ECE462 – Lecture 14

Wavelet Transform

- Brief review of orthogonal signal expansion

Often, a function f(t) can be better analyzed if it is expressed as

$$f(t) = \sum_k a_k \psi(t)$$

a_k: set of coefficients

$\psi(t)$: set of “basis” functions

Orthogonal set of basis:

$$\int \psi_k(t)\psi_l(t)dt = 0, k \neq l$$

Wavelet Transform

- Orthogonal set if in additional to orthogonality:

$$\int \psi_k(t)\psi_l(t)dt = 1, k = l$$

Then, a_k= $\int f(t)\, \psi_k(t)dt$

Note: Similar relations hold for discrete signals where summation may replace integration

Wavelet Transform

Examples:

IDFT: $f(n) = \frac{1}{N}\sum_{k=0}^{N-1} F(k)e^{\frac{j2\pi kn}{N}}, n = 0, \ldots, N-1$

IDCT: $x(n) = c(k)\sum_{k=0}^{N-1} X(k)\cos\left(\frac{(2n+1)k\pi}{2N}\right), n = 0, \ldots, N-1$

It is easy to show that

$\sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)k\pi}{2N}\right)\cos\left(\frac{(2n+1)l\pi}{2N}\right) = 0$, k≠l

Similarly, for the exponential basis of IDFT above

Wavelet Transform

- C(k) and 1/N makes the basis orthonormal.
- Even though the DFT and DCT provide an orthonormal expansion that is useful in applications nevertheless do not provide sufficient time frequency localization.
- In other words: A local change in the time or frequency affects all frequencies or times respectively.
- A better time-frequency localization is provided by the wavelet transform.

$$f(t) = \sum_{k} \sum_{j} a_{j,k} \psi_{j,k}(t)$$

Where, the basis, $\psi_{j,k}(t) = 2^{\frac{j}{2}} \psi(2^{j} t - k)$ is called a wavelet and it is generated by "scaling" and "translation" of the mother wavelet $\psi(t)$

Wavelet Transform

- Wavelet transform provides a multiresolution expansion that allows to new the signal in various times and scales (like zooming in a map).
- Actually, the expansion takes the form:

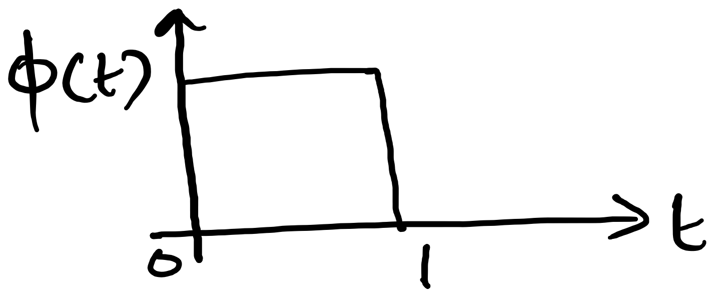
$$f(t) = \sum_{k=-\infty}^{\infty} C_k \phi(t-k) + \sum_{k=-\infty}^{\infty} \sum_{j=0}^{\infty} d_{j,k}.\psi(2^j t - k)$$

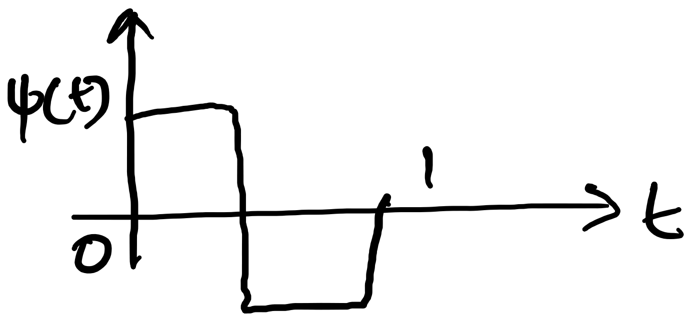
Scaling function $\phi(t)$

Wavelet $\psi(t)$

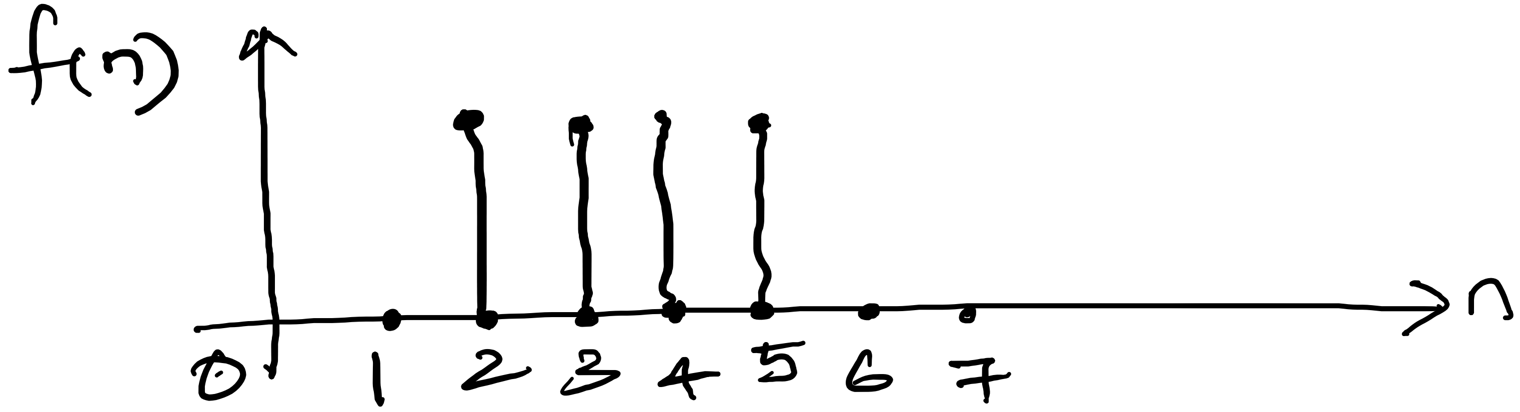
Wavelet Transform

- Ex: Haar scaling function and wavelet





- Example: Consider a discrete pulse



Wavelet Transform

- 8-DFT: $\mathrm{F(k)} = \sum_{n=0}^{7} f(n) e^{-\frac{j2\pi kn}{8}}, k = 0, \ldots, 7$

$\mathrm{F(k)} = \sum_{n=2}^{5} e^{-\frac{j2\pi kn}{8}} = e^{-\frac{j2\pi k.2}{8}} \sum_{n=0}^{3} e^{-\frac{j2\pi kn}{8}} = e^{-\frac{j\pi k}{2}} \quad \frac{1- e^{-j\pi k}}{1-e^{\wedge}(-\frac{j\pi k}{4})} =$

$e^{-\frac{j\pi k}{8}}. \sin(\frac{\pi k}{2}) / \sin\left(\frac{\pi k}{8}\right)$, k=0,1,…..,7

Then,

k = 0, 1, 2, 3, 4 , 5, 6 , 7

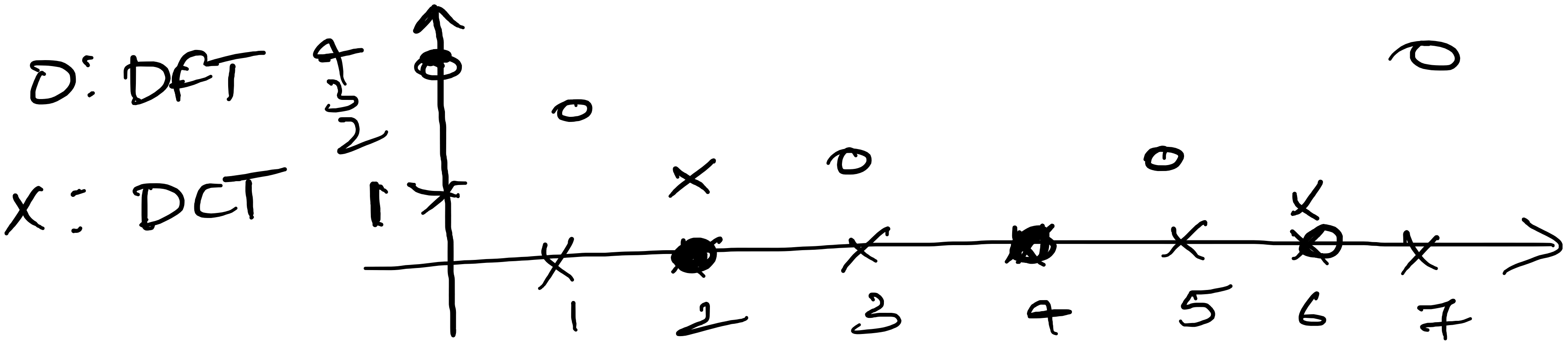
|F(k)|= 4, 2.61,0, 1.08, 0, 1.08, 0, 2.61

Wavelet Transform

- Similarly the DCT will give

k = 0 1 2 3 4 5 6 7

|DCT(k)| = 1.41, 0, 1.3, 0, 0, 0, 0.54, 0

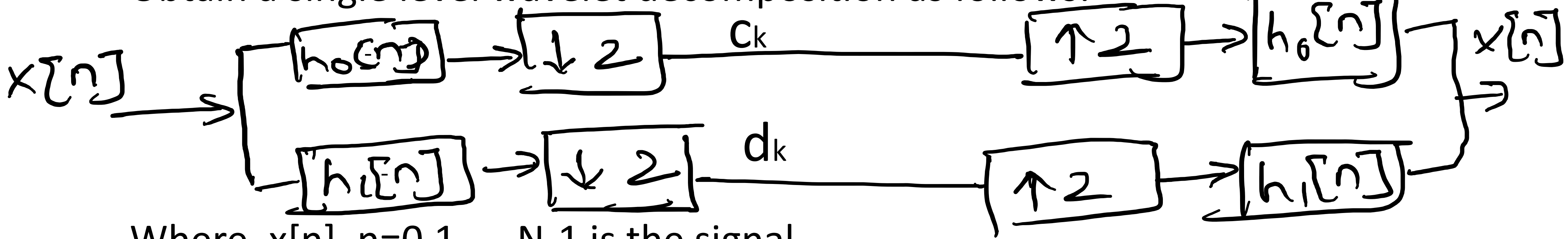


- So once again we witness the better energy compaction for the DCT.
- However, we do not get information regarding the localization of the pulse in time.

Implementation of Wavelet transform (dyadic)

- Given two filters:
  - low pass - $h_0(n)$
  - high pass - $h_1(n) = -1^n h_0(1-n)$

  Obtain a single level wavelet decomposition as follows:



Where, x[n], n=0,1,…, N-1 is the signal

C_k, k=0,1,…,N/2-1 is the “approximation”

d_k, k=0,1,…,N/2-1 is the “detail” signal

Thus, the 1-level wavelet decomposition $\{C_0, C_1, \ldots, C_{N/2-1}, d_0, d_1, \ldots d_{N/2-1}\}$

Notice that in decomposition section the time reversed filters are used while in the reconstruction section filters are not time reversed.

Implementation of Wavelet transform (dyadic)

- For example using the Haar Wavelet functions

$$C_k = \frac{1}{2}\left(X(2k+1) + X(2k)\right), d_k = \frac{1}{2}(x(2k+1) - x(2k))$$

(X above should be x)

So in our previous example with the discrete pulse

k = 0,1,2,3

C_k=0,1,1,0

d_k=0,0,0,0

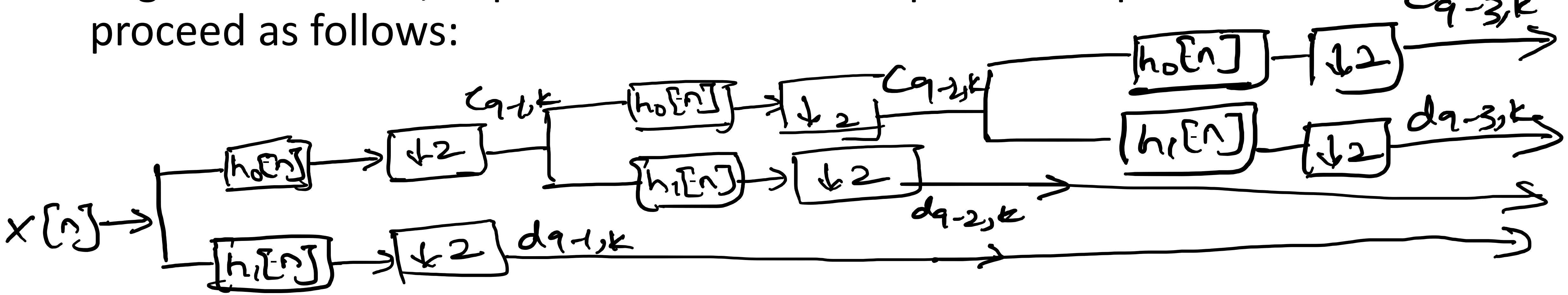
So the wavelet transform is

$\{C_k, d_k\}$ = {0,1,1,0,0,0,0,0}

Implementation of Wavelet transform (dyadic)

Notice that k above corresponds to the downsampled rate so the values of 1 in C_k correspond to the edges of the pulse (better localization)

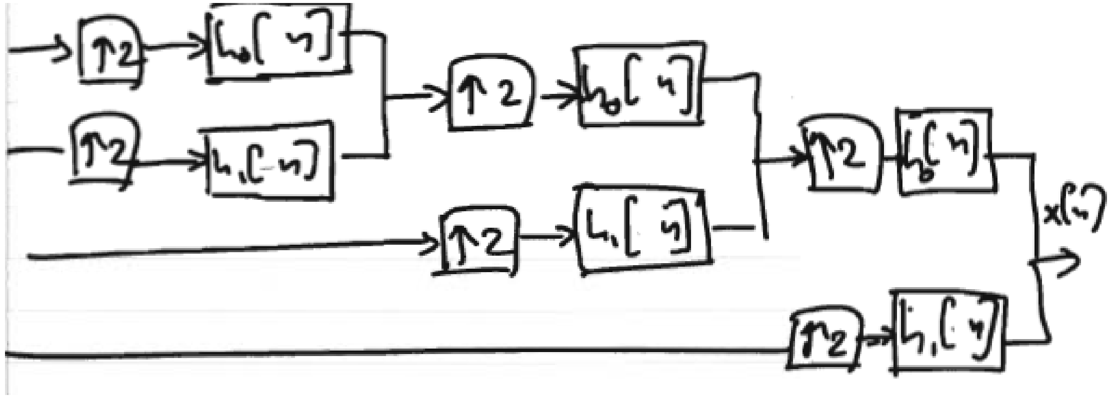
In general if $N=2^q$, a q-level wavelet decomposition is possible. We proceed as follows:



Notice that in the decomposition stages all the filters are reversed in time, that is h{-n] is used to filter the inputs

Implementation of Wavelet transform (dyadic)

And so on. Then, reconstruction of x[n] is based on reverse operations, that is:



Thus, a q-level wavelet decomposition (where $N=2^q$) takes the following form.

Notice that in reconstruction section above the non-reversed in time filters $h[n]$ are used.

Implementation of Wavelet transform (dyadic)

$\{C_{0,k}, \ d_{0,k}, \ d_{1,k}, \ d_{2,k}, \ldots., \ d_{q-1,k}\}$

Rate: $\frac{1}{N}, \ \frac{1}{N}, \ \frac{2}{N}, \frac{4}{N}, \ldots\ldots, \frac{\frac{N}{2}}{N} = 1/2$

Samples : 1 1 2 4 ….. N/2= N samples

So we observe that for high frequencies we have low frequency resolution but high time resolution while for low frequencies high frequency resolution but low time resolution.

- Back to the discrete pulse example where N=8

Level 3 decomposition

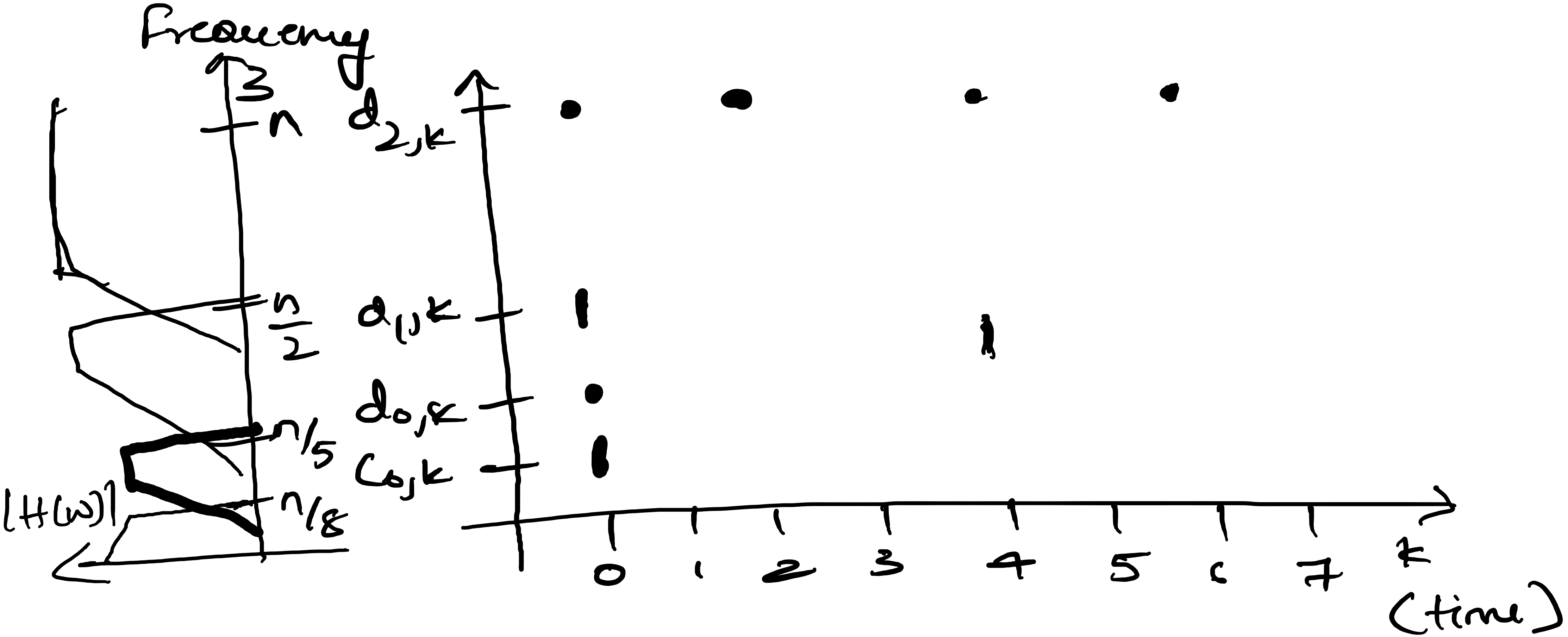
$\{C_{2,k}, d_{2,k}\} = \{0,1,1,0,0,0,0,0\}$

Level 2 decomposition

$\{C_{1,k}, d_{1,k}, d_{2,k}\} = \{1/2,1/2,1/2,-1/2,0,0,0,0\}$

Level 1 decomposition

$\{C_{0,k}, d_{0,k}, d_{1,k}, d_{2,k}\} = \{1/2,0,1/2,-1/2,0,0,0,0\}$



- Once again: high frequency -> high time resolution
- Low frequency -> low time resolution

Let us repeat the example with a signal of length 16 samples

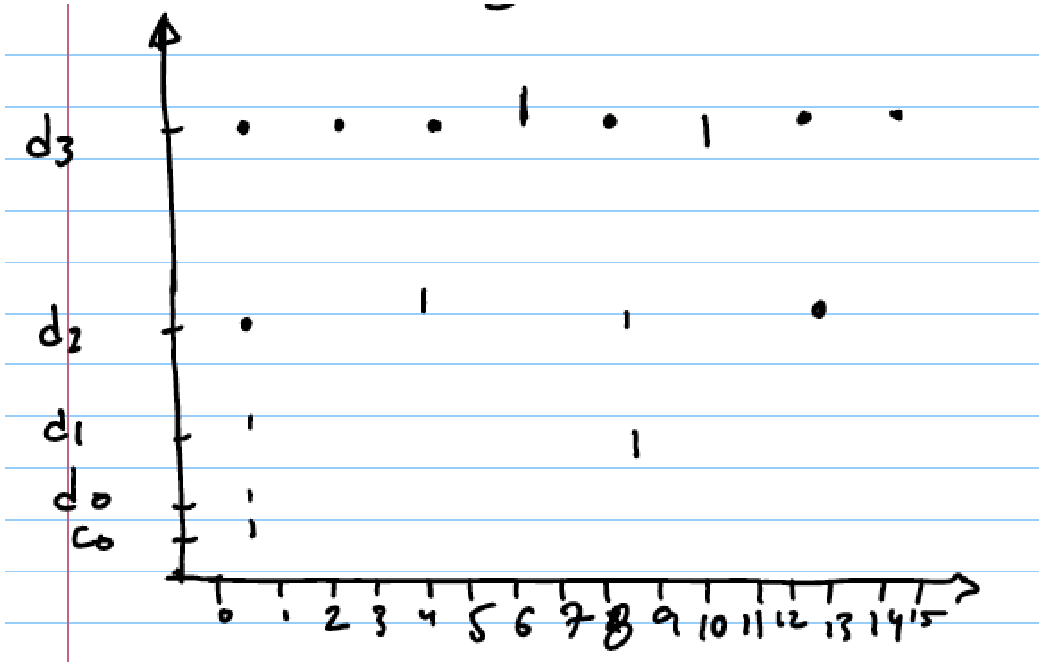
$x[n] : \{0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\}$

$\{c_{3,k}, d_{3,k}\} : \{0\ 0\ 0\ \tfrac{1}{2}\ 1\ \tfrac{1}{2}\ 0\ 0\ \ 0\ 0\ 0\ \tfrac{1}{2}\ 0\ -\tfrac{1}{2}\ 0\ 0\}$

$\{c_2, d_2, d_3\} : \{0\ \tfrac{1}{4}\ \tfrac{3}{4}\ 0\ 0\ \tfrac{1}{4}\ -\tfrac{1}{4}\ 0\ \ 0\ 0\ 0\ \tfrac{1}{2}\ 0\ -\tfrac{1}{2}\ 0\ 0\}$

$\{c_1, d_1, d_2, d_3\} : \{\tfrac{1}{8}\ \tfrac{3}{8}\ \tfrac{1}{8}\ -\tfrac{3}{8}\ 0\ \tfrac{1}{4}\ -\tfrac{1}{4}\ 0\ \ 0\ 0\ 0\ \tfrac{1}{2}\ 0\ -\tfrac{1}{2}\ 0\ 0\}$

$\{c_0, d_0, d_1, d_2, d_3\} : \{\underbrace{\tfrac{1}{4}}_{c_0}\ \underbrace{\tfrac{1}{8}}_{d_0}\ \underbrace{\tfrac{1}{8}\ -\tfrac{3}{8}}_{d_1}\ \underbrace{0\ \tfrac{1}{4}\ -\tfrac{1}{4}\ 0}_{d_2}\ \underbrace{0\ 0\ 0\ \tfrac{1}{2}\ 0\ -\tfrac{1}{2}\ 0\ 0}_{d_3}\}$



2-D Wavelet Decomposition

Given an image of NxN pixels $N=2^q$

For orthogonal wavelet analysis specify four filters

$h_{0,0}[x, y]$ 2-D low pass

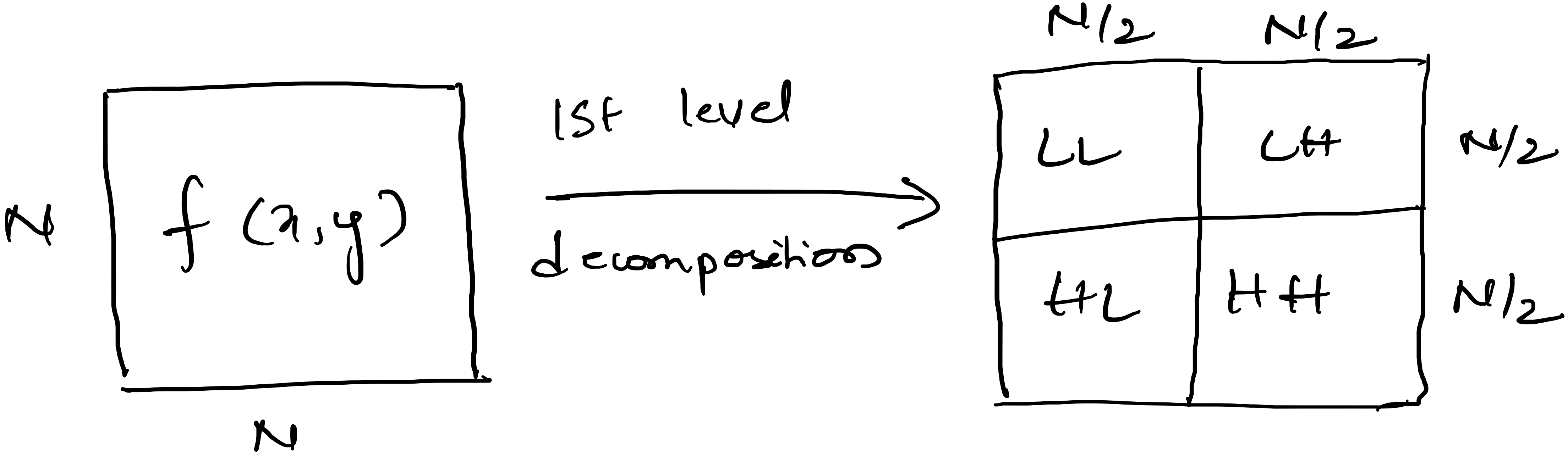
$h_{0,1}[x, y]$ LP vertically, HP horizontally

$h_{1,0}[x, y]$ HP vertically, LP horizontally

$h_{1,1}[x, y]$ HP vertically, HP horizontally

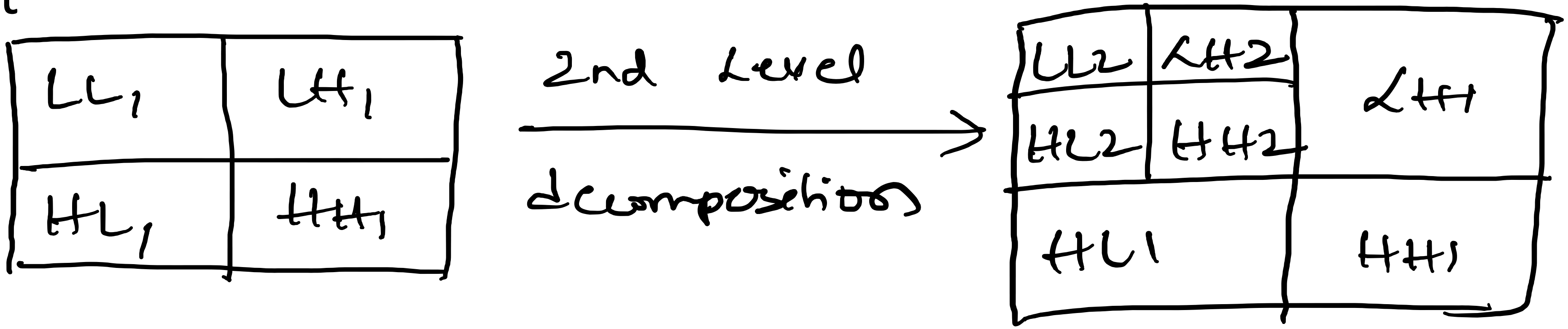
2-D Wavelet Decomposition

- Convolve the image with each filter (space reversed)
- Downsample by 2 vertically and horizontally
- Form the first level decomposition as follows:



2-D Wavelet Decomposition

- For two level decomposition repeat the process with the LL component



and so on:

Let

$f(x,y)$

1	0	0	0
1	0	0	0
1	0	0	0
1	1	1	1

2D Haar filters

$h_{0,0}(x,y) = \begin{matrix} 1 & 1 \\ 1 & 1 \end{matrix}$

$h_{0,1}(x,y) = \begin{matrix} 1 & -1 \\ 1 & -1 \end{matrix}$

$h_{1,0}(x,y) : \begin{matrix} 1 & 1 \\ -1 & -1 \end{matrix}$

$h_{1,1}(x,y) : \begin{matrix} 1 & -1 \\ -1 & 1 \end{matrix}$

Separable Wavelet transforms

Scaling functions $\phi(x, y) = \phi(x)\phi(y)$

Wavelet functions $\psi(x, y) = \psi(x)\psi(y)$

So similarly for the filters

$$h_{00}(x, y) = h_0(x)h_0(y)$$

$$h_{0l}(x, y) = h_0(x)h_l(y)$$

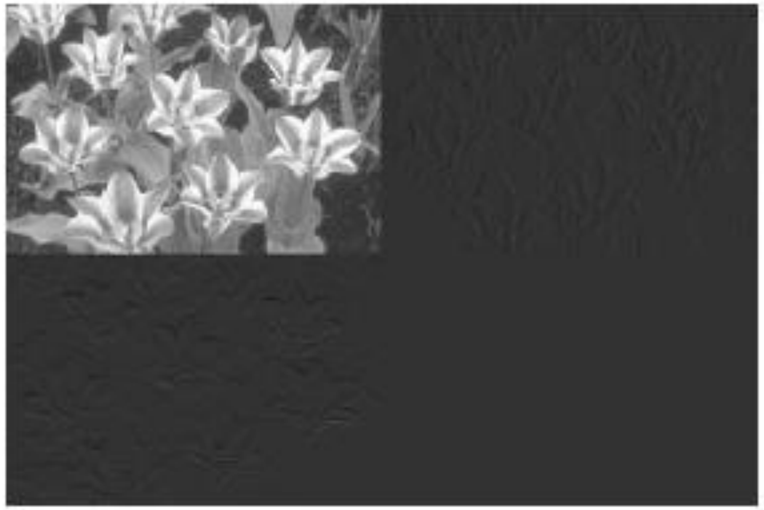
Etc.

Separable Wavelet transforms Implementation

- Convolve each row of the image with $h_0(n)$ and $h_l(n)$ discard the odd number columns and concatenate them.
- Convolve each column of the result with $h_0(n)$ and $h_l(n)$ discard the odd number rows and concatenate.

2-D DWT Image Example

Image



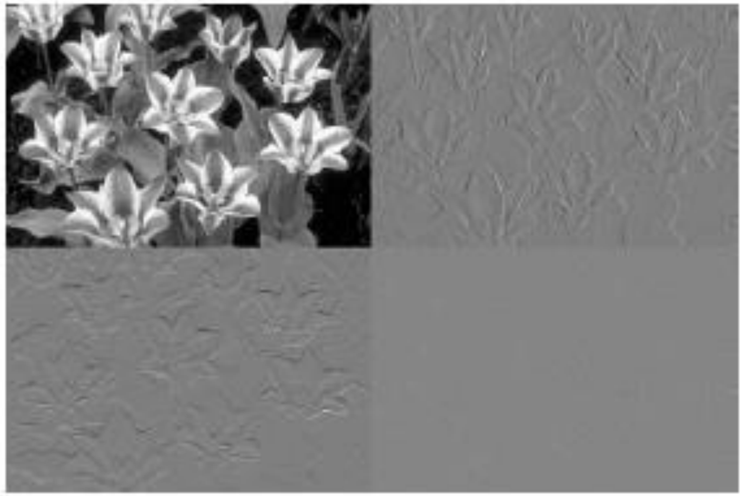
1st Level DWT

2nd Level DWT

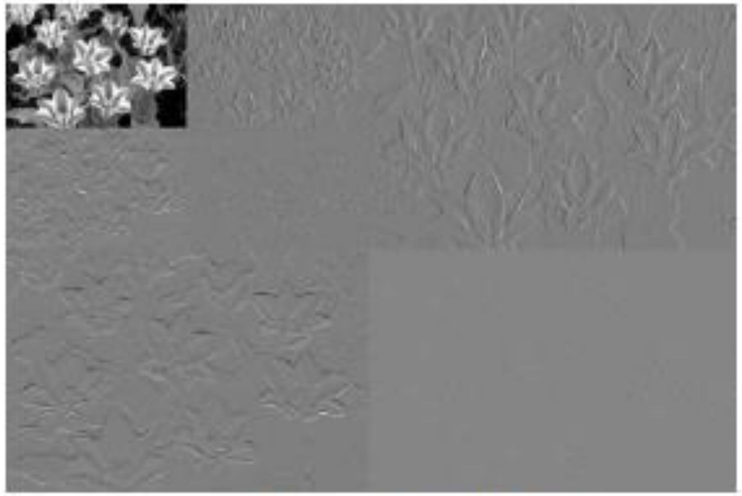
3rd Level DWT

2-D DWT Image Example

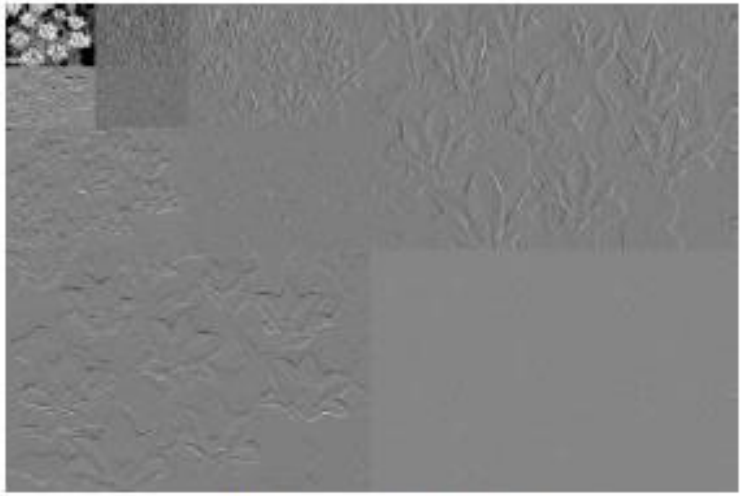
Increase contrast to see detail in high-frequency subbands



1^{st} Level DWT



2^{nd} Level DWT



3^{rd} Level DWT