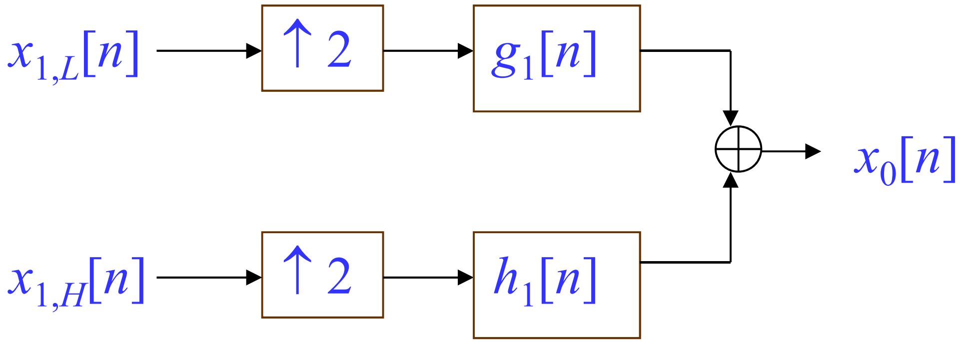
Original Structure:

Two Convolutions of an N-length input and an L-length filter

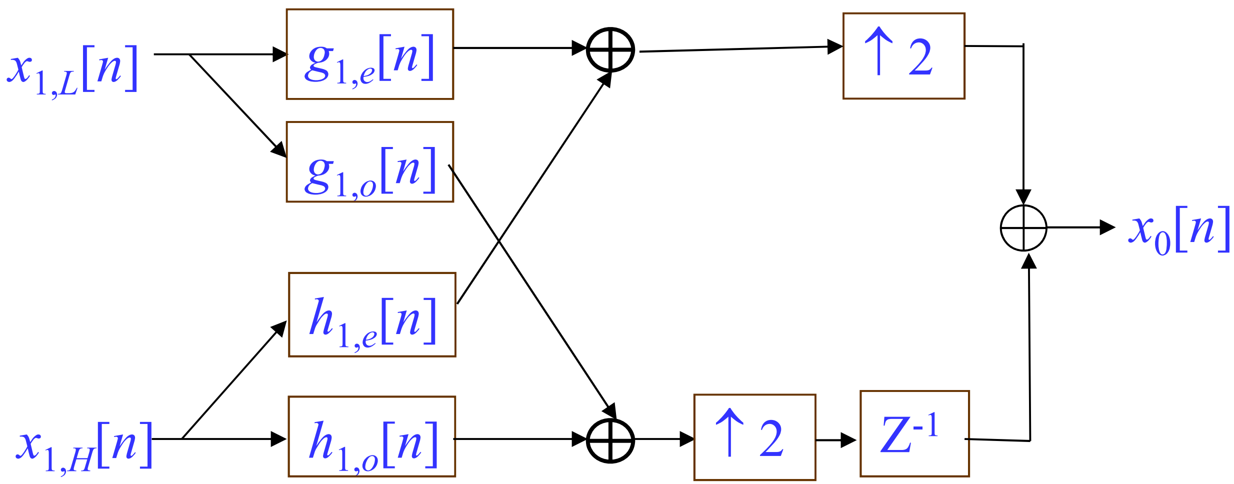
New Structure:

Four Convolutions of an $(N/2)$-length input and an $(L/2)$-length filter, which is more efficient. (Why?)

Similarly, the synthesis part



can be changed into the following 2x2 structure



where

$$g_{1,e}[n] = g_1[2n]$$

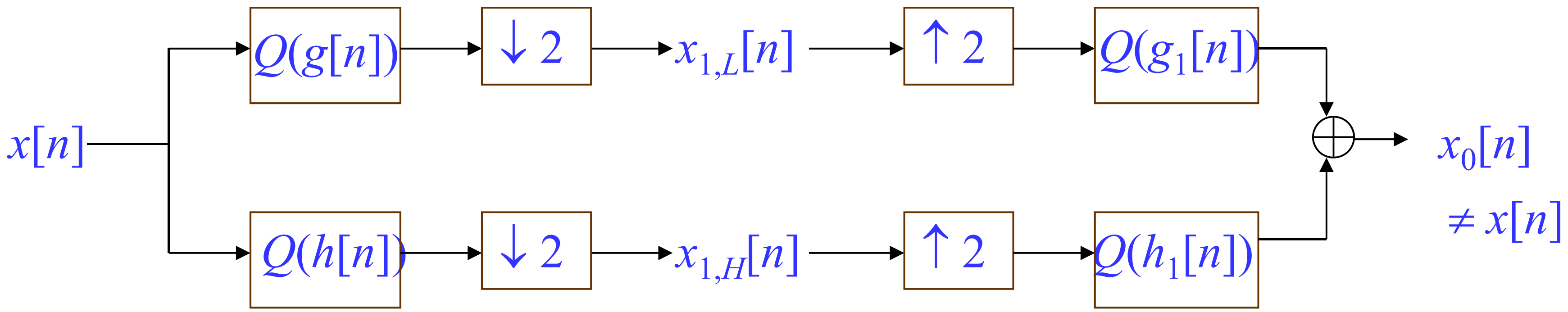
$$g_{1,o}[n] = g_1[2n+1]$$

$$h_{1,e}[n] = h_1[2n]$$

$$h_{1,o}[n] = h_1[2n+1]$$

14.14 Lifting Scheme

After performing quantization, the DWT may not be perfectly reversible



$Q(\ )$ means quantization (rounding, flooring, ceiling ……)

Lifting Scheme:

Reversible After Quantization

From page 468

$$\begin{bmatrix} G_e(z) & G_o(z) \\ H_e(z) & H_o(z) \end{bmatrix}\begin{bmatrix} X_e(z) \\ X_o(z) \end{bmatrix} = \begin{bmatrix} X_{1,L}(z) \\ X_{1,H}(z) \end{bmatrix}$$

Since

$$G_e(z) = \left[G\left(z^{1/2}\right) + G\left(-z^{1/2}\right)\right]/2 \qquad G_o(z) = z^{1/2}\left[G\left(z^{1/2}\right) - G\left(-z^{1/2}\right)\right]/2$$

$$H_e(z) = \left[H\left(z^{1/2}\right) + H\left(-z^{1/2}\right)\right]/2 \qquad H_o(z) = z^{1/2}\left[H\left(z^{1/2}\right) - H\left(-z^{1/2}\right)\right]/2$$

$$\det\left(\begin{bmatrix} G_e(z) & G_o(z) \\ H_e(z) & H_o(z) \end{bmatrix}\right) = z^{\frac{1}{2}}\left(G\left(-z^{\frac{1}{2}}\right)H\left(z^{\frac{1}{2}}\right) - G\left(z^{\frac{1}{2}}\right)H\left(-z^{\frac{1}{2}}\right)\right)/2$$

from page 455 one set that

$$\det\left(\mathbf{H}_m(z)\right) = G(z)H(-z) - H(z)G(-z) = -z^{-2m-1}$$

then

$$\det\left(\begin{bmatrix} G_e(z) & G_o(z) \\ H_e(z) & H_o(z) \end{bmatrix}\right) = z^{-m}/2$$

Then $\begin{bmatrix} G_e(z) & G_o(z) \\ H_e(z) & H_o(z) \end{bmatrix}$ can be decomposed into

$$\boxed{\begin{bmatrix} G_e(z) & G_o(z) \\ H_e(z) & H_o(z) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & z^{-m} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ L_1(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & L_2(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ L_3(z) & 1 \end{bmatrix}}$$

where

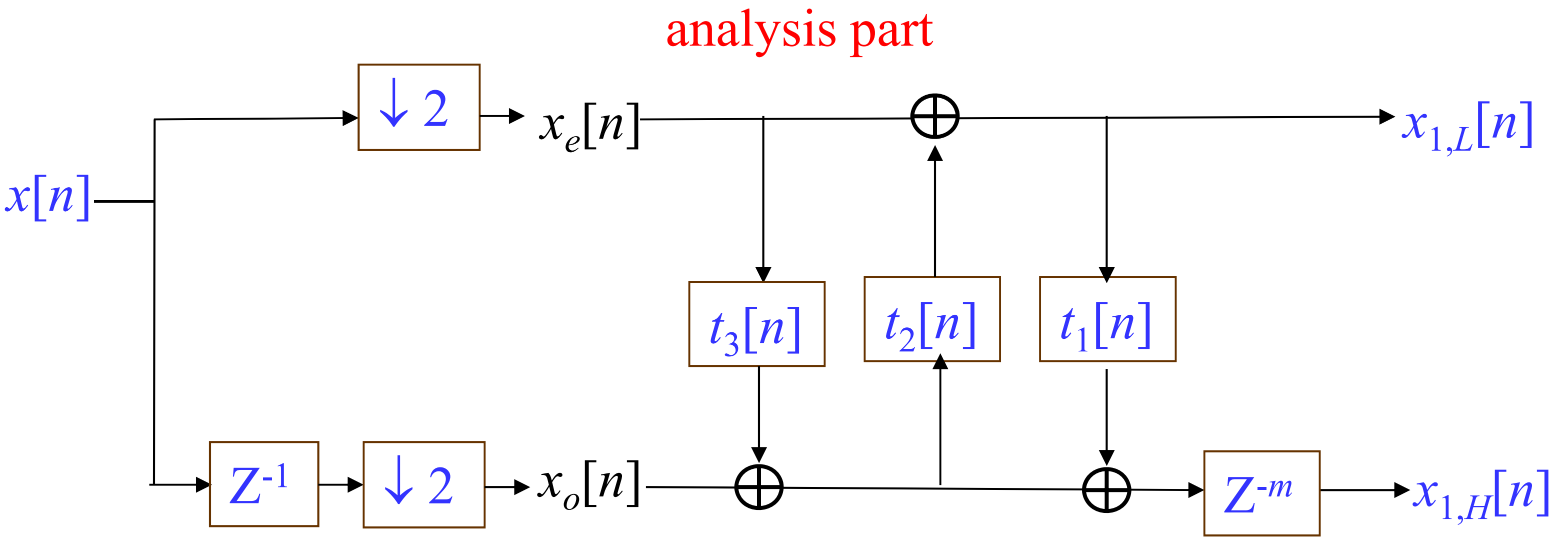
$$L_1(z) = \frac{z^m H_o(z) - 1}{G_o(z)} \qquad L_2(z) = G_o(z) \qquad L_3(z) = \frac{G_e(z) - 1}{G_o(z)}$$

Then the DWT can be approximated by

$$\boxed{\begin{bmatrix} 1 & 0 \\ 0 & z^{-m} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ T_1(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & T_2(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ T_3(z) & 1 \end{bmatrix} \begin{bmatrix} X_e(z) \\ X_o(z) \end{bmatrix} = \begin{bmatrix} X_{1,L}(z) \\ X_{1,H}(z) \end{bmatrix}}$$

where $T_1(z) \cong L_1(z)$, $T_2(z) \cong L_2(z)$, $T_3(z) \cong L_3(z)$

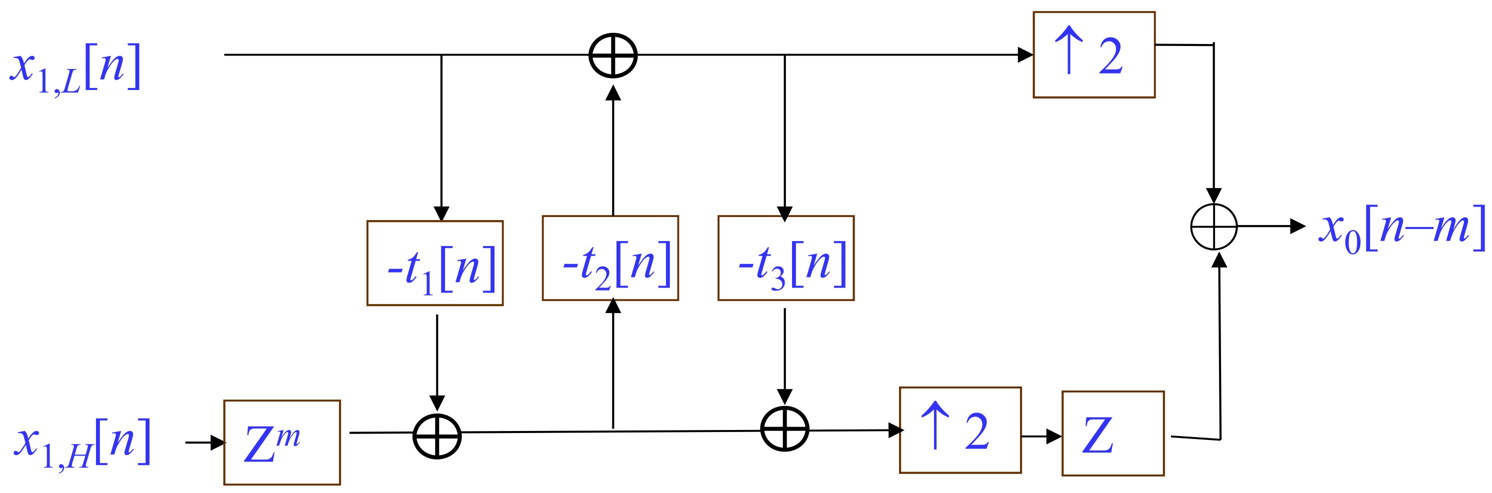
Lifting Scheme



The Z transforms of $t_1[n]$, $t_2[n]$, and $t_3[n]$ are $T_1(z)$, $T_2(z)$, and $T_3(z)$, respectively.

Lifting Scheme

synthesis part



If one perform quantization for $t_1[n]$, $t_2[n]$, and $t_3[n]$, then the discrete wavelet transform is still reversible.

$$\begin{bmatrix} 1 & 0 \\ L_1(z) & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 \\ -L_1(z) & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ Q\left(L_1(z)\right) & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 \\ -Q\left(L_1(z)\right) & 1 \end{bmatrix}$$

W. Sweldens, "The lifting scheme: a construction of second generation wavelets," *Applied Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186-200, 1996.

I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Applicat.*, vol. 4, pp. 246-269. 1998.

附錄十四 誤差計算的標準

若原來的信號是 $x[m, n]$，要計算 $y[m, n]$ 和 $x[m, n]$ 之間的誤差，有下列幾種常見的標準

(1) maximal error

$$Max\left(\left|y[m,n]-x[m,n]\right|\right)$$

(2) square error

$$\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2$$

(3) error norm (i.e., Euclidean distance)

$$\sqrt{\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2}$$

(4) mean square error (MSE)，信號處理和影像處理常用

$$\frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2$$

(5) root mean square error (RMSE)

$$\sqrt{\frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2}$$

(6) normalized mean square error (NMSE)

$$\frac{\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2}{\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|x[m,n]\right|^2}$$

(7) normalized root mean square error (NRMSE)，
信號處理和影像處理常用

$$\sqrt{\frac{\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2}{\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|x[m,n]\right|^2}}$$

(8) signal to noise ratio (SNR)，信號處理常用

$$10\log_{10}\left(\frac{\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|x[m,n]\right|^2}{\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2}\right)$$

(9) peak signal to noise ratio (PSNR)，影像處理常用

$$10\log_{10}\left(\frac{X_{Max}^2}{\frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y[m,n]-x[m,n]\right|^2}\right)$$

X_{Max}: the maximal possible value of $x[m, n]$

In image processing, $X_{Max} = 255$

for color image: $$10\log_{10}\left(\frac{X_{Max}^2}{\frac{1}{3MN}\sum_{R,G,B}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|y_{color}[m,n]-x_{color}[m,n]\right|^2}\right)$$

color = R, G, or B

(10) structural dissimilarity (DSSIM)

有鑑於 MSE 和 PSNR 無法完全反應人類視覺上所感受的誤差，在 2004 年被提出來的新的誤差測量方法

$$DSSIM\left(x,y\right)=1-SSIM\left(x,y\right)$$

$$SSIM\left(x,y\right)=\frac{\left(2\mu_x\mu_y+c_1L\right)}{\left(\mu_x^2+\mu_y^2+c_1L\right)}\frac{\left(2\sigma_{xy}+c_2L\right)}{\left(\sigma_x^2+\sigma_y^2+c_2L\right)}$$

μ_x, μ_y: means of x and y $\quad$ σ_x^2, σ_y^2: variances of x and y

$\sigma_x\sigma_y$: covariance of x and y $\quad$ c_1, c_2: adjustable constants

L: the maximal possible value of x − the minimal possible value of x

Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Trans. Image Processing*, vol. 13, no. 4, pp. 600−612, Apr. 2004.

XV. Applications of Wavelet Transforms

Wavelet 所適用的 applications，通常有以下兩大特點：

(1) 信號的頻率分佈，會隨著不同的時間(或地點)有較大變異

(2) Multiscale 的分析扮演重要的角色

Larger sampling interval $\rightarrow$ ignoring the detail

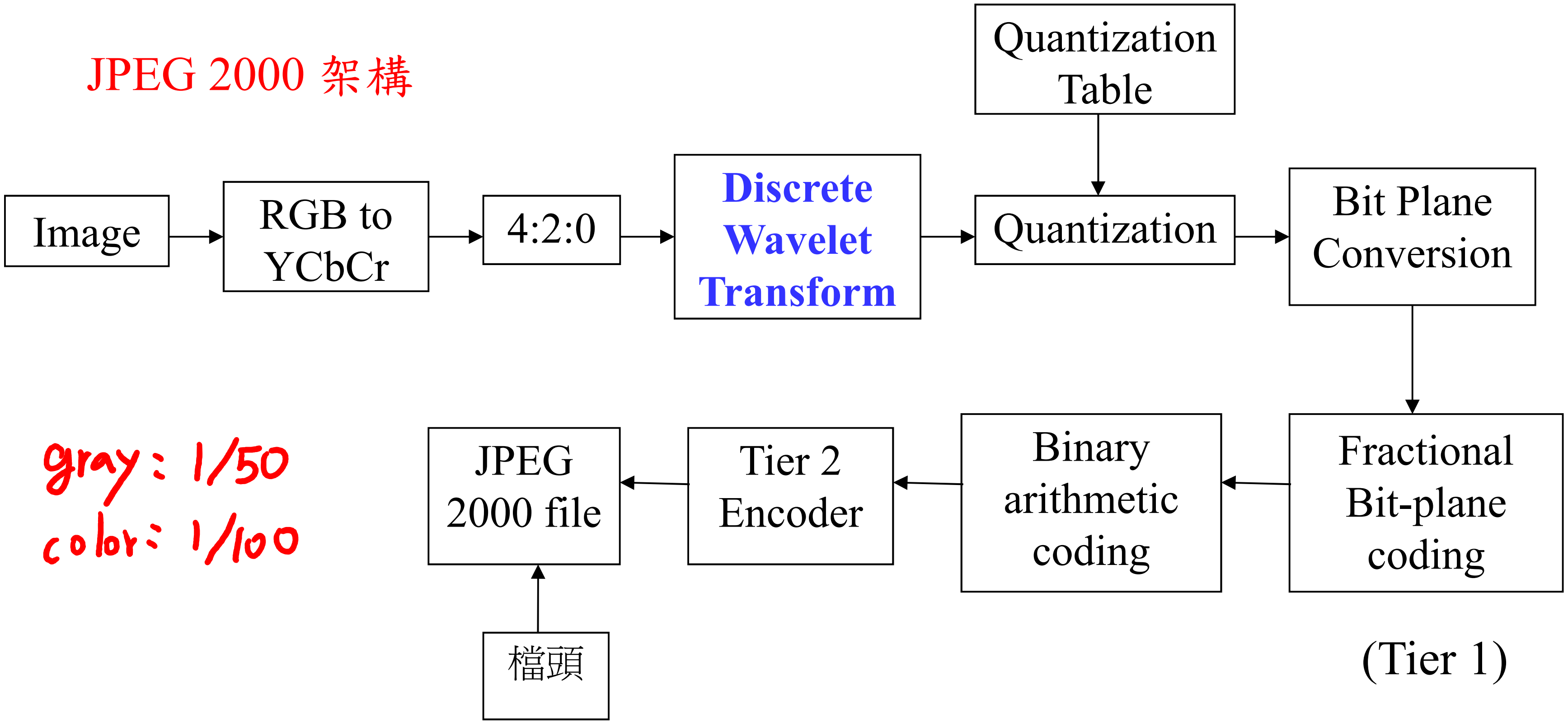
Smaller sampling interval $\rightarrow$ requiring a lot of data

Wavelet transforms compromise them.

目前，文獻上，80% 以上的應用和 image processing 有關

(1) Image Compression (JPEG 2000)

JPEG 2000 架構



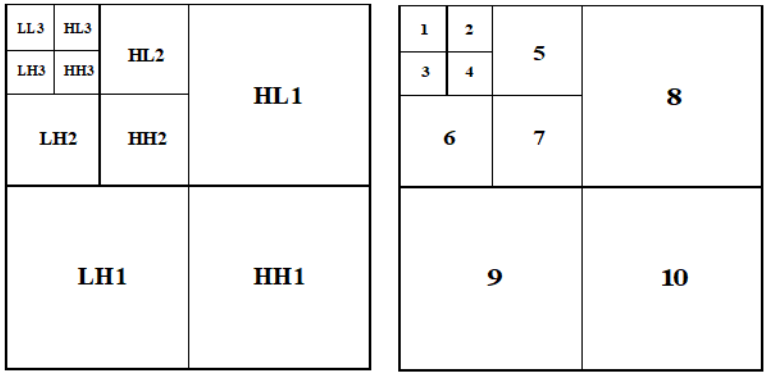
gray: 1/50
color: 1/100

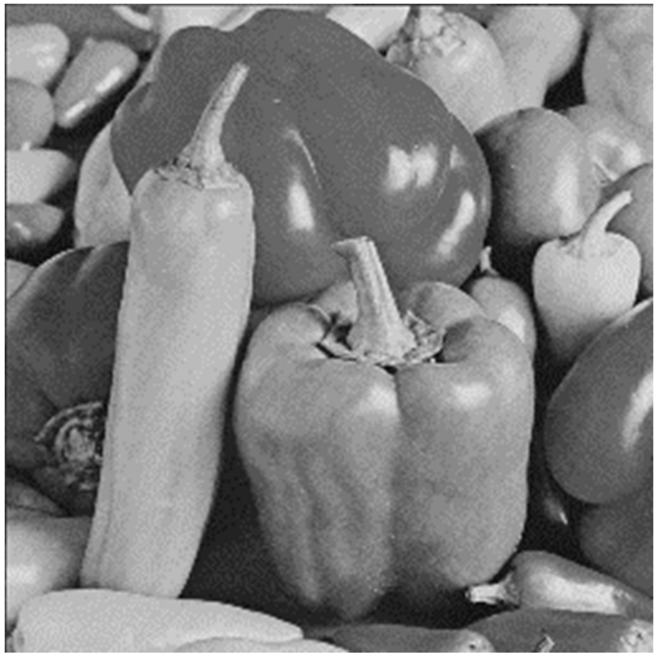
Tier 1: zero coding, sign coding, magnitude refinement coding, run length coding

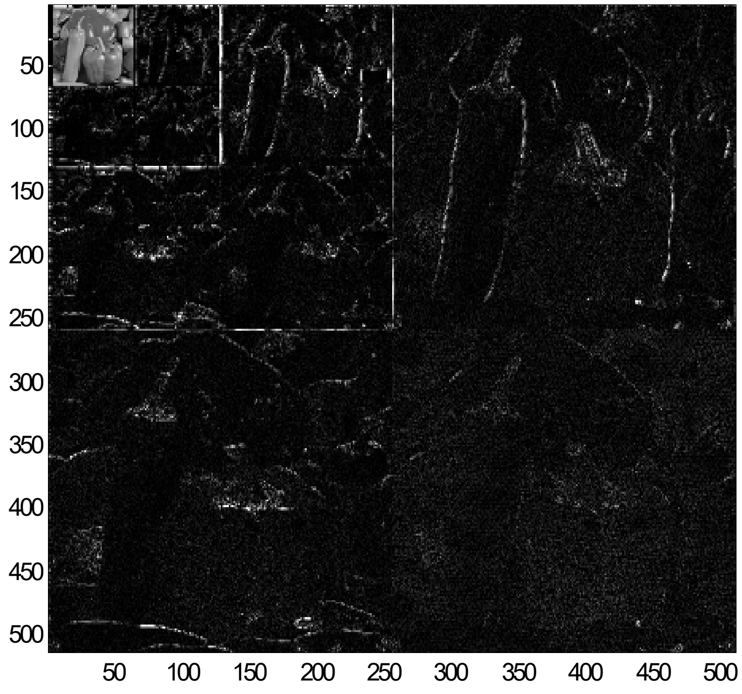
Tier 2: 用以控制檔案大小(例如只取比較重要的地方編碼)

註：感謝2010年修課的潘冠臣同學幫忙整理

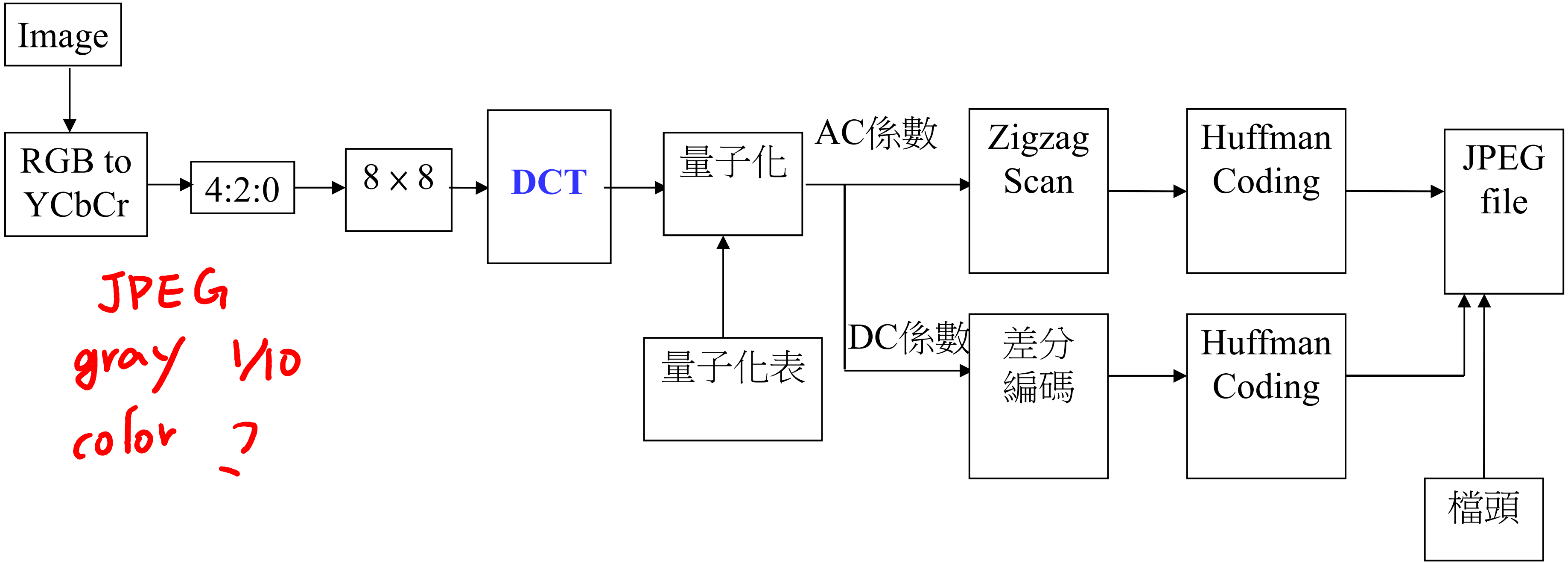
The subbands of the discrete wavelet transform (DWT)







比較：傳統 JPEG 架構



問題：由於 8 × 8 的切割，在高壓縮率時會造成 blocking effect

Original image

JPEG

DCT-based
image compression

CR = 53.4333

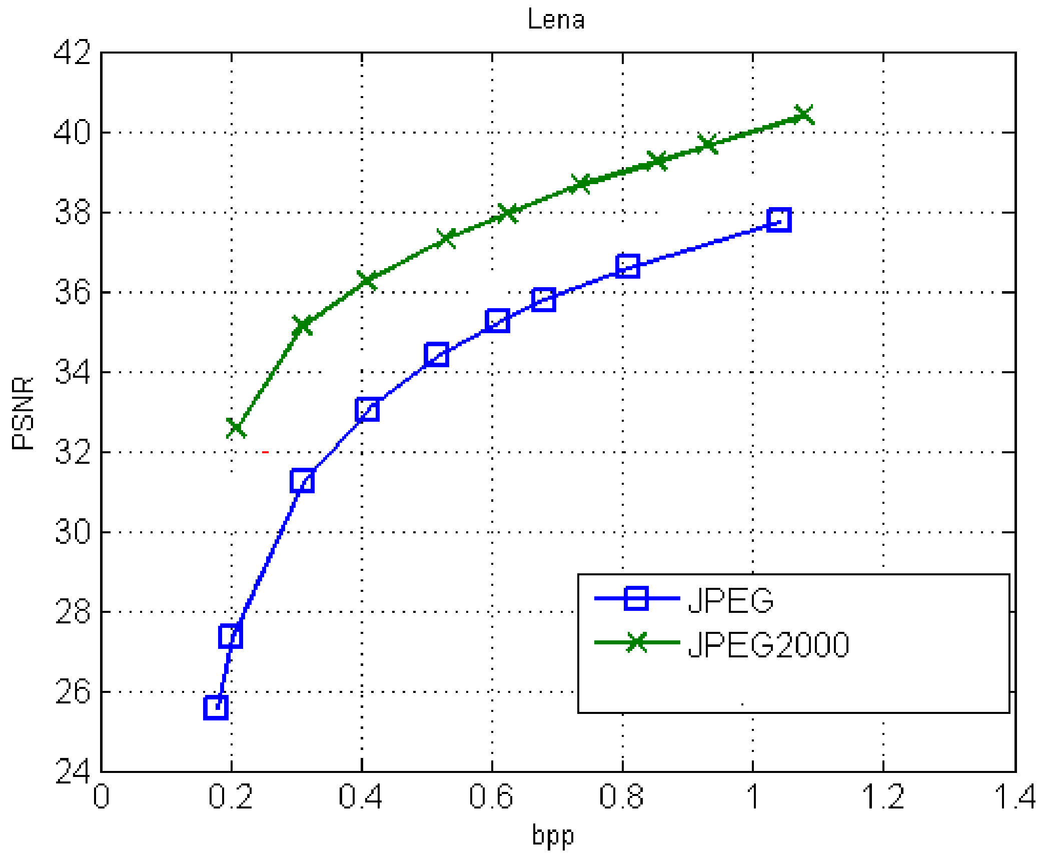
JPEG2000

Wavelet-based
image compression

CR = 51.3806

CR: compression ratio

註：感謝2006年修課的黃俊德同學



bpp: bit per pixel (每一點平均需要多少個 bits)

PSNR: peak signal to noise ratio (PSNR), see page 480

使用 JPEG 2000 做影像壓縮的優點：

(1)

(2)

(3)

所以，在高壓縮率之下，重建的影像仍有不錯的品質

Question:

Why JPEG 2000 has not replaced the status of JPEG now?

參考資料

C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consumer Electronics*, vol. 46, no. 4, pp.1103-1127, Nov. 2000.

Another Compression Algorithm: SPIHT

Using the correlation among high frequency parts in different layers

B.J. Kim, Z. Xiong, and W.A. Pearlman. "Low bit-rate scalable video coding with 3-D set partitioning in hierarchical trees (3-D SPIHT)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 1374-1387, 2000.

(2) Edge and Corner Detection

(3) Pattern recognition

(a) Feature extraction

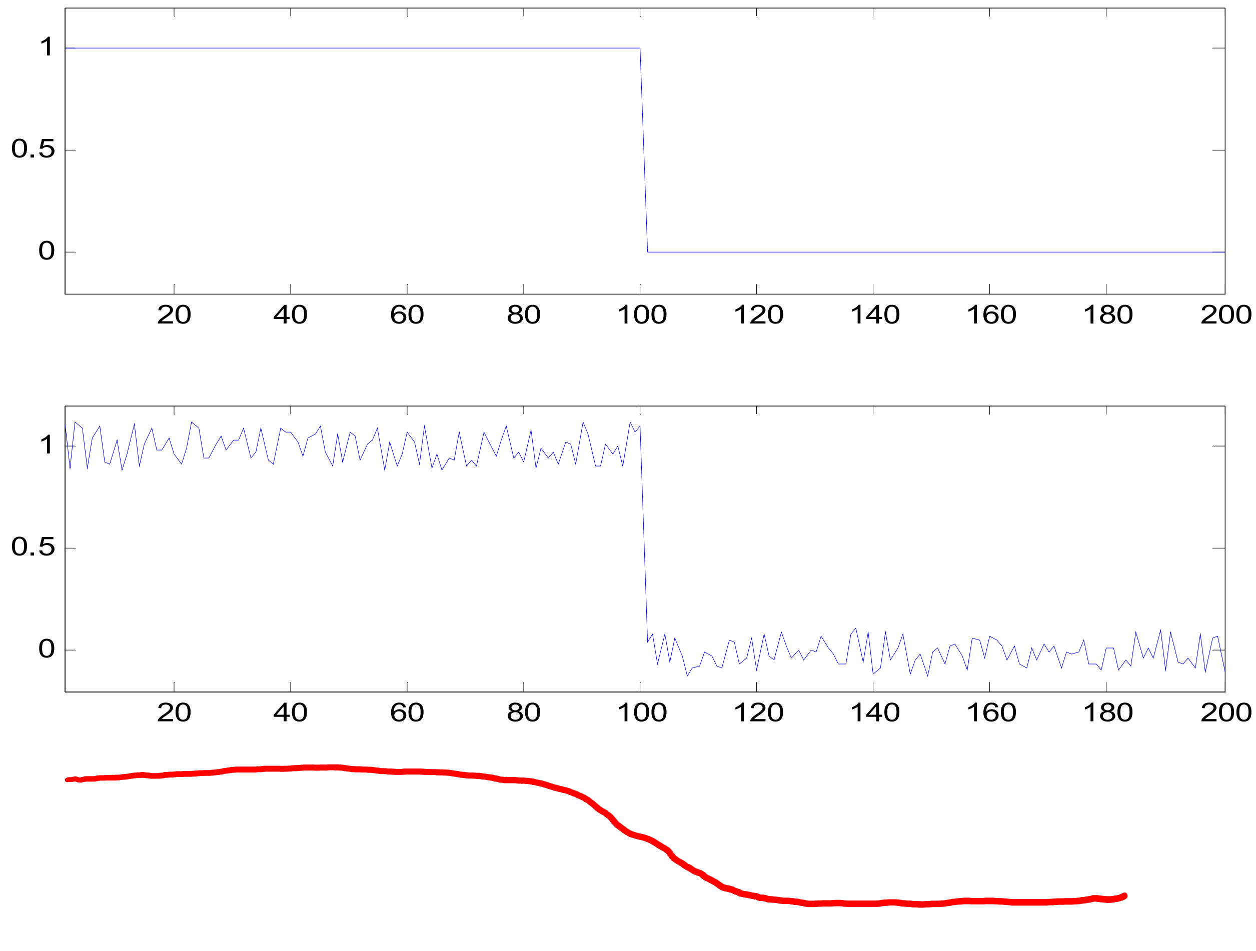
(Using the wavelet features)

(b) Computation Time 和縮小的 pattern 互相比較 (節省運算)

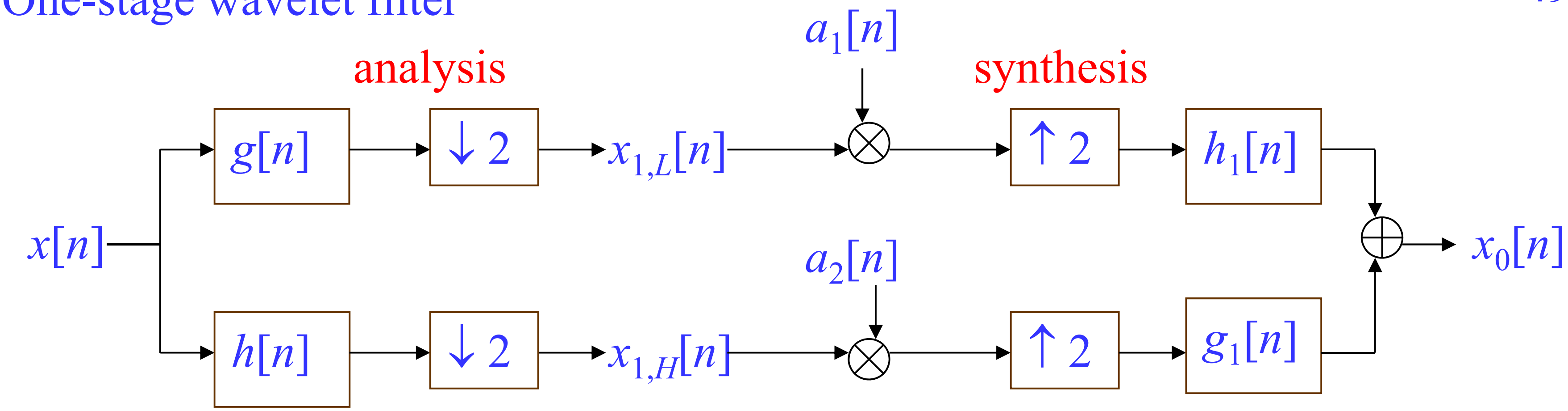
(4) 強調前景，壓縮背景

(5) Filter Design

如何不傷到 edge，又能夠將 noise 去除掉？



One-stage wavelet filter



做 filter design 時，可以令

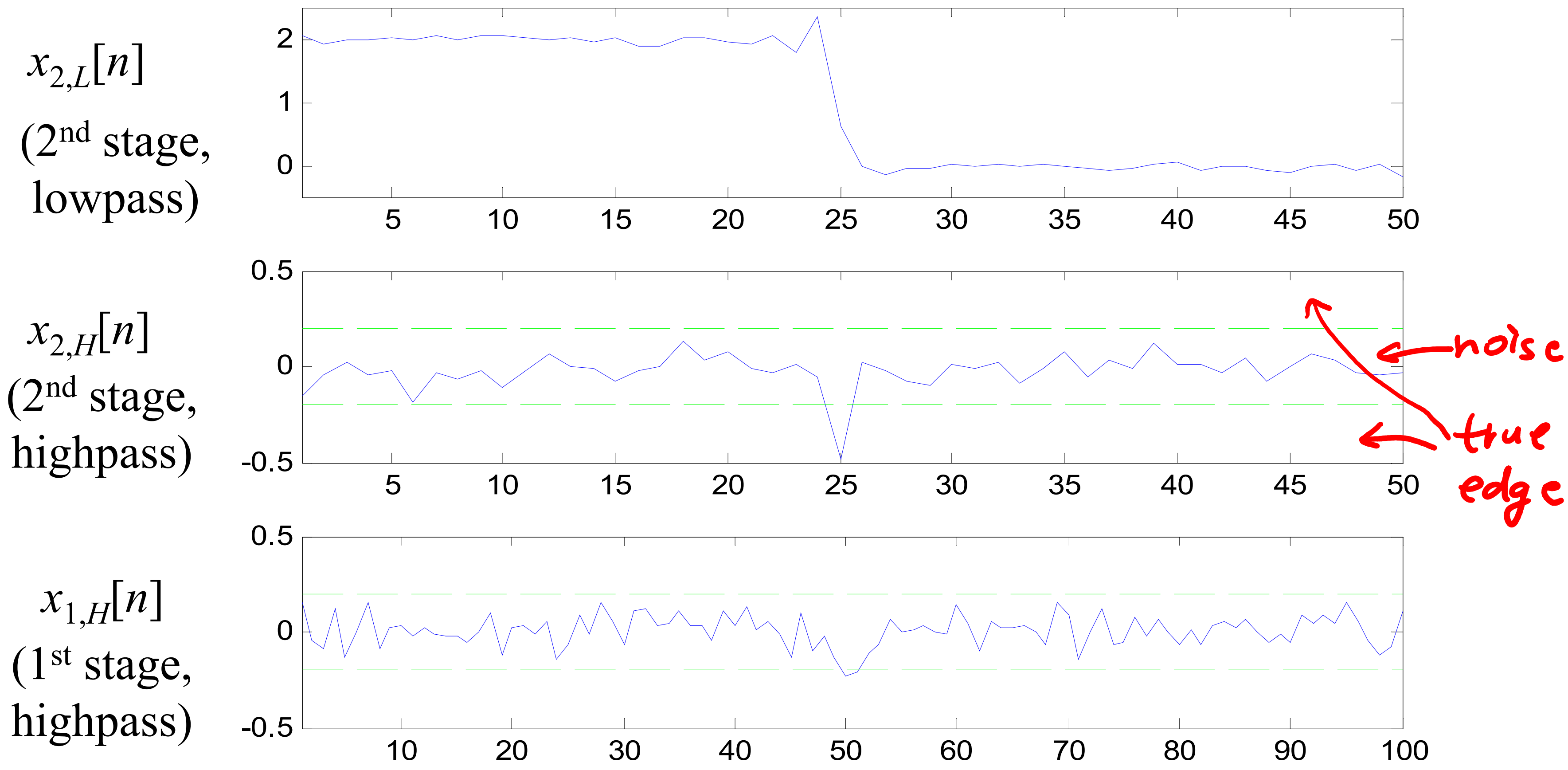
$a_1[n] = 1,$

$a_2[n] = 0$ for non-edge region

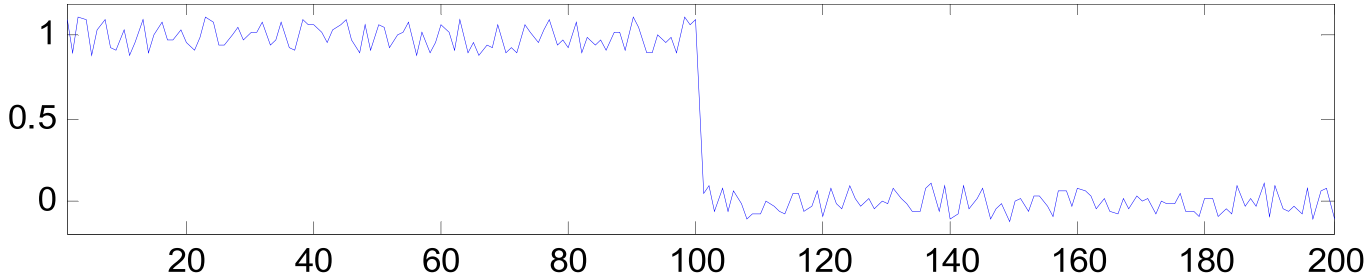
$a_2[n] = 1$ for edge region

以 $x_{1,H}[n]$ 的 amplitude 來區分

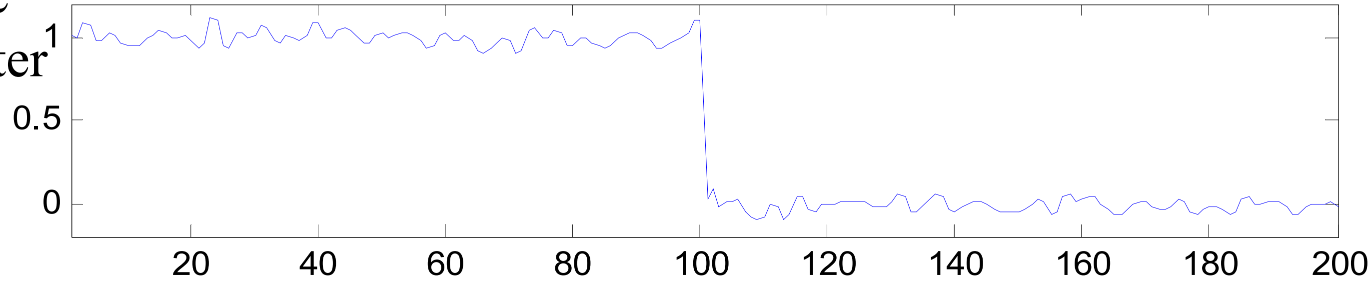
必要時可使用 two-stage 以上的 wavelet filter


$x_{2,L}[n]$
(2nd stage, lowpass)
$x_{2,H}[n]$
(2nd stage, highpass)
$x_{1,H}[n]$
(1st stage, highpass)
noise
true edge

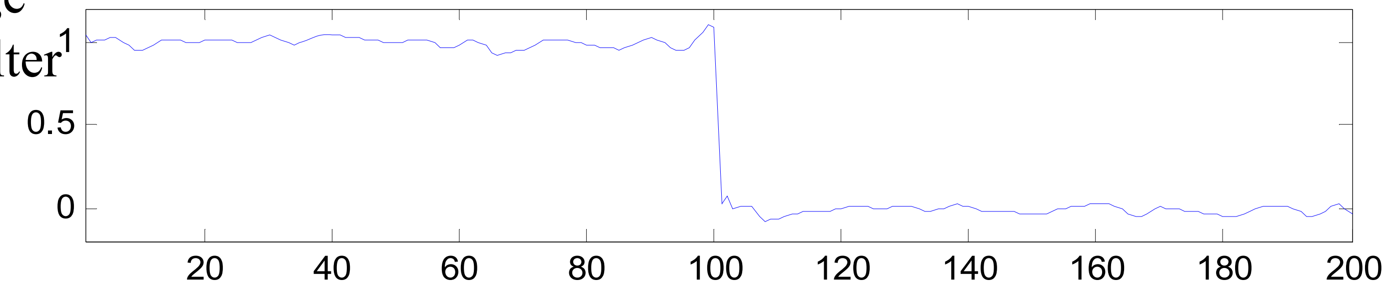
原信號



使用one-stage
的 wavelet filter



使用two-stage
的 wavelet filter



(6) Music

音樂當中，音每高一個音階，頻率就增為二倍

音樂 每一音階有12個半音，增加一個半音，頻率增加 $2^{1/12}$ 倍

(等比級數)

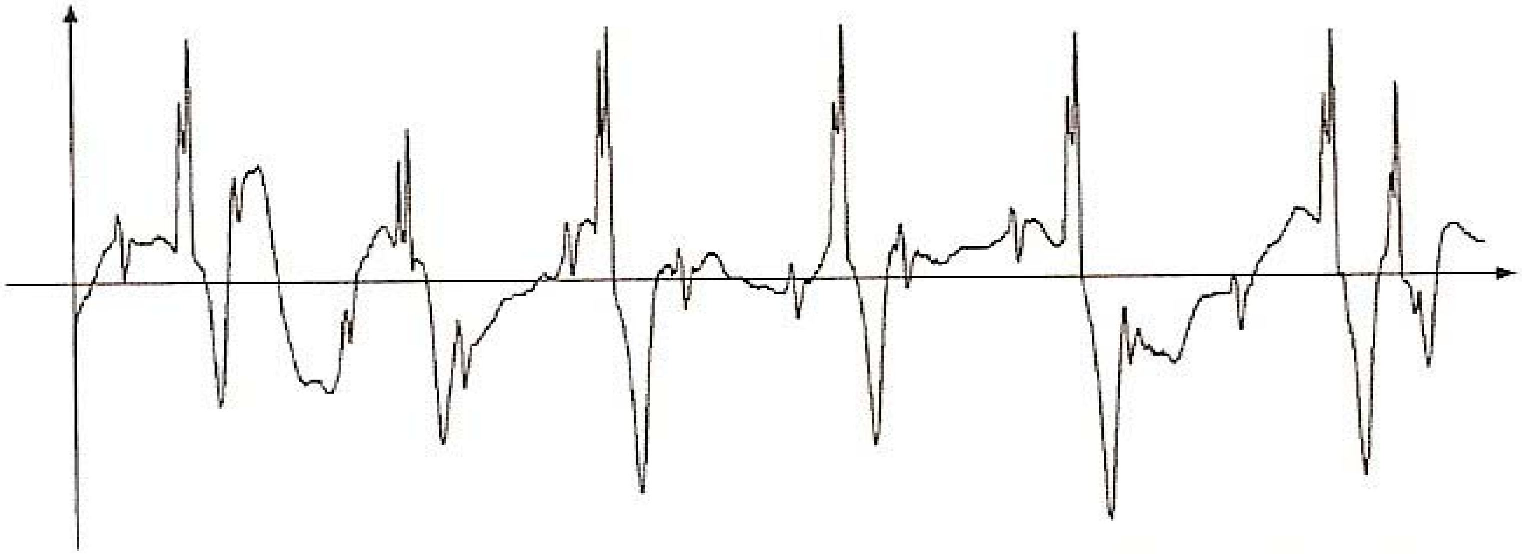
	Do	升Do	Re	升Re	Me	Fa	升Fa	So	升So	La	升La	Si
Hz	270	286	303	321	340	360	382	405	429	454	481	510
Hz	540	572	606	642	680	721	764	810	857	908	962	1019

(7) Acoustics

(8) Analyzing the Electrocardiogram (ECG)

心電圖

- Is the rhythm of the cardiac valve in synchronization with that of the heart muscle?
- Does the heart muscle relax between beats?



From: A. K. Louis, P. Maab, and A. Rieder, "*Wavelets Theory and Applications*", John Wiley & Sons, Chichester, 1997.

(9)「短期因素」和「長期因素」的分析

population

economical data

temperature

(10) 其他奇奇怪怪的應用

指紋的辨識

羊毛質料的辨識

Time-frequency Analysis 和Wavelet 在應用上的異同處

相同：都能夠處理一個信號的頻率分佈會隨時間而改變的情形

不同：Time frequency analysis 對於瞬間頻率的分析比較精確

Wavelet 可作「巨觀」和「微觀」的分析

由於 memory requirement 較少，適合 2D 的
image analysis 和 3D 的 video analysis

附錄十五　希臘字母大小寫與發音一覽表

大寫	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ
小寫	α	β	γ	δ	ε	ζ	η	θ
英文拚法	alpha	beta	gamma	delta	epsilon	zeta	eta	theta
KK 音標	\`ælfə	\`betə	\`gæmə	\`dɛltə	\`ɛpsələn	\`zetə	\`itə	\`θitə

大寫	Ι	Κ	Λ	Μ	Ν	Ξ	Ο	Π
小寫	ι	κ	λ	μ	ν	ξ	ο	π
英文拚法	iota	kappa	lambda	mu	nu	xi	omicron	pi
KK 音標	aɪ\`otə	\`kæpə	\`læmdə	mju	nu	ksɪ	\`ɑmɪkrɑn	paɪ

大寫	Ρ	Σ	Τ	Υ	Φ	Χ	Ψ	Ω
小寫	ρ	σ	τ	υ	φ, ϕ	χ	ψ	ϖ, ω
英文拚法	rho	sigma	tau	upsilon	phi	chi	psi	omega
KK 音標	ro	\`sɪgmə	taʊ	\`jupsəlɑn	faɪ	kaɪ	saɪ	\`omɪgə

附錄十六 Generalization for the Wavelet Transforms

1. Directional Form 2-D Wavelet Transforms

一般的 2-D wavelet transform，其實可分解成沿著 x-axis 以及沿著 y-axis 的 1-D wavelet transforms 的組合

其實，2-D wavelet transform 不一定要沿著 x-axis ， y-axis 來做

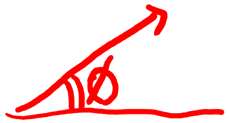
Directional 2-D wavelet transforms:

- curvelet
- contourlet
- bandlet
- shearlet
- Fresnelet
- wedgelet
- brushlet

- **Curvelet** (ridgelet)

$$F_w(a,b,\phi)=\frac{1}{\sqrt{b}}\int_{-\infty}^{\infty} f(r\cos\phi, r\sin\phi)\,\psi\left(\frac{r-a}{b}\right)dr$$

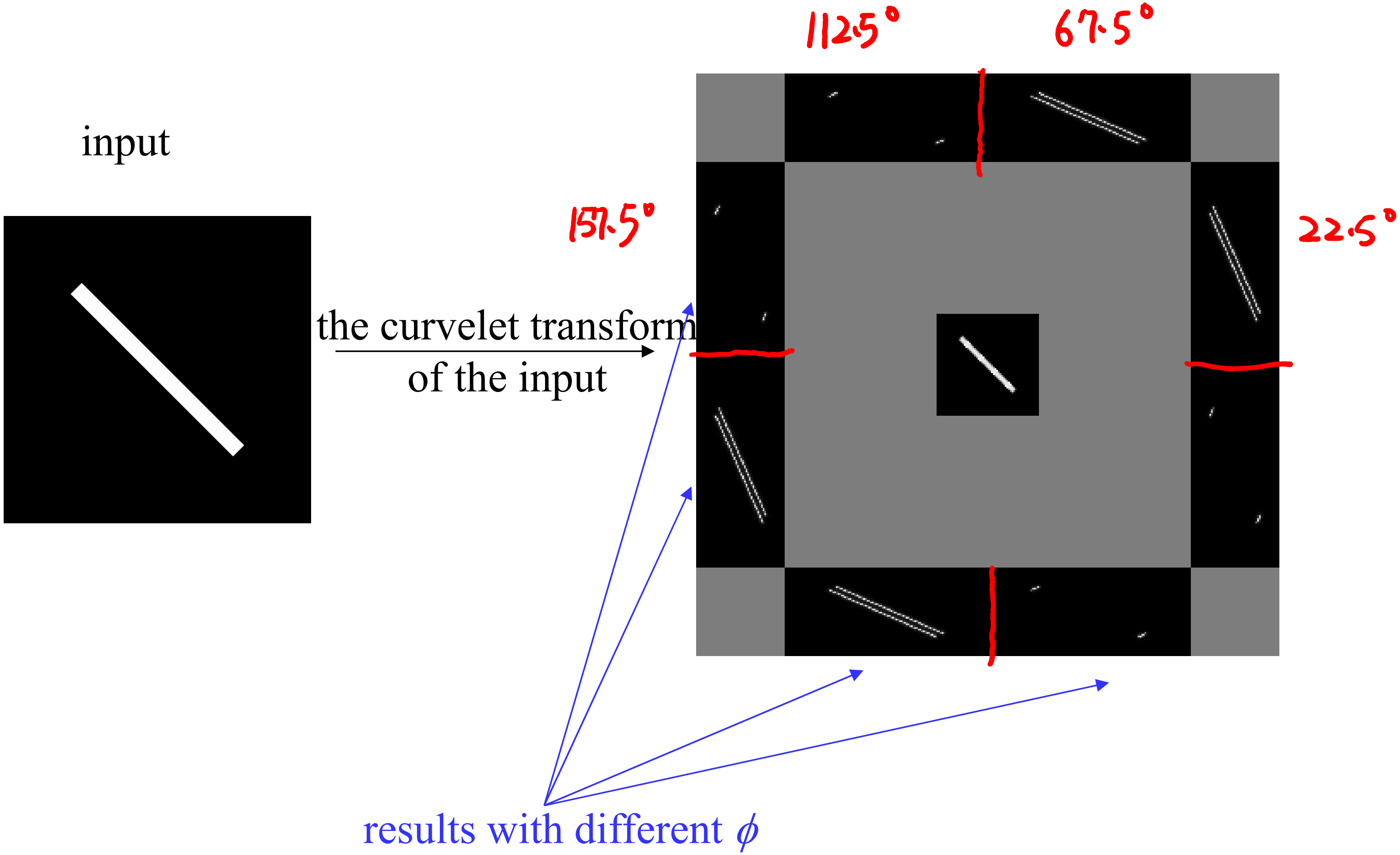
rotation



比較：原本的 1-D wavelet

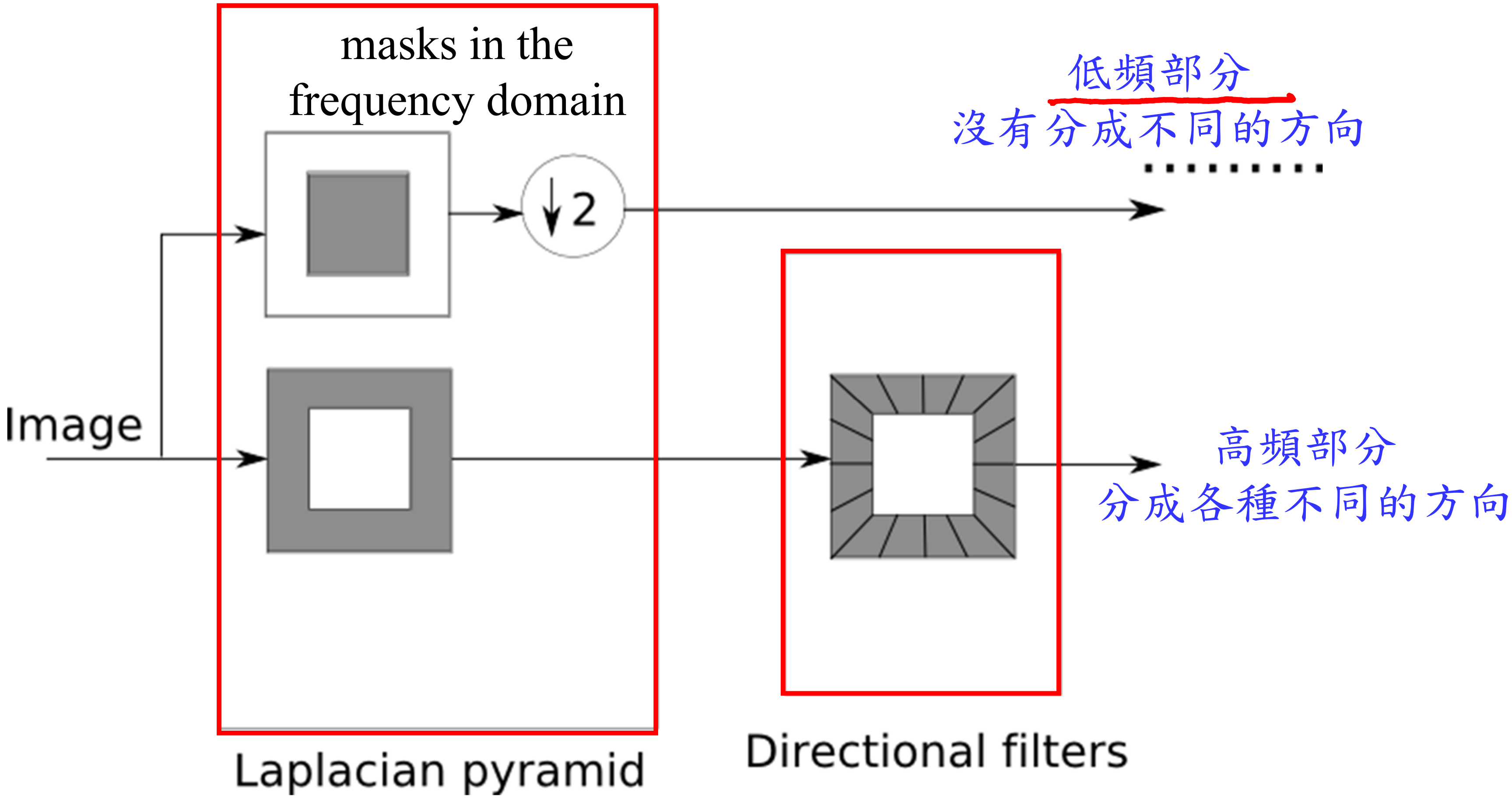
$$F_w(a,b)=\frac{1}{\sqrt{b}}\int_{-\infty}^{\infty} f(x)\,\psi\left(\frac{x-a}{b}\right)dx$$

E. Candès and D. Donoho, "Curvelets – a surprisingly effective nonadaptive representation for objects with edges." In: A. Cohen, C. Rabut and L. Schumaker, Editors, *Curves and Surface Fitting*: Saint-Malo 1999, Vanderbilt University Press, Nashville (2000), pp. 105–120.



(four direction for the high-frequency part)

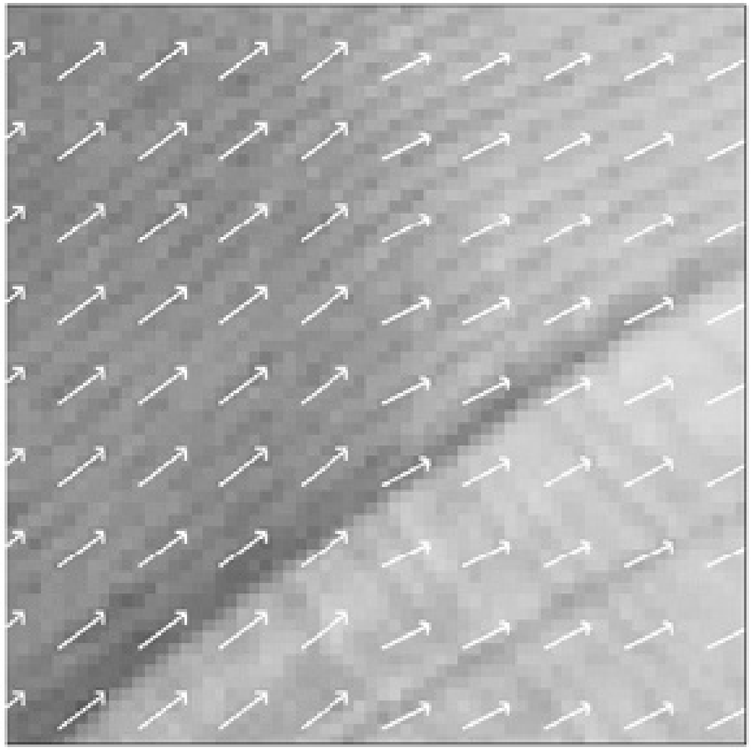
- **Contourlet**

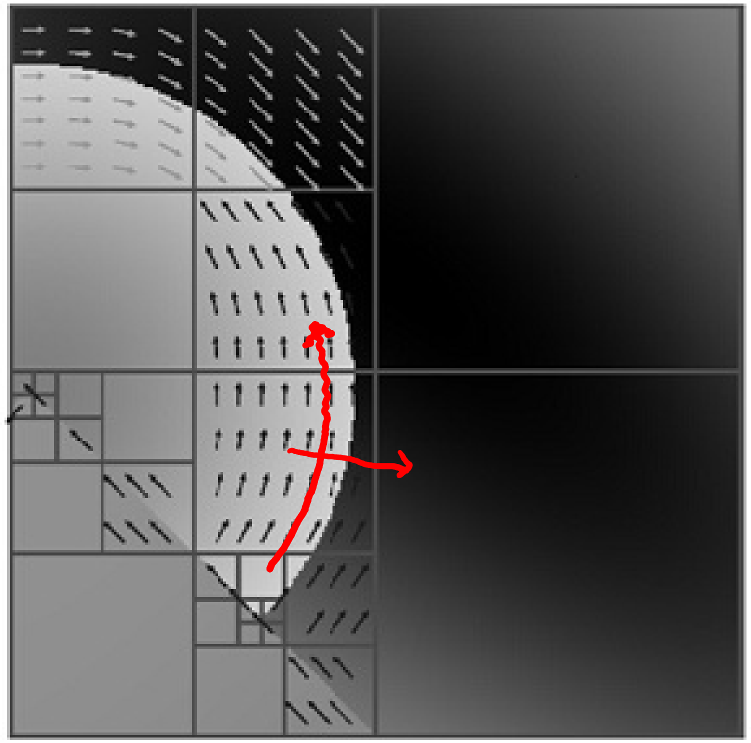


M. Do and M. Vetterli, "The contourlet transform: An efficient directional multiresolution image representation," *IEEE Trans. Image Processing*, vol.14, no.12, pp.2091–2106, Dec. 2005.

• Bandlet

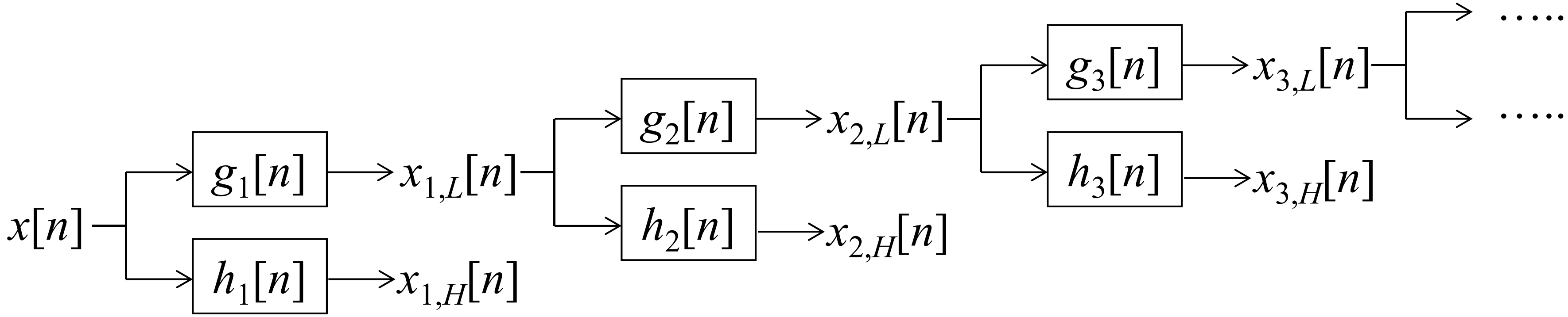
根據物體的紋理或邊界，來調整 wavelet transforms 的方向





Stephane Mallet and Gabriel Peyre, "A review of Bandlet methods for geometrical image representation," *Numerical Algorithms*, Apr. 2002.

2. Stationary Wavelet Transforms



其中

Q: 和原本 discrete wavelet transform 不一樣的地方在哪裡？

G. P. Nason and B. W. Silverman, “The stationary wavelet transform and some statistical applications,” *Lecture Notes in Statistics*, available in http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.49.2662&rep=rep1&type=pdf

3. Bandwidth Form Wavelet Transforms

A little modification for $g[n]$ and $h[n]$

4. Multi-Band Wavelet Transforms

Instead of only two outputs

Happy New Year!

祝各位期末考順利，寒假愉快！