

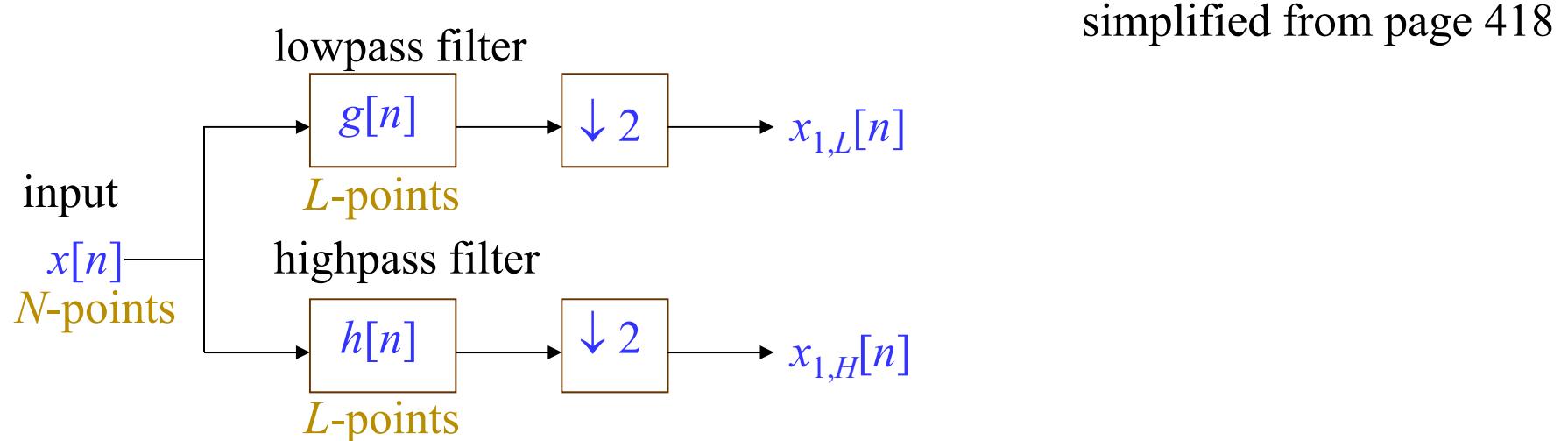
XIV. Discrete Wavelet Transform (DWT) ⁴⁵⁵

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 415)
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 (1D DWT)

456



$\downarrow 2$: downsampling by the factor of 2

$$x[n] \rightarrow \downarrow Q \rightarrow z[n] \quad z[n] = x[Qn]$$

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

直接以 $x[n]$ 作為 initial

Low pass filter $g[n]$

角色似 scaling function

(相當於 page 415 的 g_n)

High pass filter $h[n]$

角色似 wavelet function

(相當於 page 415 的 h_n)

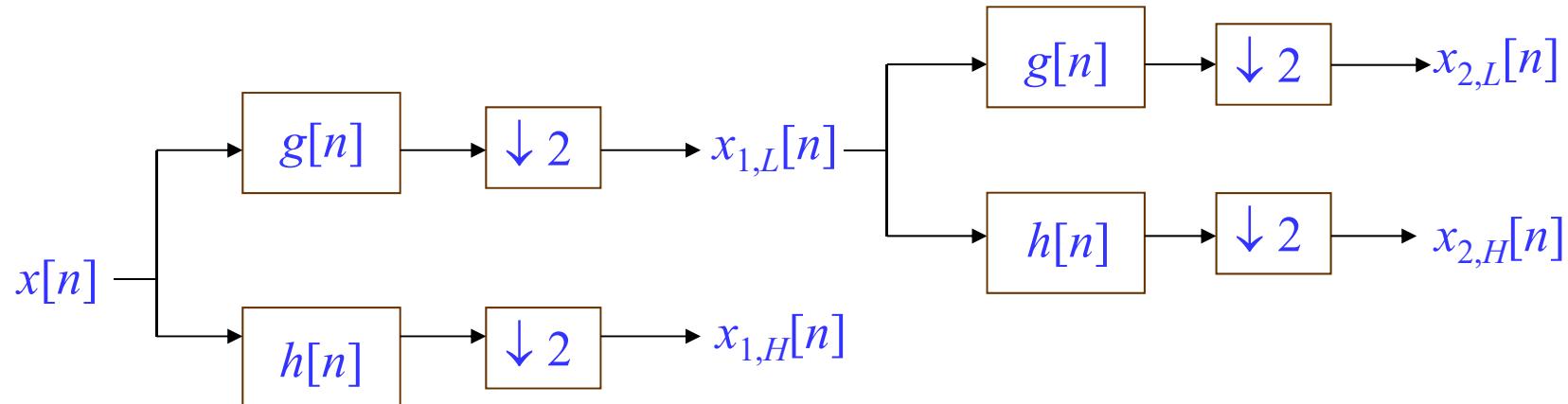
$$1^{\text{st}} \text{ stage} \quad x_{1,L}[n] = \sum_{k=0}^{K-1} x[2n-k]g[k]$$

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

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

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

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



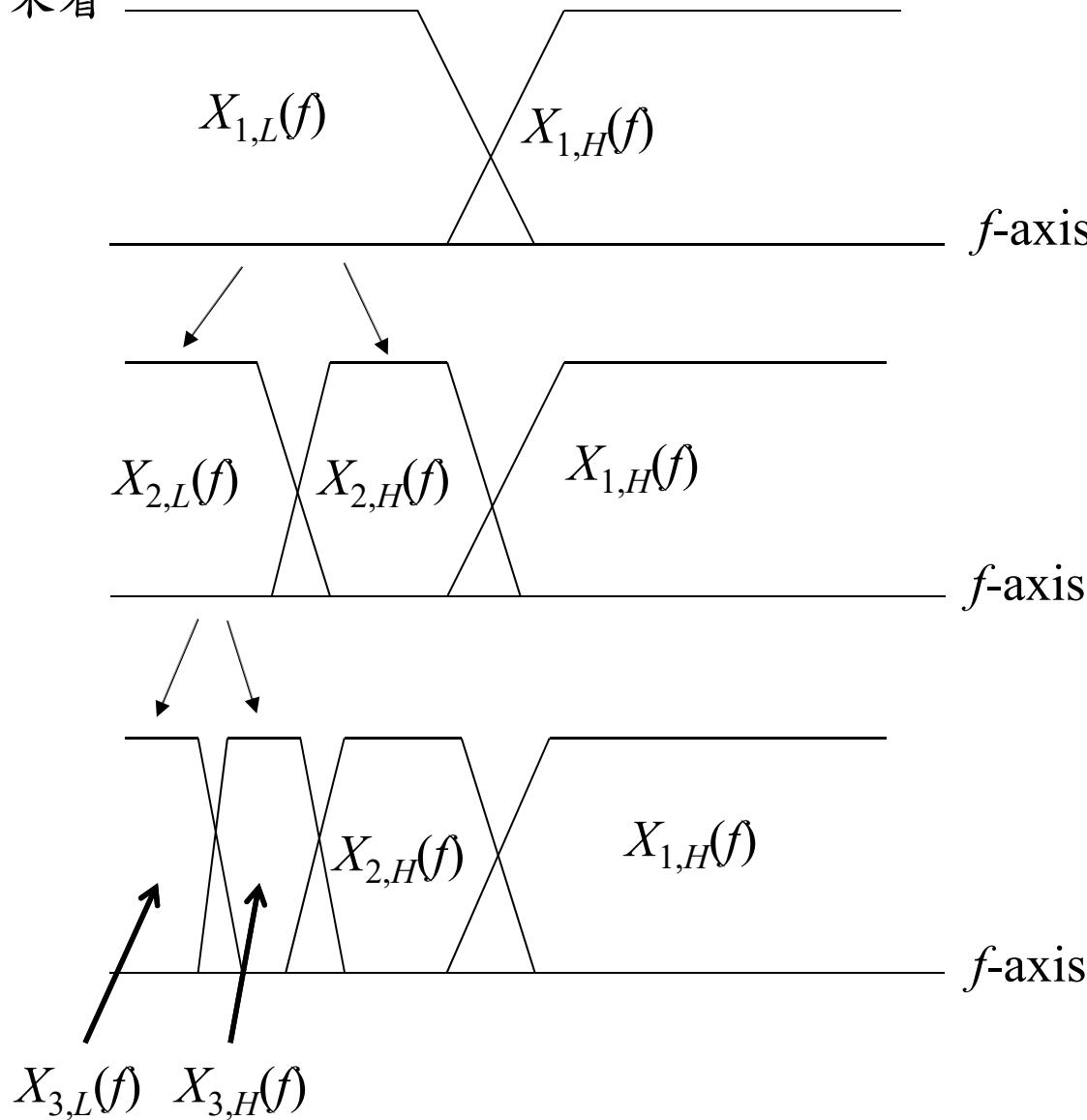
(1) 有的時候，對於 $x_{a,H}[n]$ 也再作細分

(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 $N/2^a$

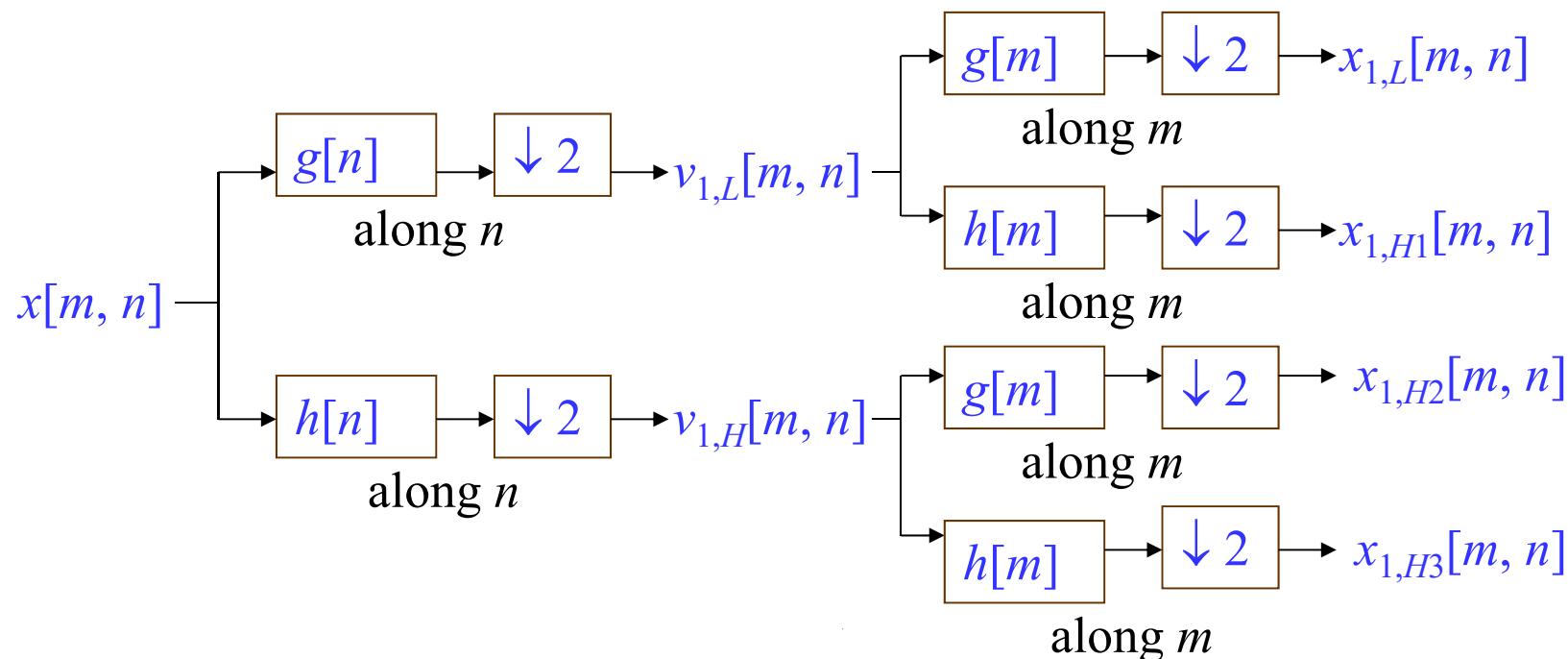
(3) 經過 DWT 之後，全部點數仍接近 N 點

(4) 以頻譜來看



14.3 2-D Discrete Wavelet Transform (2D DWT)

461



輸入 : $x[m, n]$

Low pass filter $g[n]$

High pass filter $h[n]$

- along n

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

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

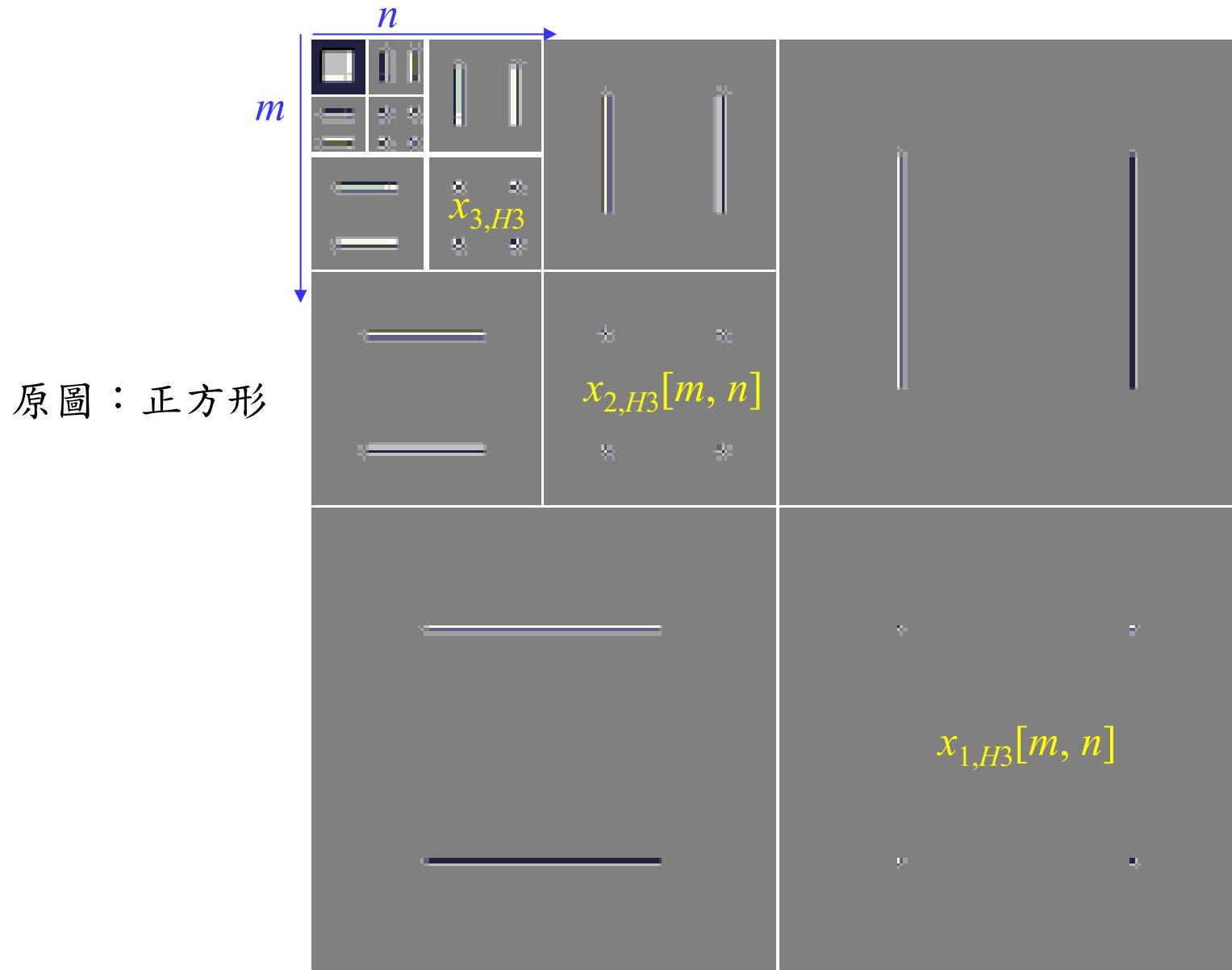
- along m

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

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

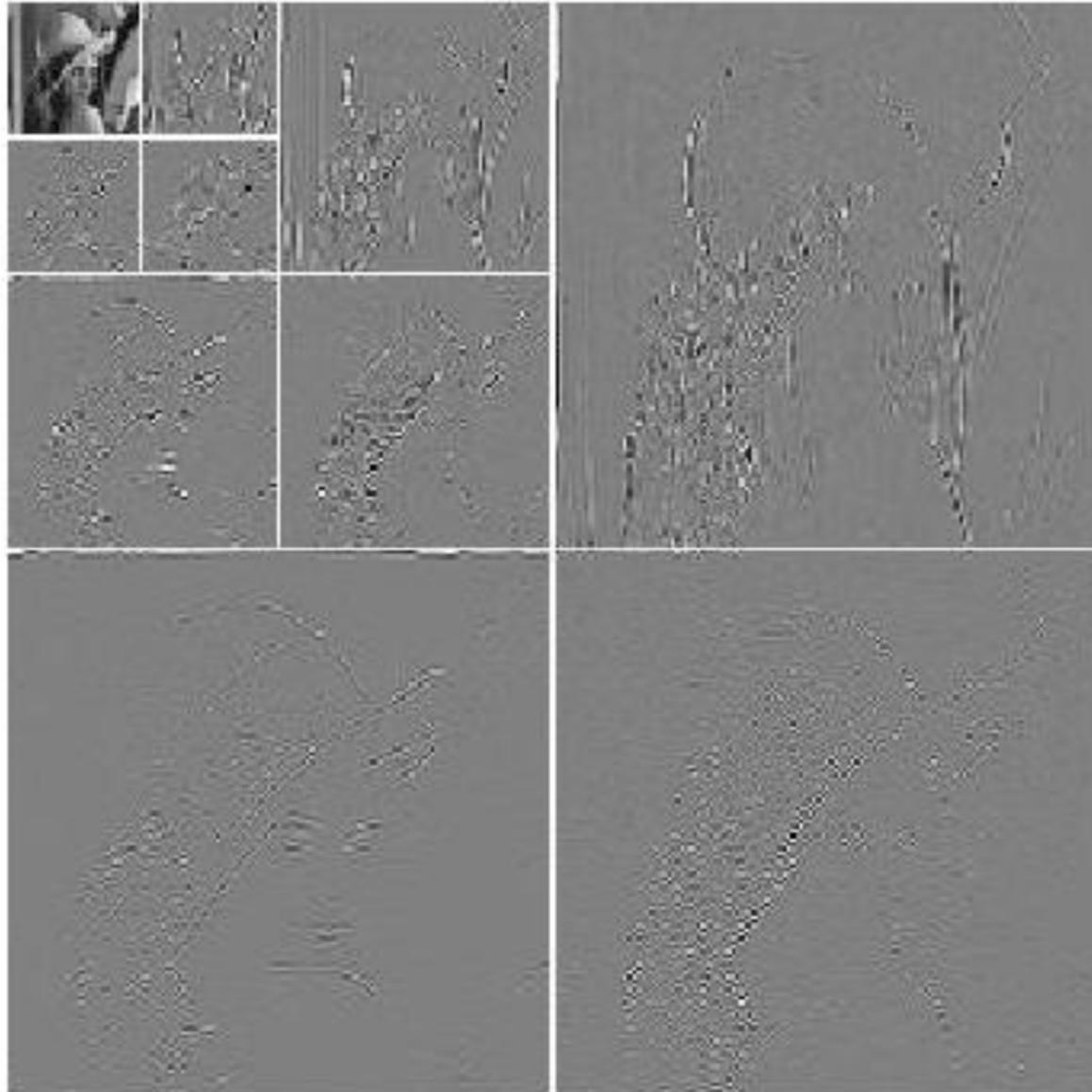
$$x_{1,H_1}[m, n] = \sum_{k=0}^{K-1} v_{1,L}[2m - k, n] h[k]$$

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



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

原圖 : Lena



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

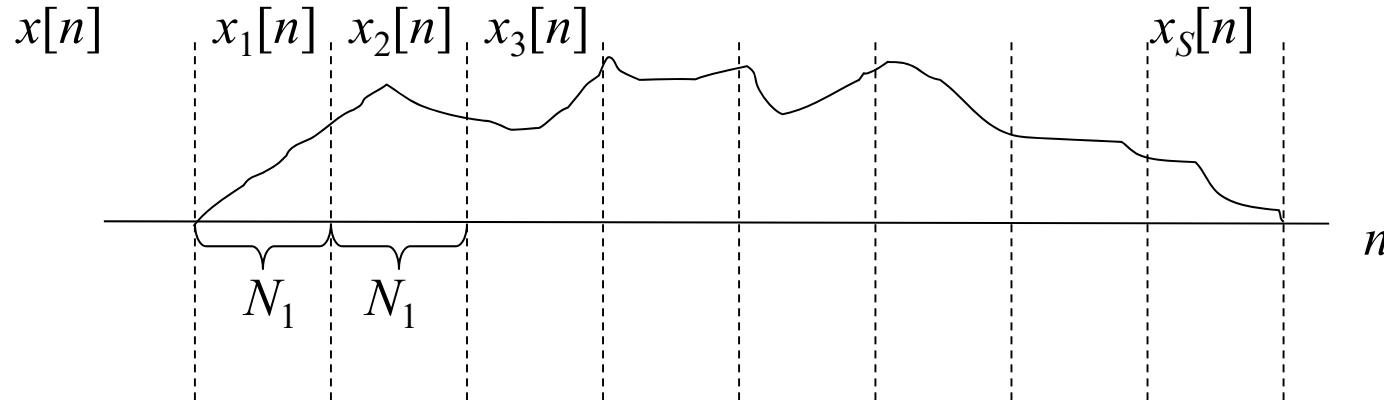
$x[n] * y[n]$, $\text{length}(x[n]) = N$, $\text{length}(y[n]) = L$,

$$\overbrace{\quad \quad \quad}^{\begin{array}{c} IDFT_{N+L-1} \left[DFT_{N+L-1}(x[n]) DFT_{N+L-1}(y[n]) \right] \\ \uparrow \\ (N+L-1)\text{-point discrete Fourier transform (DFT)} \end{array}} \\ \downarrow \\ (N+L-1)\text{-point inverse discrete Fourier transform (IDFT)}$$

(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 $N \ggg L$ 時，使用 “sectioned convolution” 的技巧



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

總共有 $S = N / N_1$ 段

$$x[n] = x_1[n] + x_2[n] + \dots + x_S[n]$$

$$x[n] * g[n] = x_1[n] * g[n] + x_2[n] * g[n] + \dots + x_S[n] * g[n]$$

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

complexity:

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

- 重要概念：

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

$$O(N)$$

when $N \ggg L$

(3) Multiple stages 的情形下

- 若 $x_{a,H}[n]$ 不再分解

$$\begin{aligned} \text{Complexity 近似於: } & \left(N + \frac{N}{2} + \frac{N}{4} + \frac{N}{8} + \dots + 2 \right) \log_2 N_1 \\ & = (2N - 2) \log_2 N_1 \approx 2N \log_2 N_1 \end{aligned}$$

- 若 $x_{a,H}[n]$ 也細分

Complexity 近似於:

$$\begin{aligned} & \left(N + 2\frac{N}{2} + 4\frac{N}{4} + 8\frac{N}{8} + \dots + \frac{N}{2} \cdot 2 \right) \log_2 N_1 \\ & = (N \log_2 N) \log_2 N_1 \\ & \quad (\text{和 DFT 相近}) \end{aligned}$$

(4) Complexity of the 2-D DWT on page 461 (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

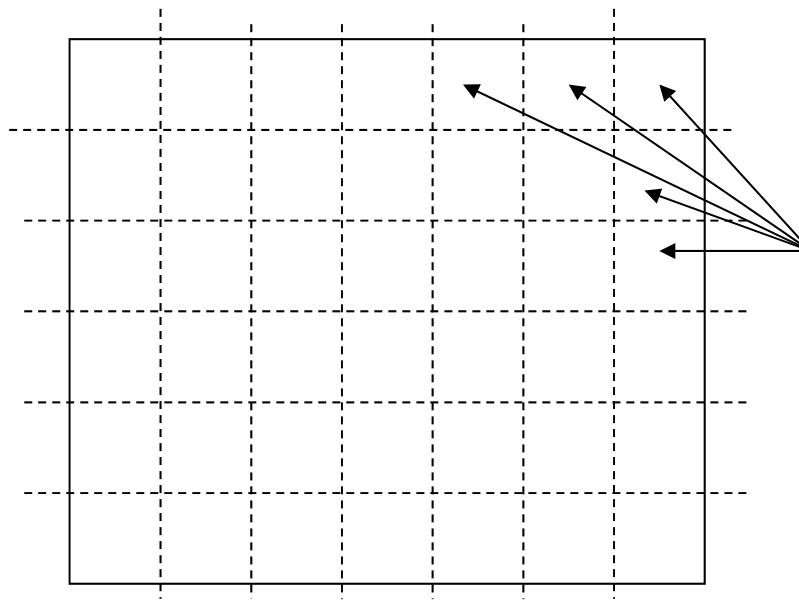
The second part needs $N+L-1$ 1-D DWTs and
the input for each 1-D DWT has M points

$$\text{complexity} \approx MN \log_2 N + MN \log_2 M$$

$$= MN(\log_2 N + \log_2 M)$$

$$= MN \log_2(MN)$$

Image



The original size: $M \times N$

The size of each part: $M_1 \times N_1$

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

• 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

$$\left(MN + \frac{MN}{4} + \frac{MN}{16} + \dots \right) \log_2(M_1 N_1) \approx \frac{4}{3} MN \log_2(M_1 N_1)$$

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

total complexity

$$\begin{aligned} & \left(MN + 4 \frac{M}{2} \frac{N}{2} + 16 \frac{M}{4} \frac{N}{4} + \dots \right) \log_2(M_1 N_1) \\ &= [MN \log_2(\min(M, N))] \log_2(M_1 N_1) \end{aligned}$$

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 $N \log_2 N$, $M \log_2(MN)$ 的觀念不同)

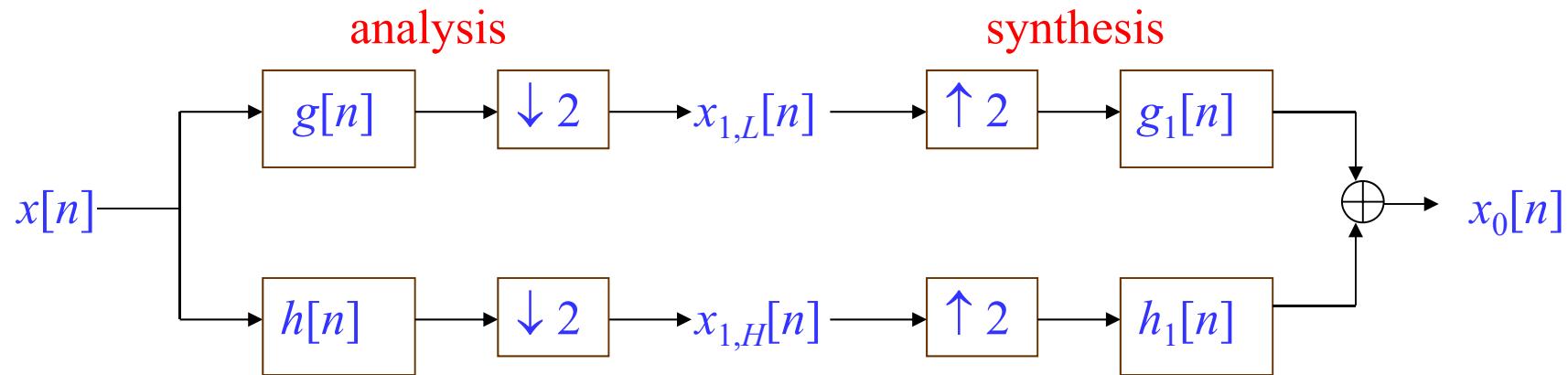
很重要的概念

- Note : DCT 的 complexity 也是 linear with MN

(divided into 8×8 blocks)

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

14.6 Reconstruction



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



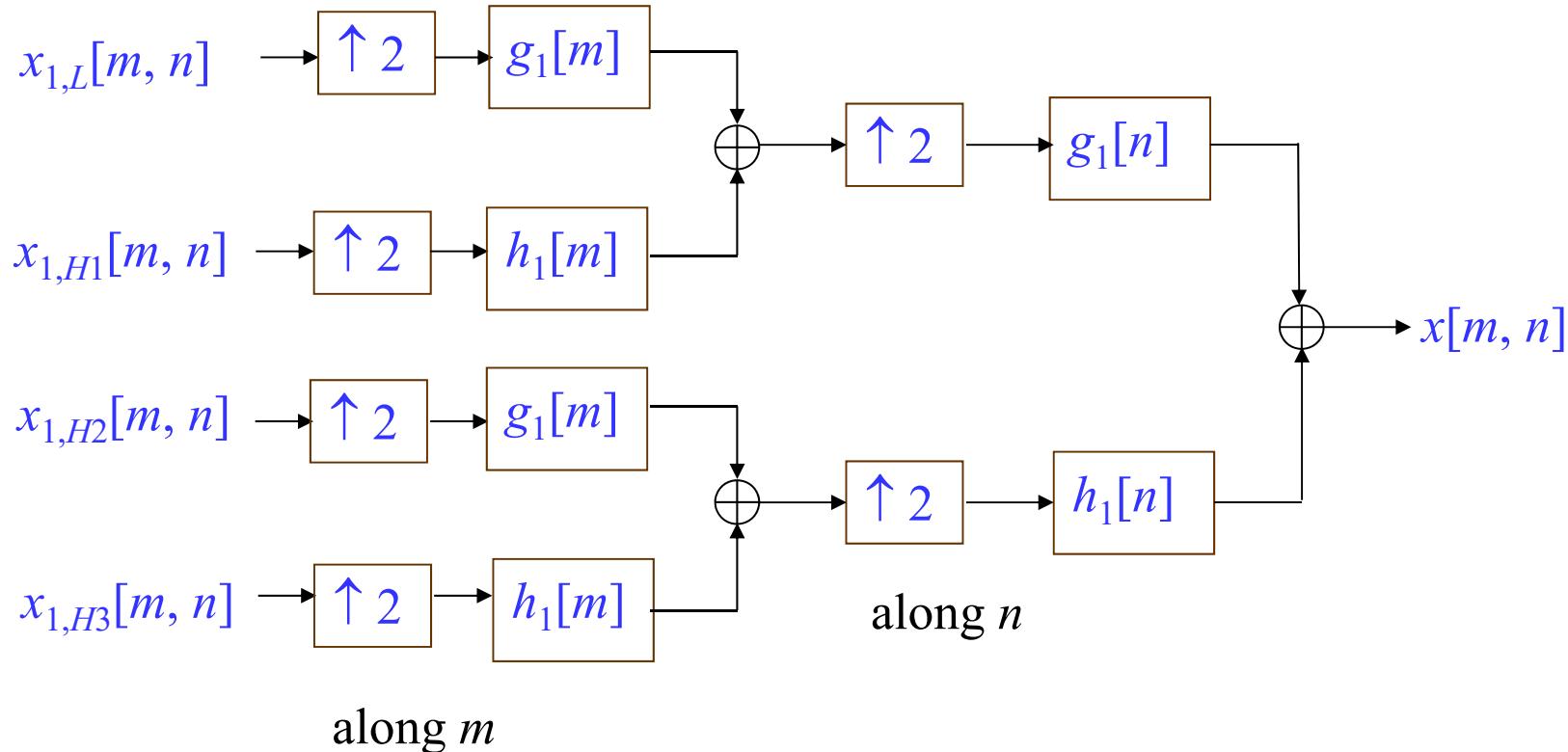
: upsampling by the factor of 2

$$a[n] \xrightarrow{\uparrow Q} b[n] \quad b[Qn] = a[n]$$

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

the analysis part of the 2D DWT: page 461

the synthesis part of the 2D DWT



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

- If $a[n] = b[2n]$, $\xrightarrow{\quad}$ $A(z) = \frac{1}{2} [B(z^{1/2}) + B(-z^{1/2})]$
 $\downarrow 2$ (downsampling)

(Proof):

$$\begin{aligned} B(z^{1/2}) + B(-z^{1/2}) &= \sum_{n=-\infty}^{\infty} b[n]z^{-n/2} + \sum_{n=-\infty}^{\infty} (-1)^n b[n]z^{-n/2} \\ &= \sum_{n=-\infty}^{\infty} (1 + (-1)^n) b[n]z^{-n/2} = 2 \sum_{n_1=-\infty}^{\infty} b[2n_1]z^{-n_1} = 2 \sum_{n_1=-\infty}^{\infty} a[n_1]z^{-n_1} = A(z) \end{aligned}$$

- If $a[2n] = b[n]$, $\xrightarrow{\quad}$ $A(z) = B(z^2)$

$$a[2n+1] = 0$$

$\uparrow 2$ (upsampling)

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

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

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

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

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

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

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

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

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

$$\text{where } \mathbf{H}_m(z) = \begin{bmatrix} G(z) & H(z) \\ G(-z) & H(-z) \end{bmatrix}$$

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

where

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

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

if and only if

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

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

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

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

這四個條件被稱作
biorthogonal conditions

(Proof)

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

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

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

Therefore,

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

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

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

\downarrow inverse Z transform

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

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



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

(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

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

(d) Since $G(z)H_1(z) + G(-z)H_1(-z)$

$$\begin{aligned} &= -G(z) \frac{G(-z)}{\det(\mathbf{H}_m(z))} - G(-z) \frac{G(z)}{\det(\mathbf{H}_m(-z))} \\ &= -\frac{G(z)G(-z)}{\det(\mathbf{H}_m(z))} + \frac{G(-z)G(z)}{\det(\mathbf{H}_m(z))} = 0 \end{aligned}$$

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

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

← orthogonality 條件 3

(e) 同理 $G_1(z)H(z) + G_1(-z)H(-z) = 0$

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

← orthogonality 條件 4

- Reconstruction
- Finite length 為了 implementation 速度的考量

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

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

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

令 $\det(\mathbf{H}_m(z)) = \alpha z^k$ 則根據 page 479,

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

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

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

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

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

k 必需為 odd

- Lowpass-highpass pair

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

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

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

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

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

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

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

(4) $h[n]$ 為 highpass filter

$(\text{lowpass and highpass pair})$

第三個條件較難達成，是設計的核心

14.10 Two Types of Perfect Reconstruction Filters

488

(1) QMF (quadrature mirror filter)

$$G(z) \quad \text{satisfy} \quad G^2(z) - G^2(-z) = 2z^k \quad k \text{ is odd}$$

$g[n]$ has finite length

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

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

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

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

(2) Orthonormal

$$G(z) \quad \text{satisfy} \quad G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 2$$

$g[n]$ has finite length

$$H(z) = -z^k G(z^{-1}) \quad \underline{k \text{ is odd}} \quad h[n] = (-1)^n g[-n-k]$$

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

$$H_1(z) = -z^{-k} G(-z) = H(z^{-1}) \quad h_1[n] = h[-n]$$

$$\begin{aligned} \det(\mathbf{H}_m(z)) &= G(z)H(-z) - H(z)G(-z) \\ &= G(z)z^k G(z^{-1}) + G(-z)z^k G(-z^{-1}) = 2z^k \end{aligned}$$

大部分的 wavelet 屬於 orthonormal wavelet

For the orthonormal wavelet

$$\sum_{n=0}^{N-\tau-1} g[n]g[n+\tau] = 0 \quad \text{for } \tau = 2, 4, \dots, N-2$$

$$\sum_{n=0}^{N-\tau-1} h[n]h[n+\tau] = 0$$

(orthonormal to the shift versions of themselves)

It can be proved by pages 481 and 489.

(Note): 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

- discrete Haar wavelet (最簡單的)

$$g[-1] = g[0] = 1 \quad g[n] = 0 \quad \text{otherwise}$$

$$h[-1] = -1, \quad h[0] = 1 \quad h[n] = 0 \quad \text{otherwise}$$

$$g_1[0] = g_1[1] = 1 \quad g_1[n] = 0 \quad \text{otherwise}$$

$$h_1[0] = 1, \quad h_1[1] = -1 \quad h_1[n] = 0 \quad \text{otherwise}$$

是一種 orthonormal filter

- discrete Daubechies wavelet (8-point case)

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

$$n = 0 \sim 7 \quad g[n] = 0 \quad \text{otherwise}$$

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

$$n = 0 \sim 7 \quad h[n] = 0 \quad \text{otherwise}$$

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

$$n = -7 \sim 0 \quad g_1[n] = 0 \quad \text{otherwise}$$

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

$$n = -7 \sim 0 \quad h_1[n] = 0 \quad \text{otherwise}$$

- 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)

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

- discrete Daubechies wavelet (10-point case)

$$\begin{aligned} g[n] = & [0.0033 \quad -0.0126 \quad -0.0062 \quad 0.0776 \quad -0.0322 \quad -0.2423 \\ & 0.1384 \quad 0.7243 \quad 0.6038 \quad 0.1601] \end{aligned}$$

- discrete Daubechies wavelet (12-point case)

$$\begin{aligned} g[n] = & [-0.0011 \quad 0.0048 \quad 0.0006 \quad -0.0316 \quad 0.0275 \quad 0.0975 \\ & -0.1298 \quad -0.2263 \quad 0.3153 \quad 0.7511 \quad 0.4946 \quad 0.1115] \end{aligned}$$

symlet (6-point case)

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

symlet (8-point case)

$$\begin{aligned} g[n] = & [-0.0757 \quad -0.0296 \quad 0.4976 \quad 0.8037 \quad 0.2978 \quad -0.0992 \\ & -0.0126 \quad 0.0322] \end{aligned}$$

symlet (10-point case)

$$\begin{aligned} g[n] = & [0.0273 \quad 0.0295 \quad -0.0391 \quad 0.1993 \quad 0.7234 \quad 0.6339 \\ & 0.0166 \quad -0.1753 \quad -0.0211 \quad 0.0195] \end{aligned}$$

Daubechies wavelets and symlets are defined for N is a multiple of 2

coiflet (6-point case)

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

coiflet (12-point case)

$$\begin{aligned} g[n] = & [0.0232 \quad -0.0586 \quad -0.0953 \quad 0.5460 \quad 1.1494 \quad 0.5897 \\ & -0.1082 \quad -0.0841 \quad 0.0335 \quad 0.0079 \quad -0.0026 \quad -0.0010] \end{aligned}$$

Coiflets are defined for N is a multiple of 6

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 vanishing moment of p .

- Symlet

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

- Coiflet

The ? point coiflet has the vanishing moment of p .

The scaling function also has the vanishing moment.

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

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 $z^k P_1(z)$ 的根 (i.e., $z^k P_1(z) = 0$ 的地方)

Q: 在 Matlab 當中應該用什麼指令

(When $p = 2, \text{roots} = 3.7321, 0.2679$)

Step 4 算出

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

z_1, z_2, \dots, z_{p-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 $p = 2$, $g_0[n] = [1 \quad 1.7321 \quad 0.4641 \quad -0.2679]$)

$$n = -3 \sim 0$$

Step 6 Normalization

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

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

$$n = -3 \sim 0$$

Step 7 Time reverse

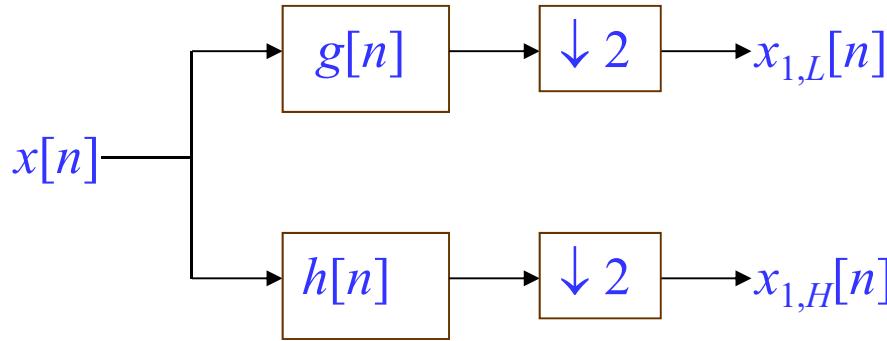
$$g[n] = g_1[-n] \quad h[n] = (-1)^n g[2p-1-n]$$

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

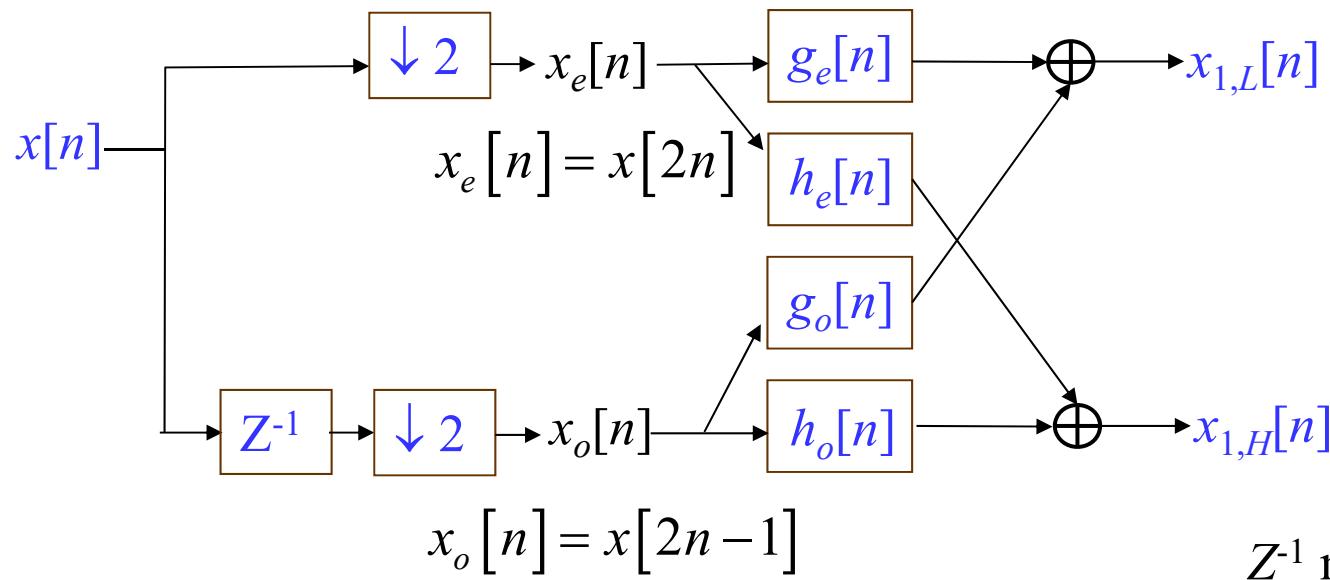
14.13 2x2 Structure Form and the Lifting Scheme

500

The analysis part



can be changed into the following **2x2 structure**



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

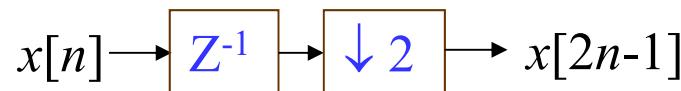
(Proof): From page 457,

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

$$\begin{aligned} x_{1,L}[n] &= \sum_{k=0}^{K/2-1} x[2n-2k]g[2k] + \sum_{k=0}^{K/2-1} x[2n-2k-1]g[2k+1] \\ &= \sum_{k=0}^{K/2-1} x_e[n-k]g_e[k] + \sum_{k=0}^{K/2-1} x_o[n-k]g_o[k] \end{aligned}$$

where

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



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:

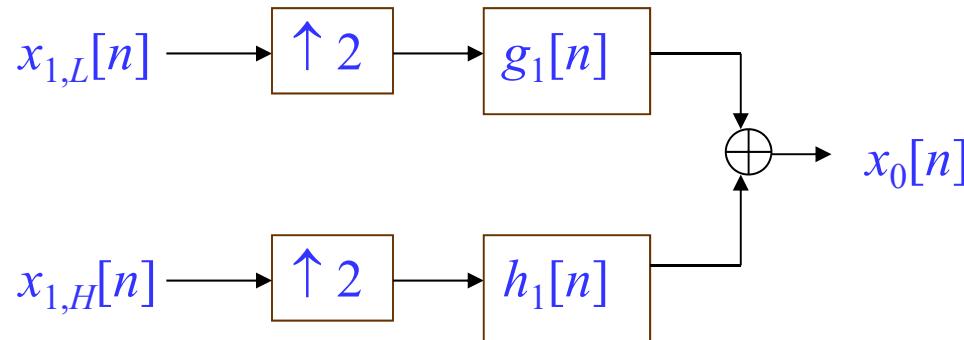
Two Convolutions of an N -length input and an L -length filter

New Structure:

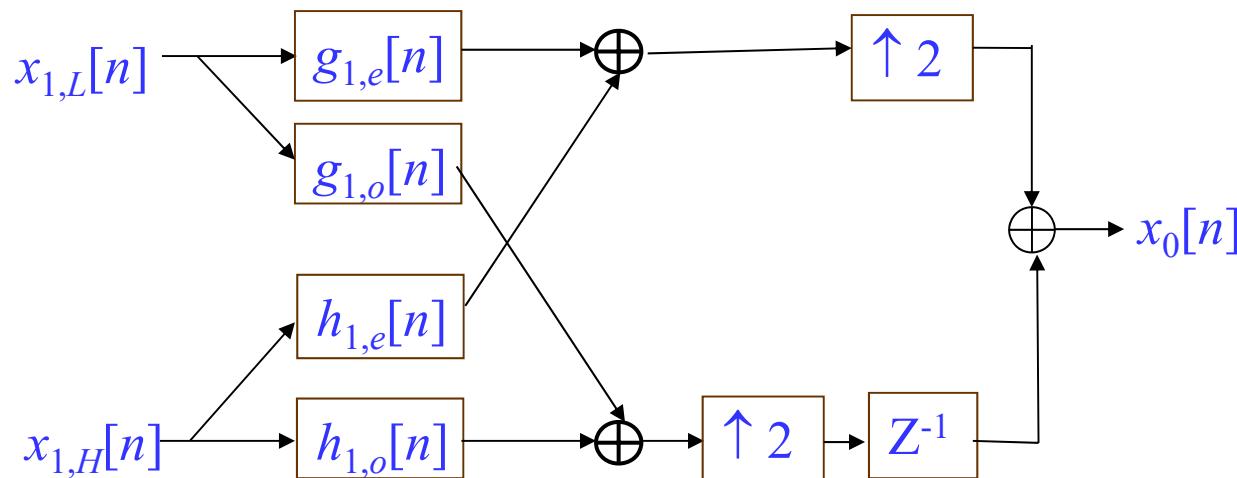
Four Convolutions of an $(N/2)$ -length input and an $(L/2)$ -length filter, which is more efficient. (Why?)

Similarly, the synthesis part

503



can be changed into the following **2x2** structure



where

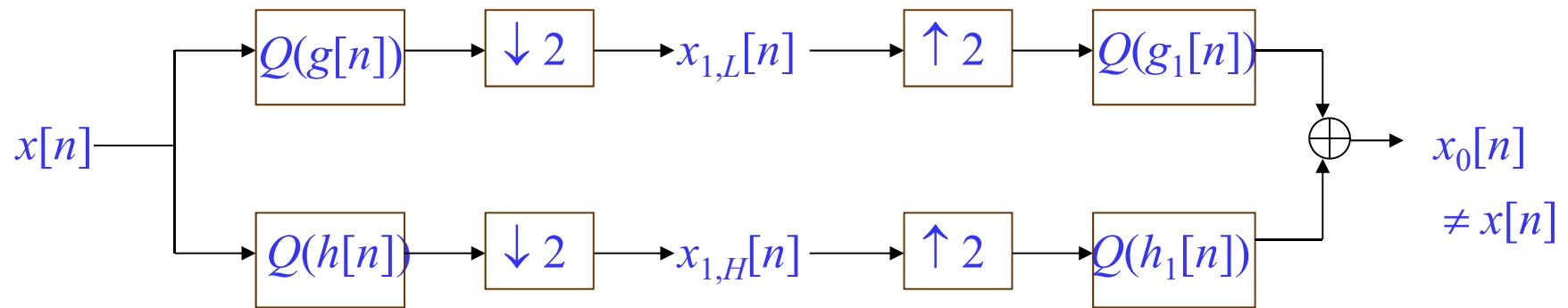
$$g_{1,e}[n] = g_1[2n]$$

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

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

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

After performing quantization, the DWT may not be perfectly reversible



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

Lifting Scheme:

Reversible After Quantization

From page 500

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

Since

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

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

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

from page 487, one set that

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

then

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

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

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

where

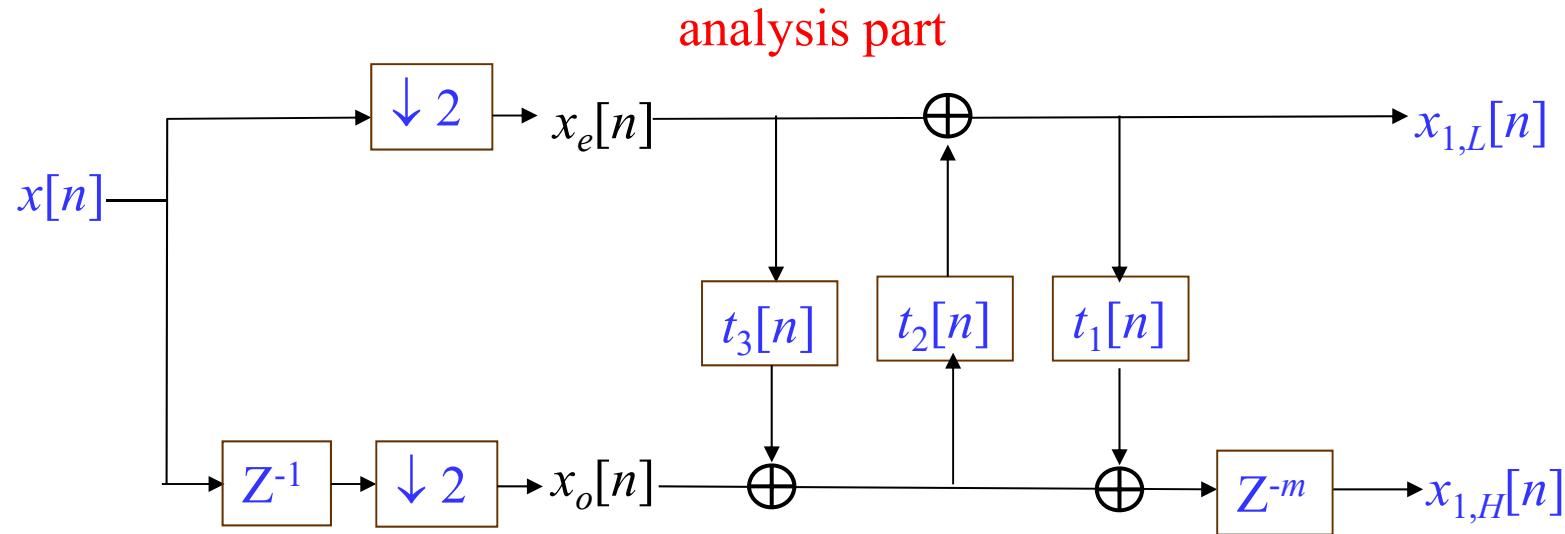
$$L_1(z) = \frac{z^m H_o(z) - 1}{G_o(z)} \quad L_2(z) = G_o(z) \quad L_3(z) = \frac{G_e(z) - 1}{G_o(z)}$$

Then the DWT can be approximated by

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

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

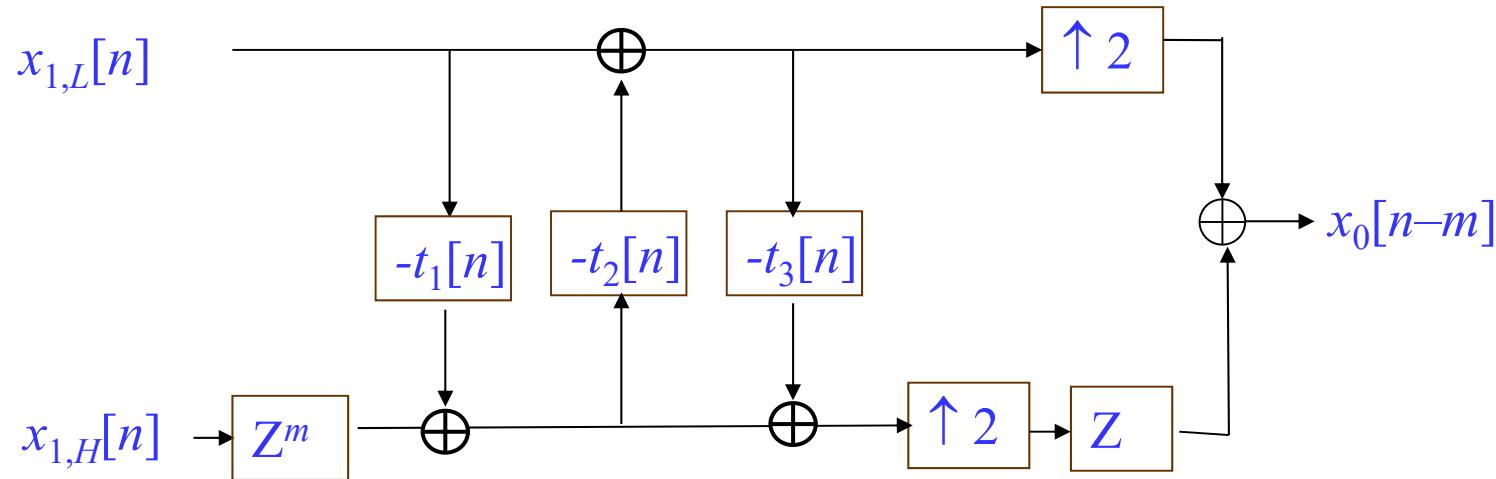
Lifting Scheme



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.

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

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

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.

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

$$\text{Max}(|y[m, n] - x[m, n]|)$$

(2) square error

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

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

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

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

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

(5) root mean square error (RMSE)

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

(6) normalized mean square error (NMSE)

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

(7) normalized root mean square error (NRMSE) ,

信號處理和影像處理常用

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

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

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

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

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

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

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

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

$\text{color} = R, G, \text{ or } B$

(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

$$DSSIM(x, y) = 1 - SSIM(x, y)$$

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

μ_x, μ_y : means of x and y

σ_x^2, σ_y^2 : variances of x and y

$\sigma_x\sigma_y$: covariance of x and y

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.