# Short Time Fourier Transform (STFT)

# Fourier Transform

- Fourier Transform reveals which frequency components are present in a function:

$$f(x) = \sum_{u=0}^{N-1} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

(inverse DFT)

where:

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{\frac{-j2\pi ux}{N}}, u = 0, 1, \ldots, N-1$$

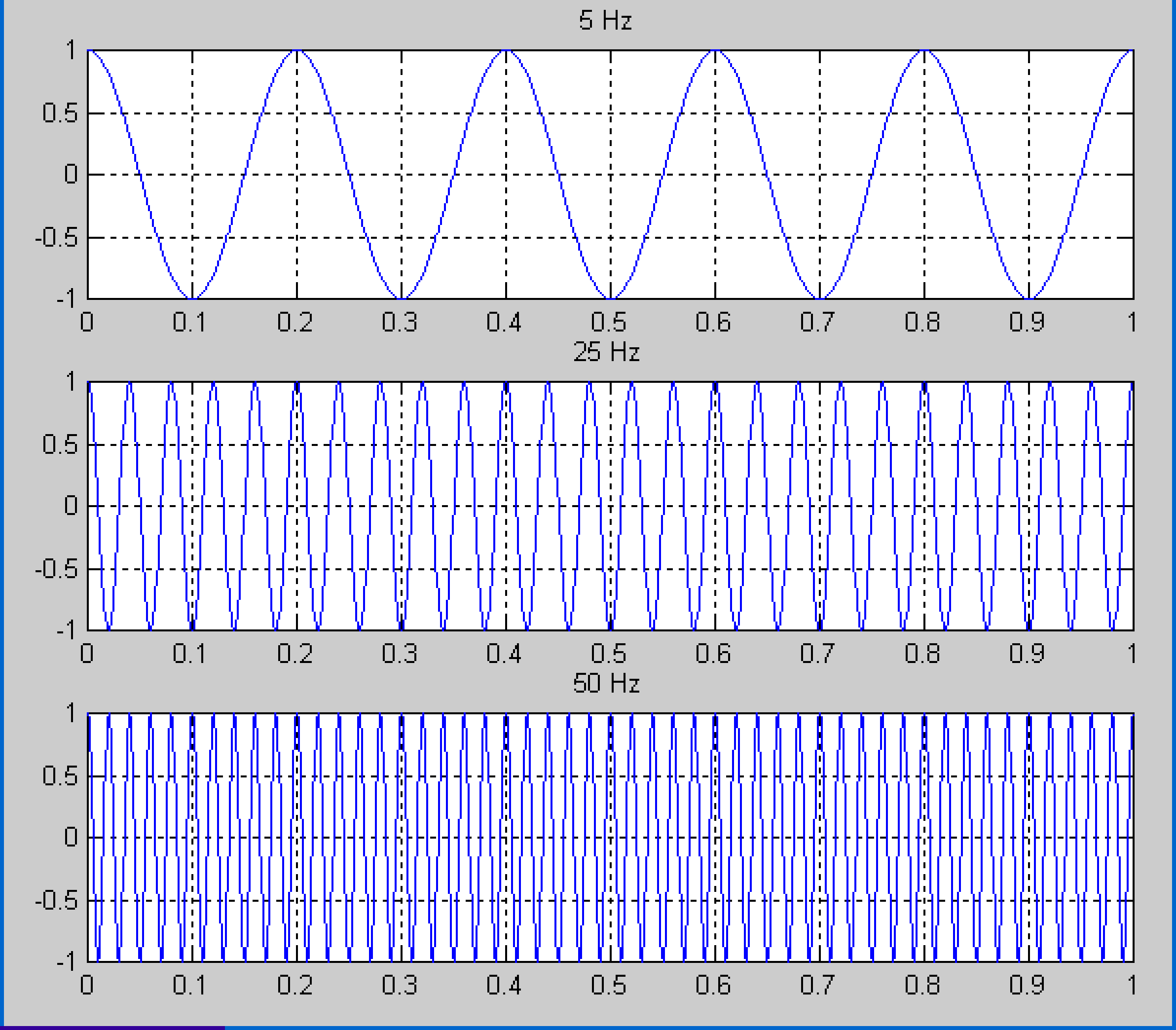
(forward DFT)

# Examples

$$f_1(t) = \cos(2\pi \cdot 5 \cdot t)$$

$$f_2(t) = \cos(2\pi \cdot 25 \cdot t)$$

$$f_3(t) = \cos(2\pi \cdot 50 \cdot t)$$

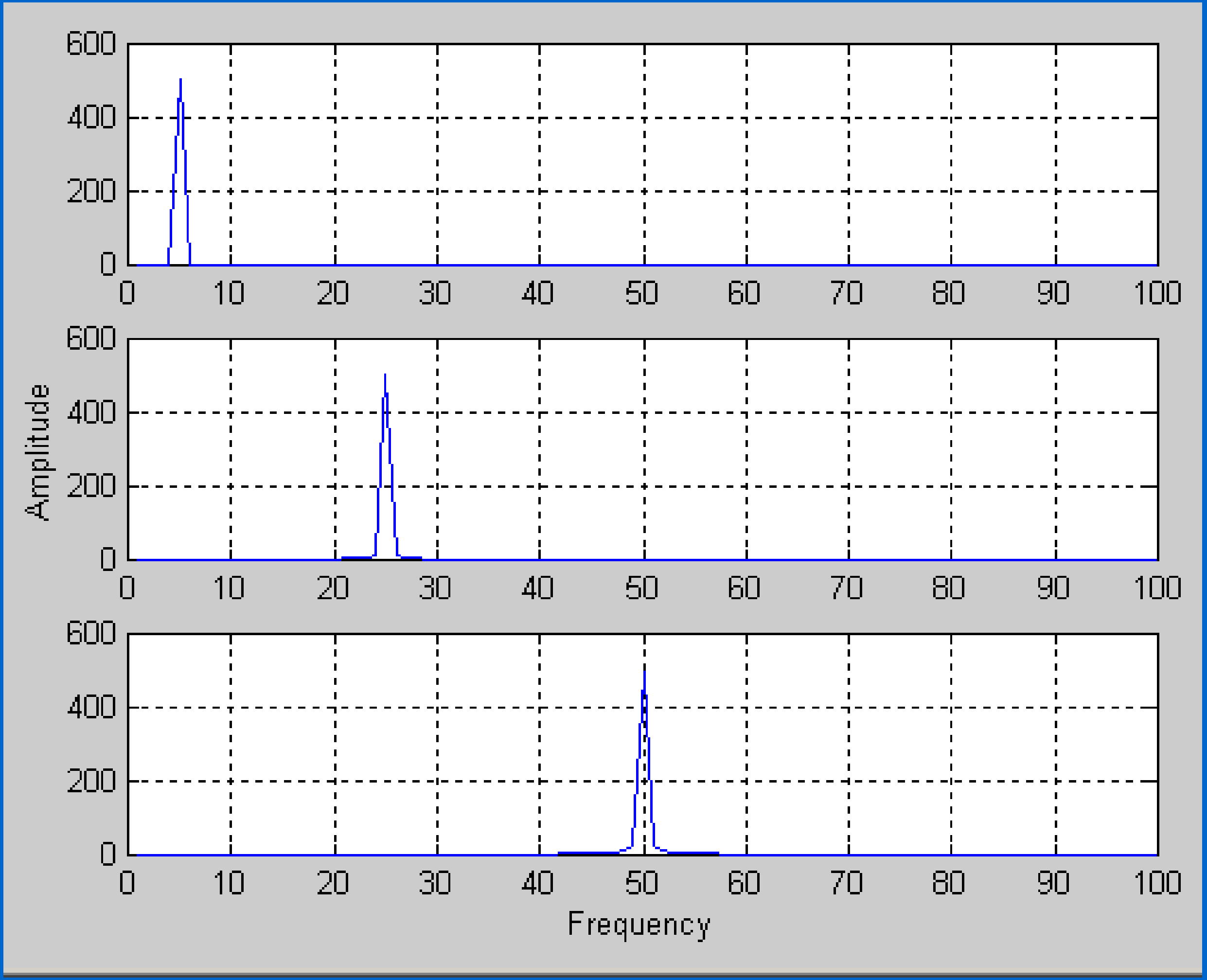


# Examples (cont'd)

$F_1(u)$

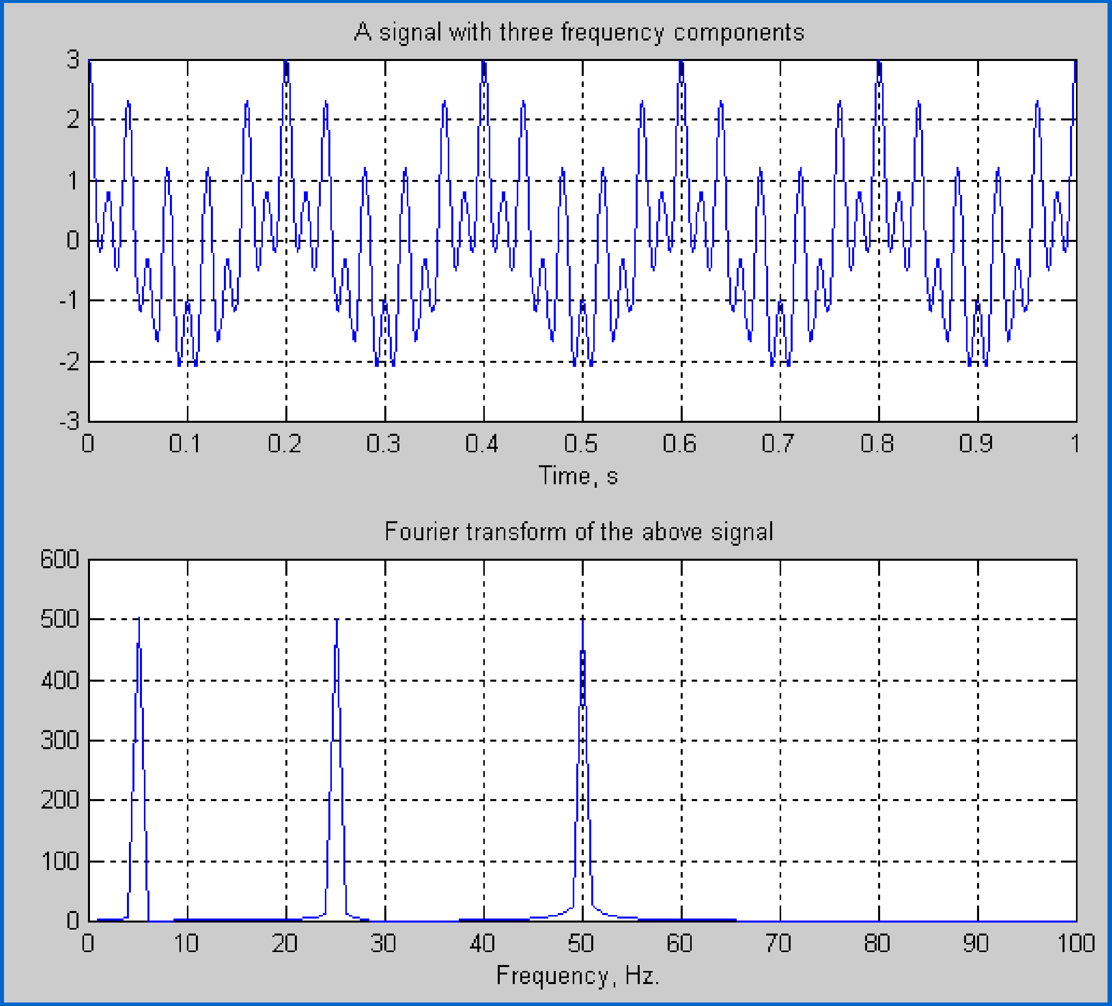
$F_2(u)$

$F_3(u)$



# Fourier Analysis – Examples (cont'd)

$$f_4(t) = \cos(2\pi \cdot 5 \cdot t) + \cos(2\pi \cdot 25 \cdot t) + \cos(2\pi \cdot 50 \cdot t)$$



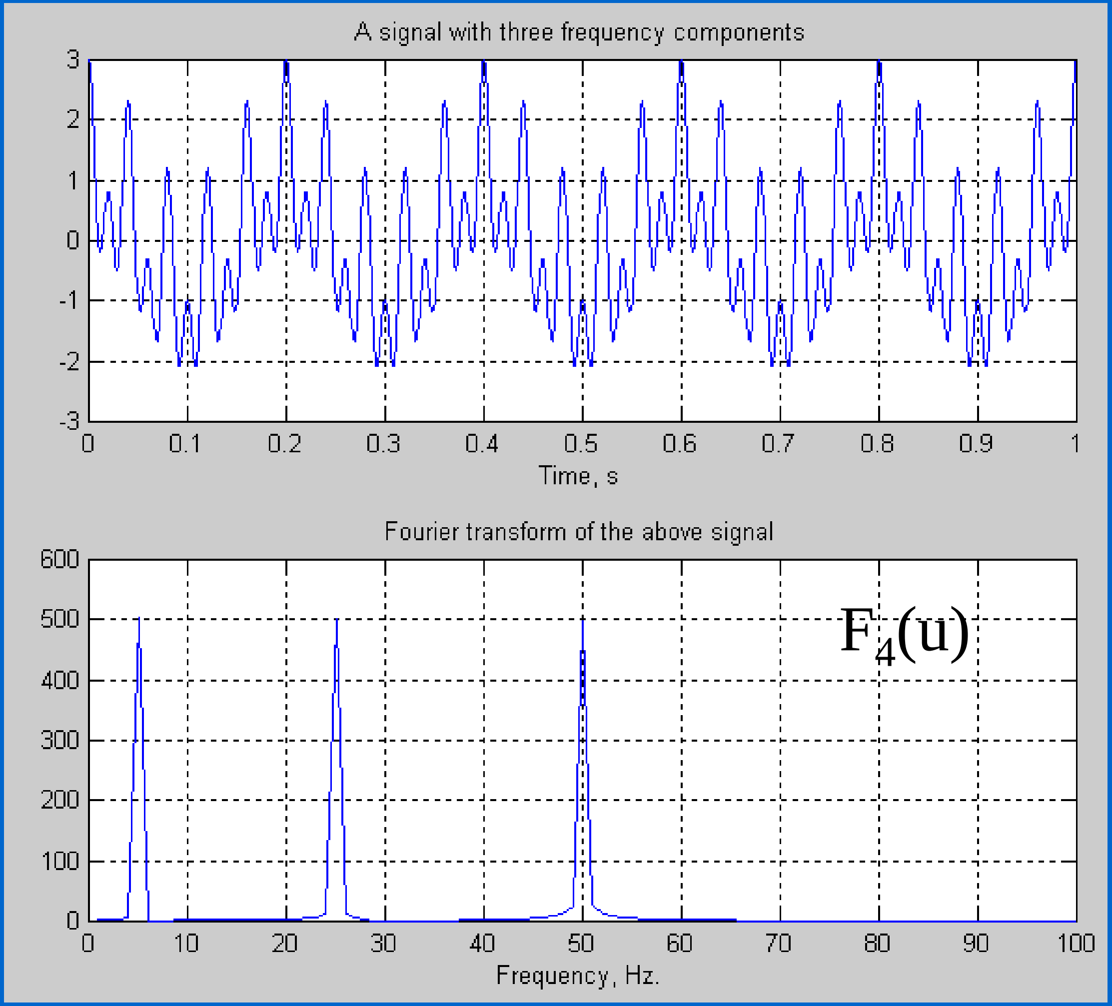

$F_4(u)$

# Limitations of Fourier Transform

1. Cannot not provide simultaneous time and frequency localization.

# Fourier Analysis – Examples (cont’d)

$$f_4(t) = \cos(2\pi \cdot 5 \cdot t) + \cos(2\pi \cdot 25 \cdot t) + \cos(2\pi \cdot 50 \cdot t)$$




Provides excellent localization in the frequency domain but poor localization in the time domain.

# Limitations of Fourier Transform (cont'd)

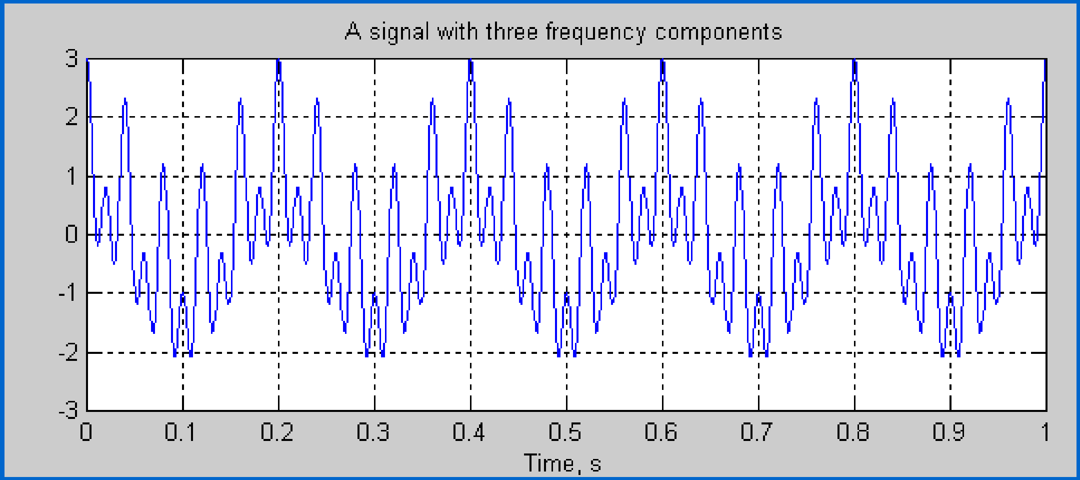
1. Cannot not provide simultaneous time and frequency localization.

2. Not very useful for analyzing time-variant, non-stationary signals.

# Stationary vs non-stationary signals

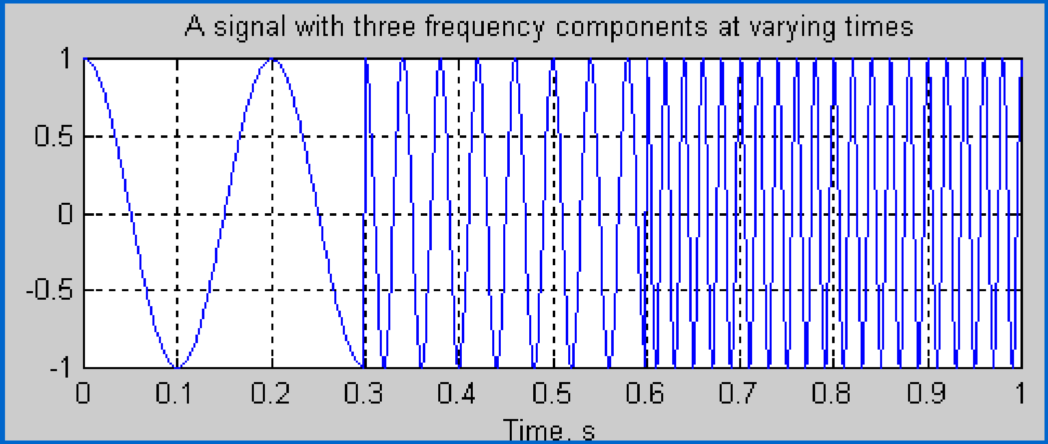
- Stationary signals: time-invariant spectra

$f_4(t)$




- Non-stationary signals: time-varying spectra

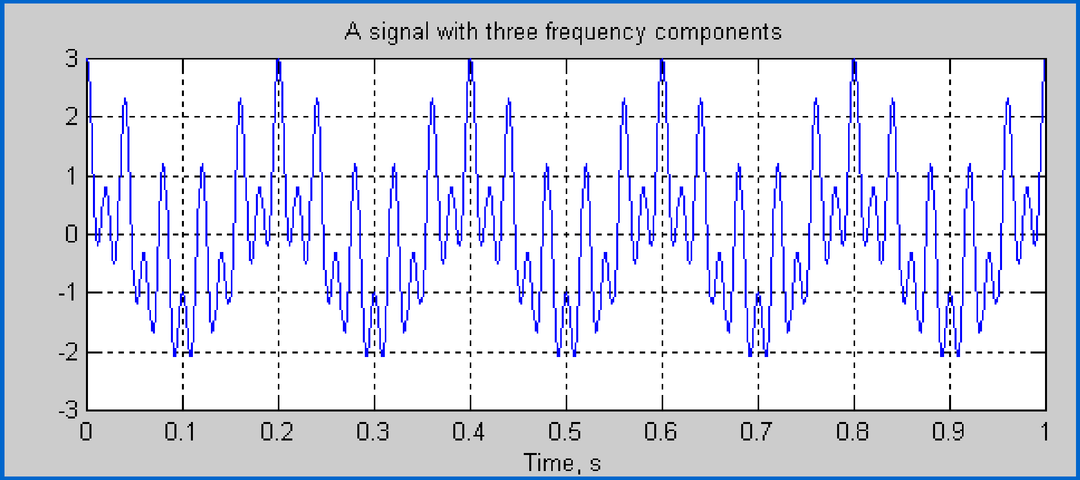
$f_5(t)$



# Stationary vs non-stationary signals (cont'd)

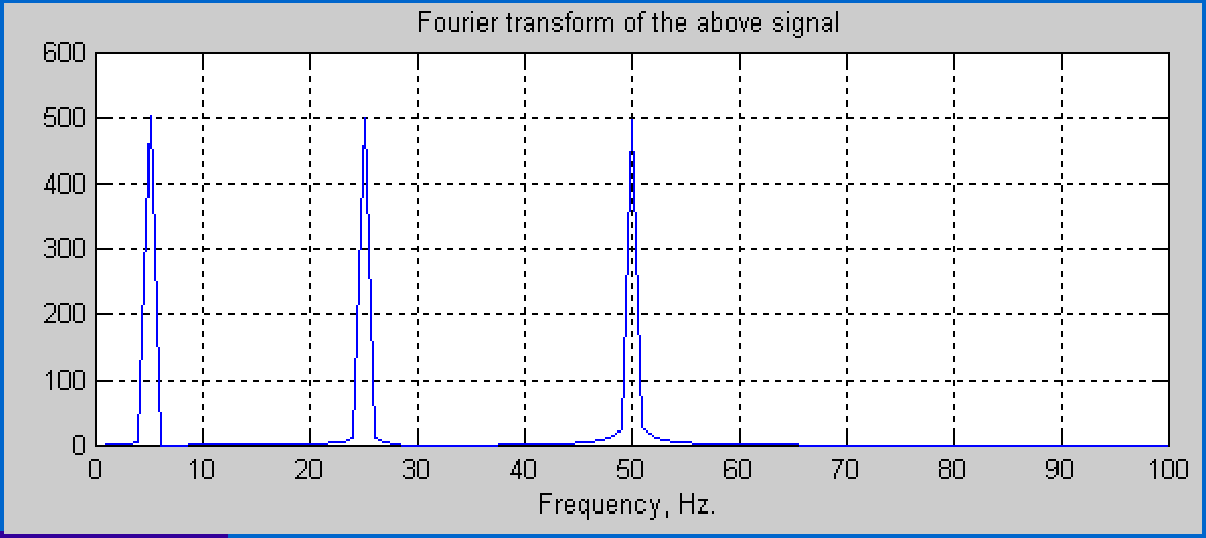
Stationary signal:

$f_4(t)$




Three frequency components, present at all times!

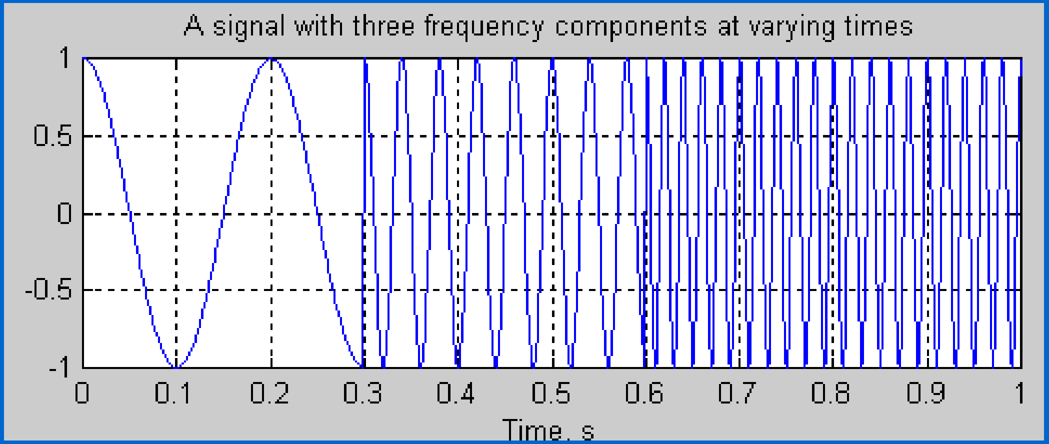
$F_4(u)$



# Stationary vs non-stationary signals (cont'd)

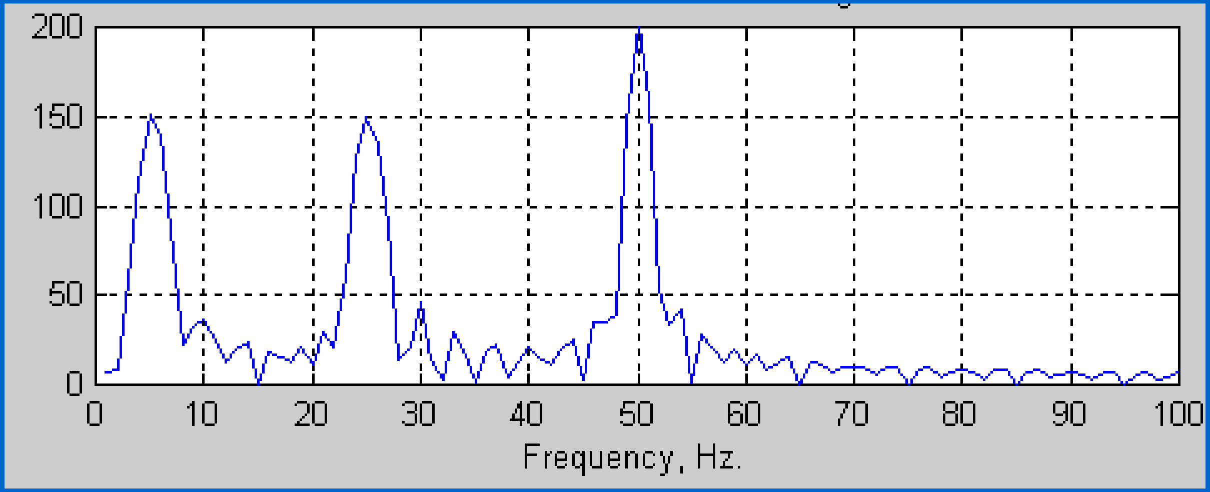
Non-stationary signal:

$f_5(t)$




Three frequency components, NOT present at all times!

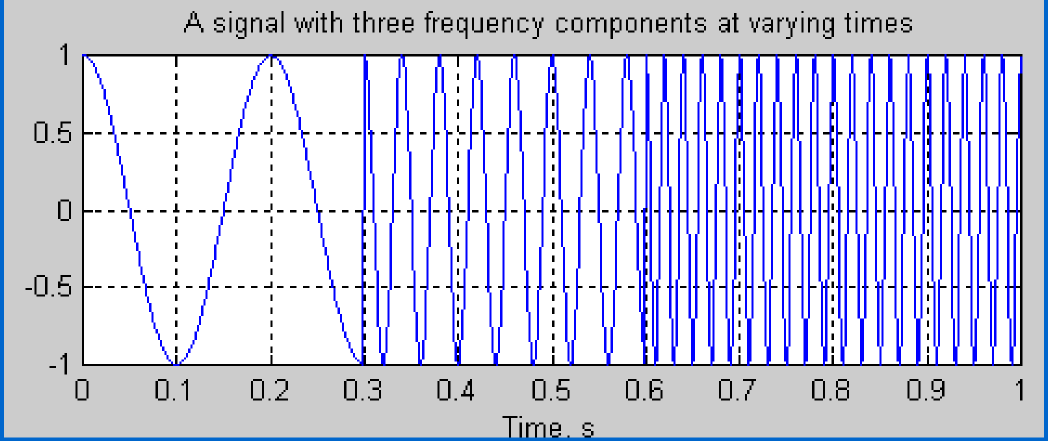
$F_5(u)$



# Stationary vs non-stationary signals (cont'd)

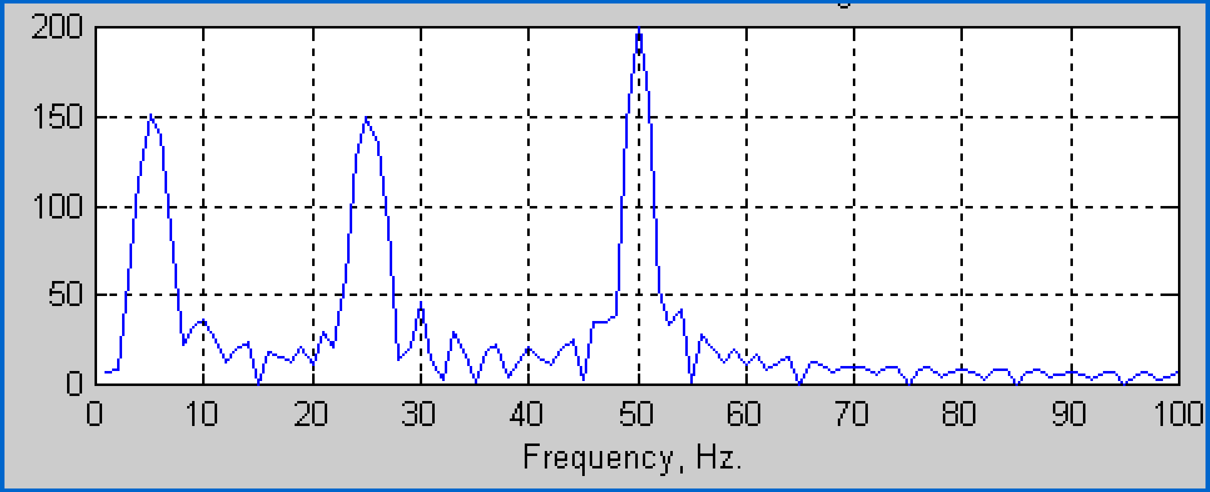
Non-stationary signal:

$f_5(t)$




Perfect knowledge of what frequencies exist, but no information about where these frequencies are located in time!

$F_5(u)$



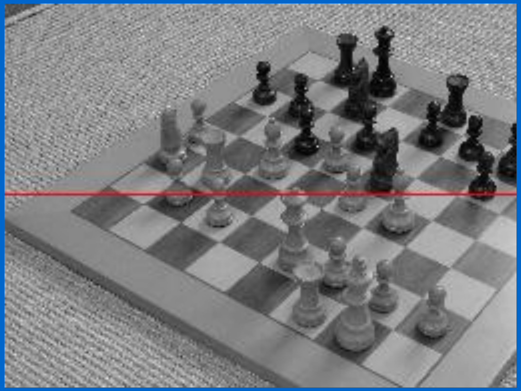
# Limitations of Fourier Transform (cont’d)

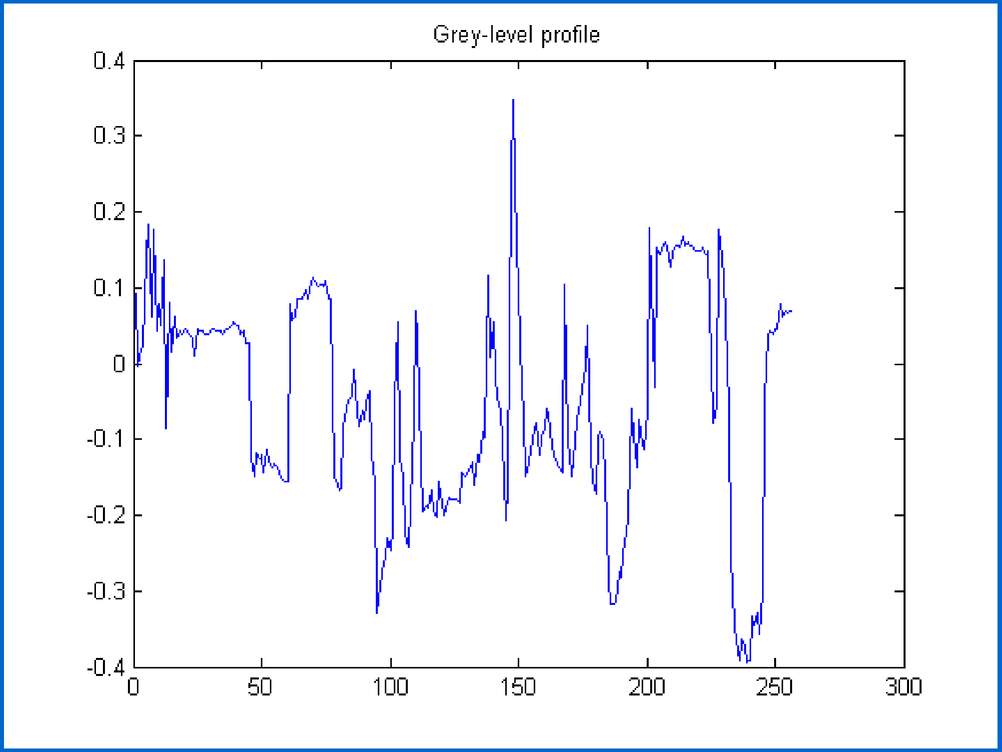
1. Cannot not provide simultaneous time and frequency localization.

2. Not very useful for analyzing time-variant, non-stationary signals.

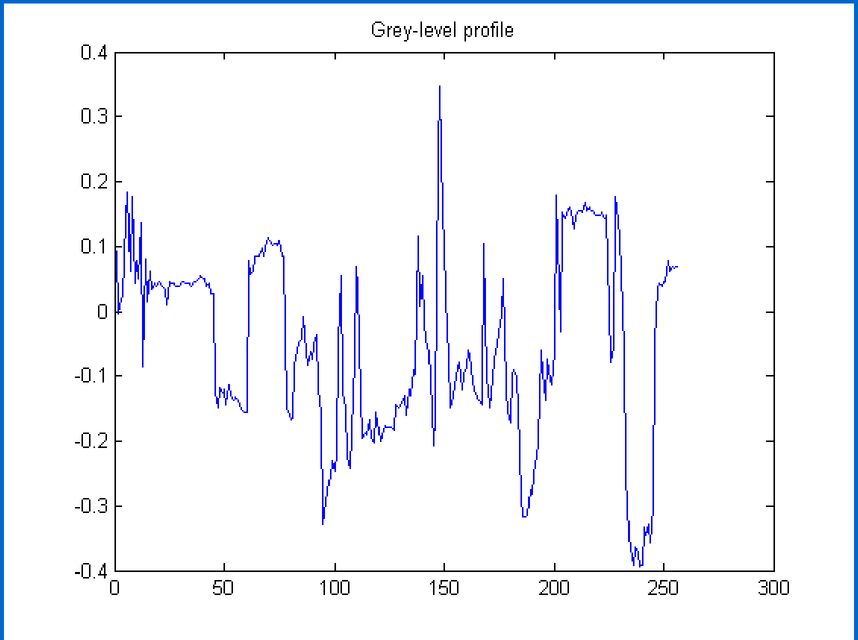
3. Not appropriate for representing discontinuities or sharp corners (i.e., requires a large number of Fourier components to represent discontinuities).

# Representing discontinuities or sharp corners

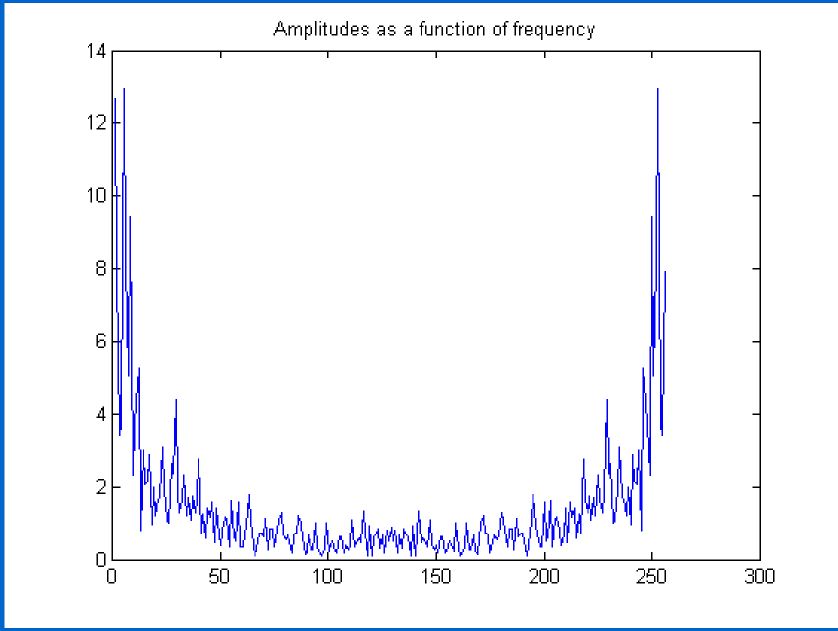




# Representing discontinuities or sharp corners (cont'd)



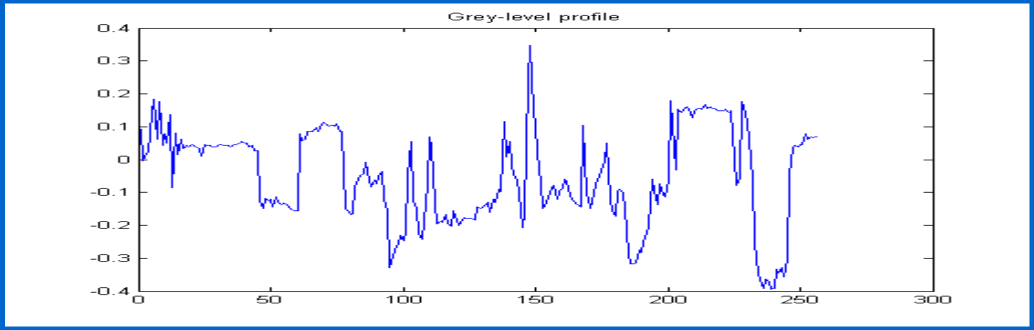

FT →

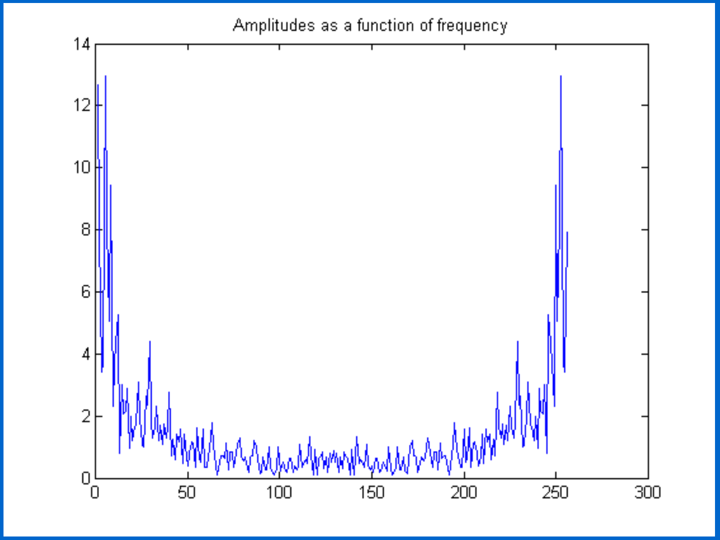



$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)e^{\frac{-j2\pi ux}{N}}, u = 0, 1, \ldots, N-1$$

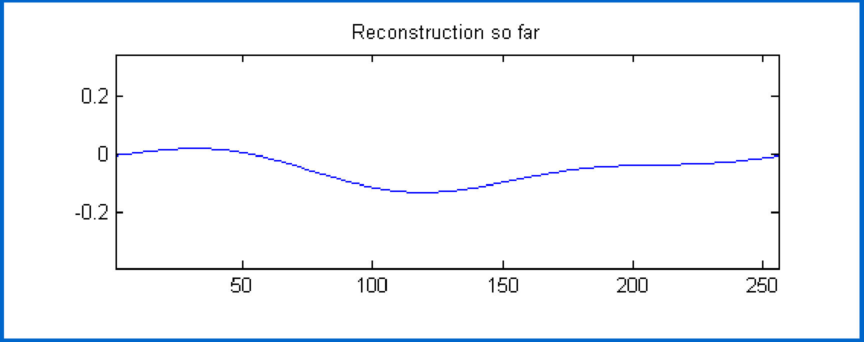
# Representing discontinuities or sharp corners (cont'd)

Original



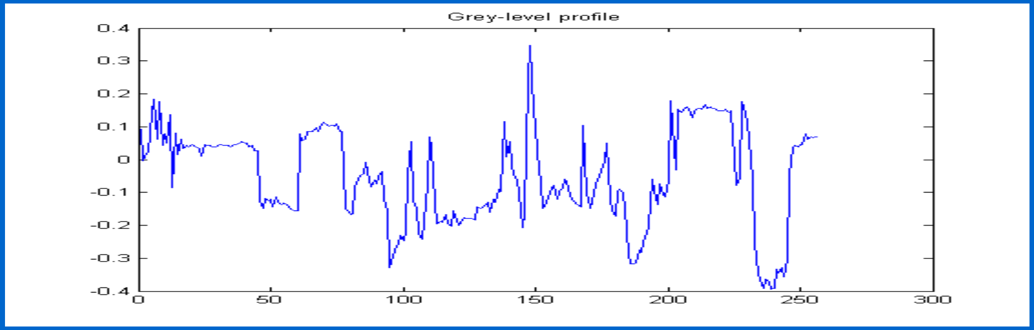

Reconstructed



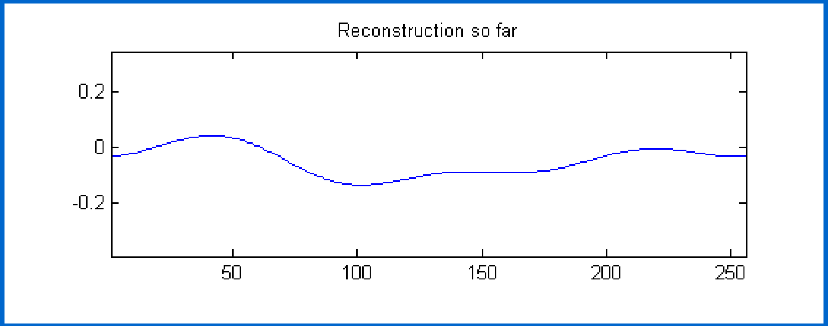

$$f(x) = \sum_{u=0}^{1} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

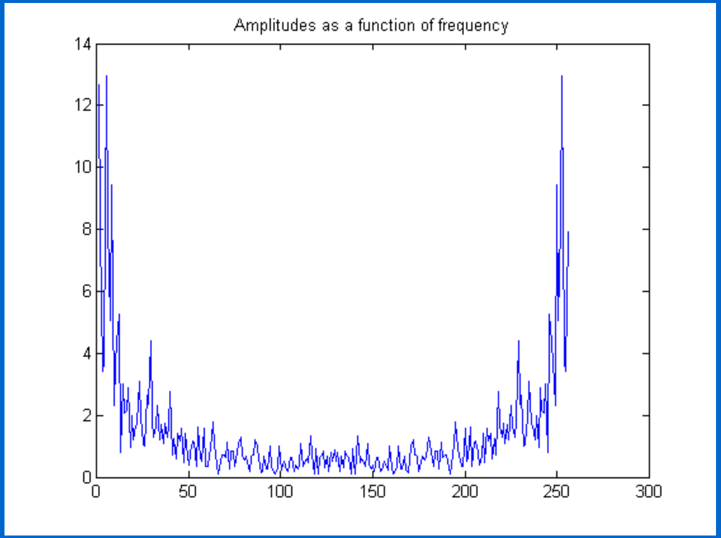
# Representing discontinuities or sharp corners (cont'd)

Original




Reconstructed

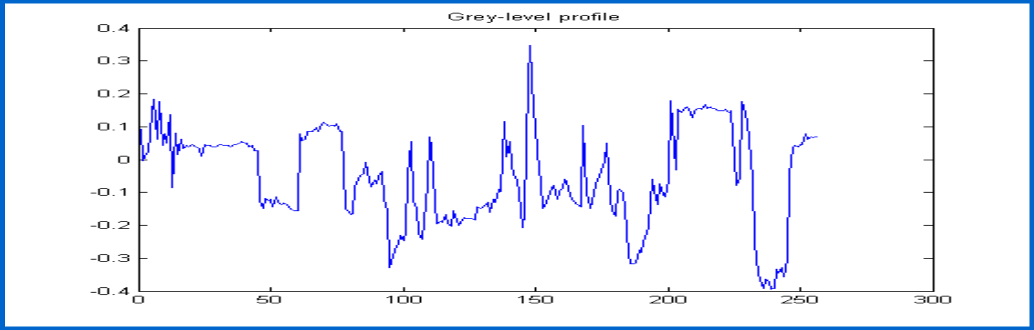


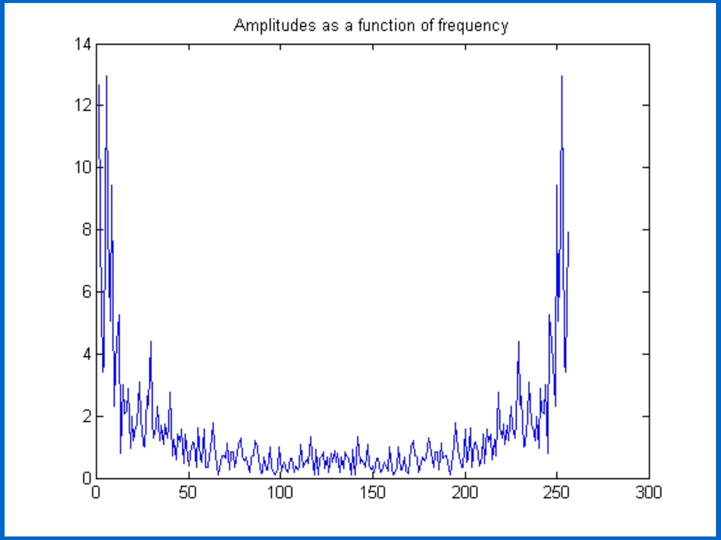




$$f(x) = \sum_{u=0}^{2} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

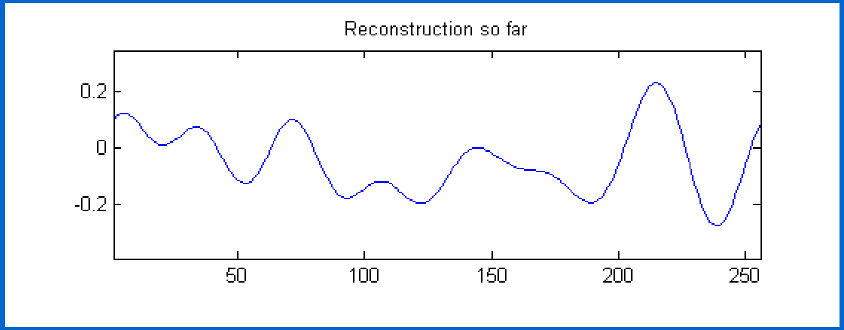
# Representing discontinuities or sharp corners (cont'd)

Original



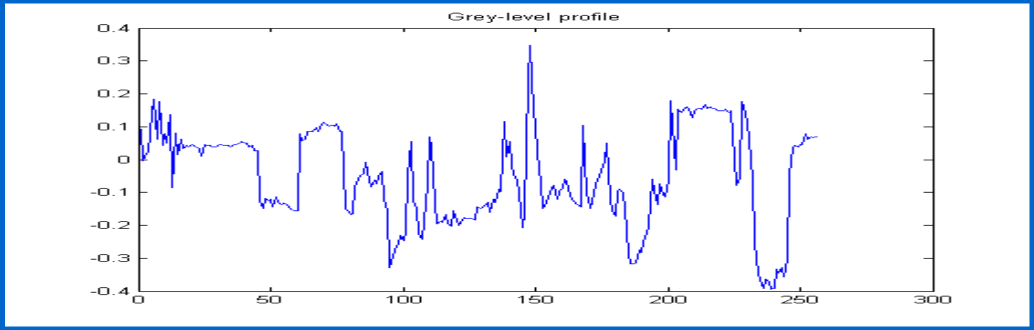

Reconstructed

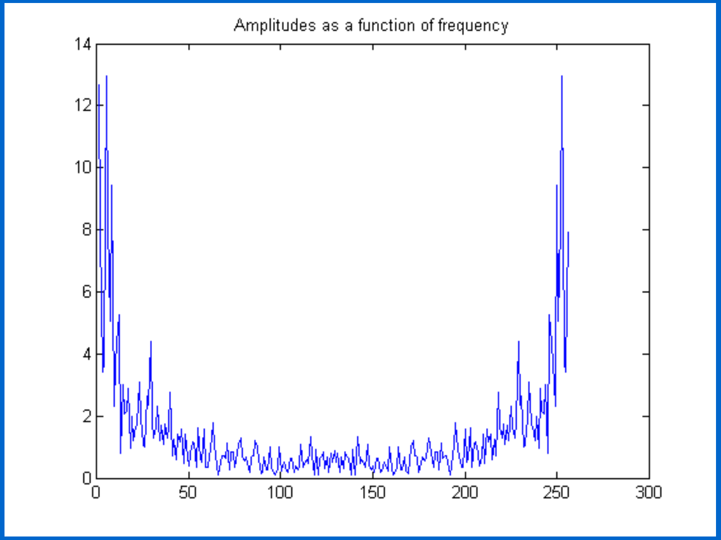



$$f(x) = \sum_{u=0}^{7} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

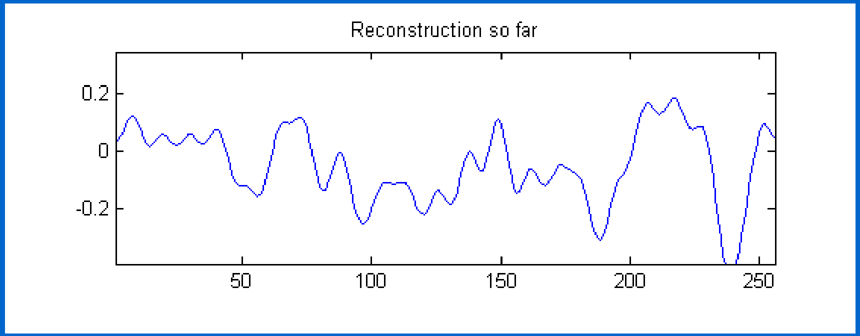
# Representing discontinuities or sharp corners (cont'd)

Original



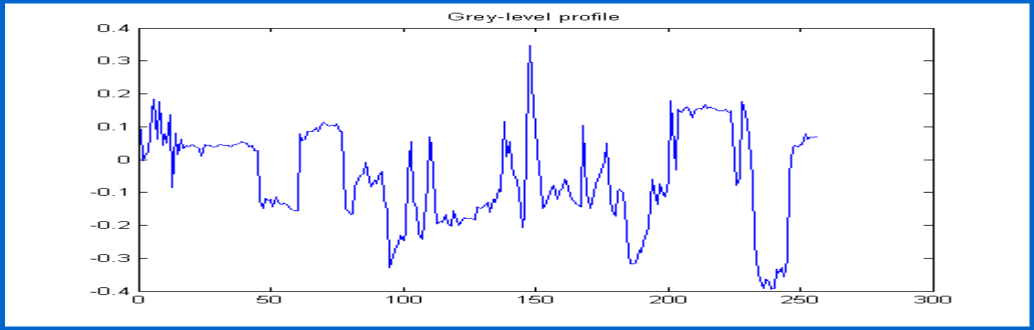

Reconstructed

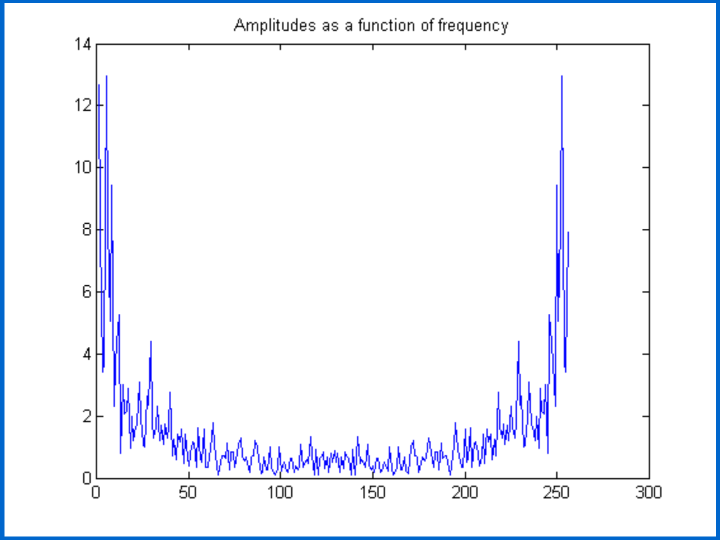



$$f(x) = \sum_{u=0}^{23} F(u) e^{\frac{j2\pi ux}{N}}, \; x = 0, 1, \ldots, N-1$$

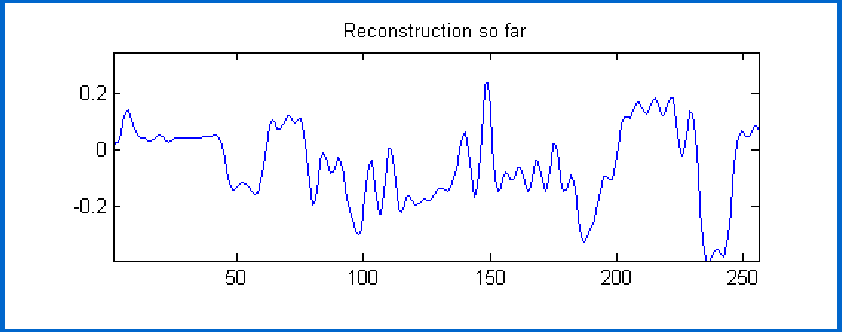
# Representing discontinuities or sharp corners (cont'd)

Original



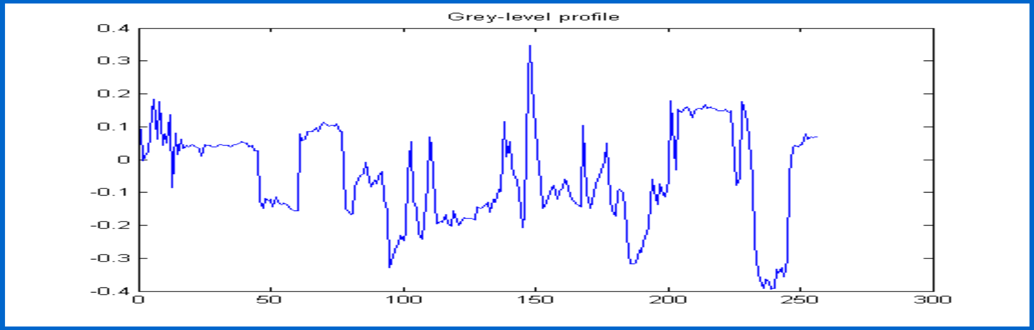

Reconstructed

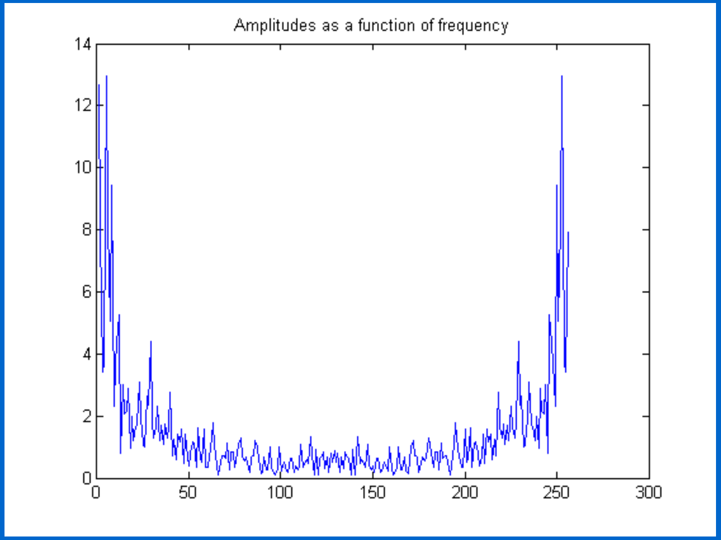



$$f(x) = \sum_{u=0}^{39} F(u) e^{\frac{j2\pi ux}{N}}, \quad x = 0, 1, \ldots, N-1$$

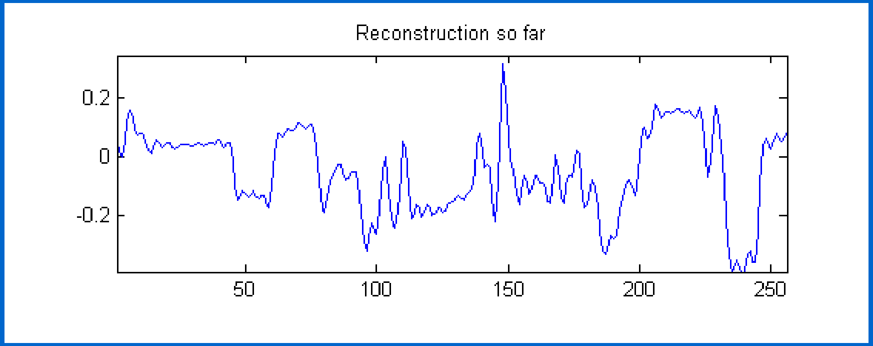
# Representing discontinuities or sharp corners (cont'd)

Original



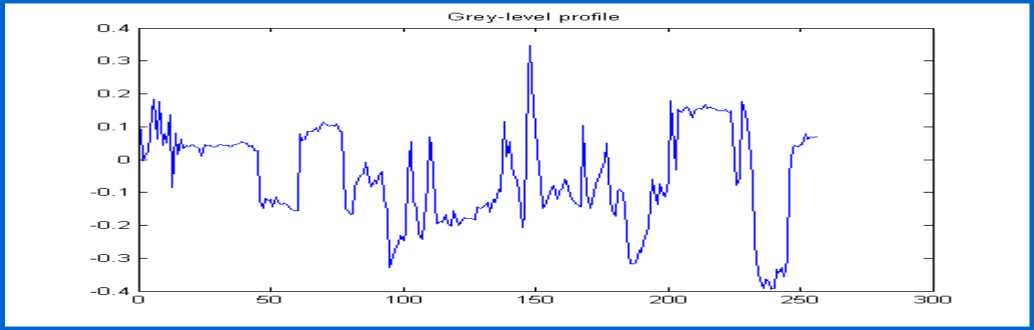

Reconstructed



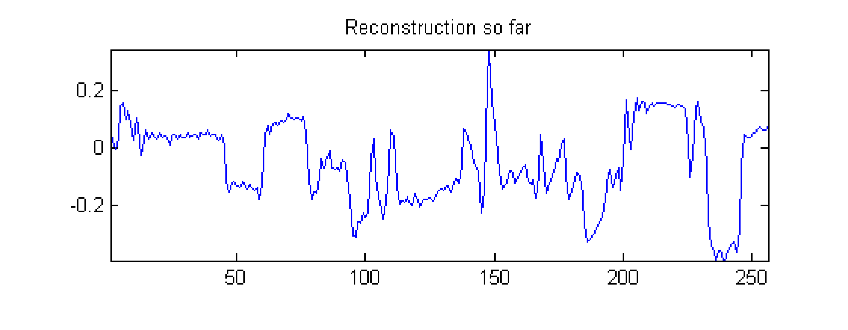

$$f(x) = \sum_{u=0}^{63} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

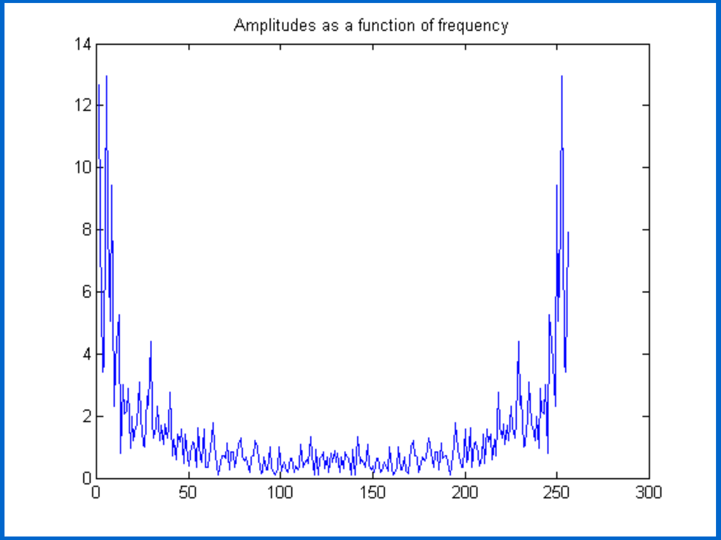
# Representing discontinuities or sharp corners (cont'd)

Original




Reconstructed

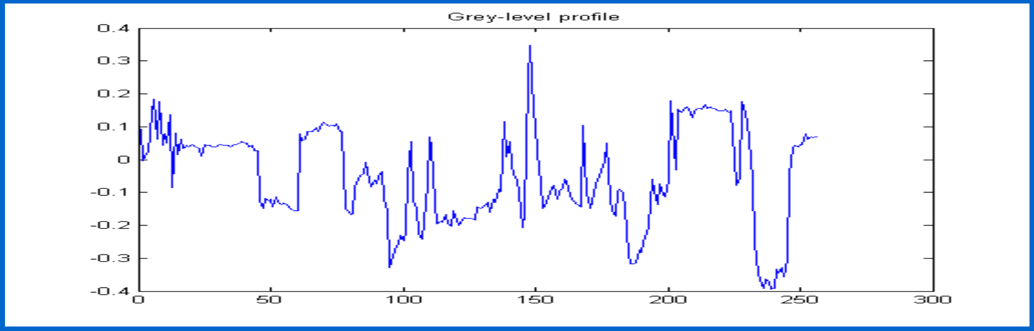


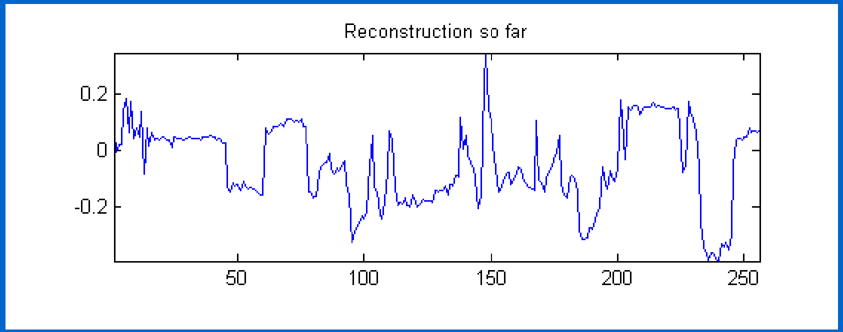

$$f(x) = \sum_{u=0}^{95} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

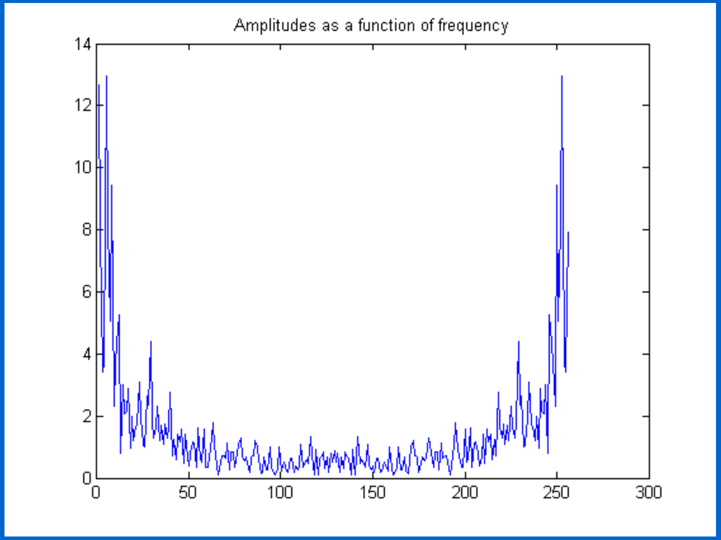
# Representing discontinuities or sharp corners (cont'd)

Original




Reconstructed

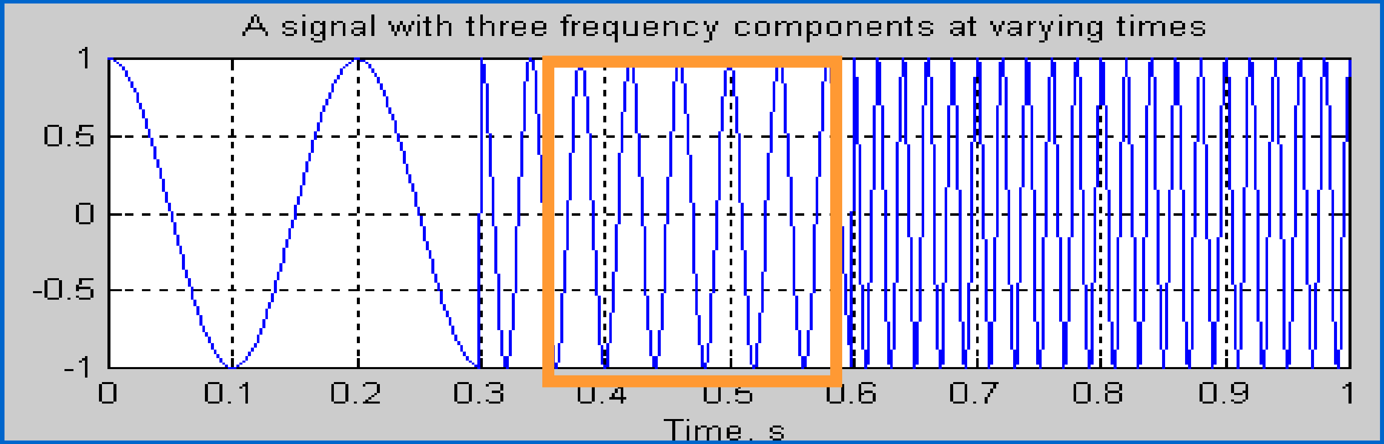



$$f(x) = \sum_{u=0}^{127} F(u) e^{\frac{j2\pi ux}{N}}, x = 0, 1, \ldots, N-1$$

# Short Time Fourier Transform (STFT)

- Segment the signal into narrow time intervals (i.e., narrow enough to be considered stationary) and take the FT of each segment.
- Each FT provides the spectral information of a separate time-slice of the signal, providing simultaneous time and frequency information.



# STFT - Steps

(1) Choose a window function of finite length

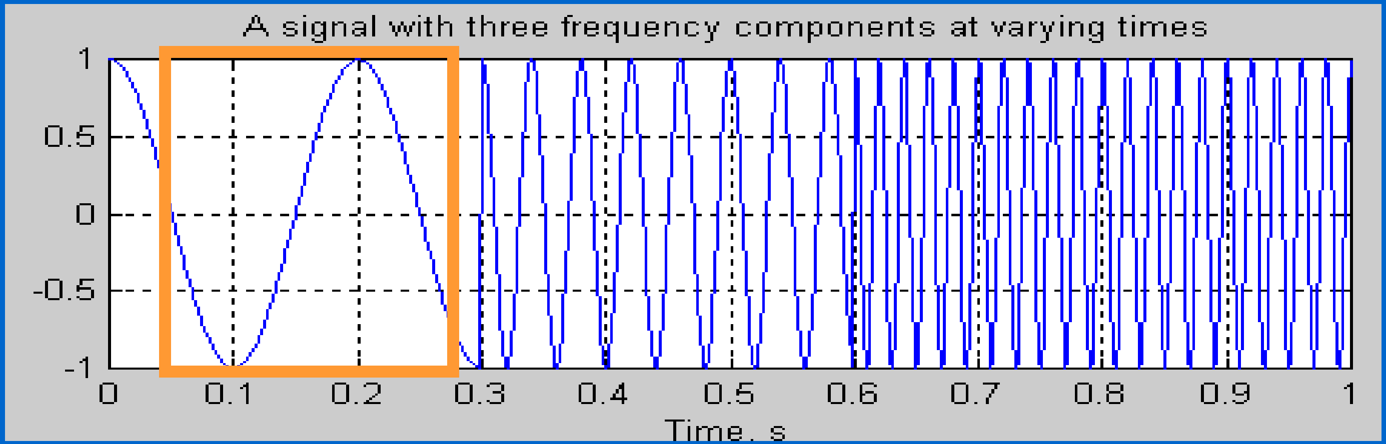
(2) Place the window on top of the signal at t=0

(3) Truncate the signal using this window

(4) Compute the FT of the truncated signal, save results.

(5) Incrementally slide the window to the right

(6) Go to step 3, until window reaches the end of the signal



# STFT - Definition

Time parameter

Frequency parameter

Signal to be analyzed

$$STFT_f^u(t',u) = \int_t [\, f(t) \cdot W(t-t') \,] \cdot e^{-j2\pi ut} dt$$

2D function

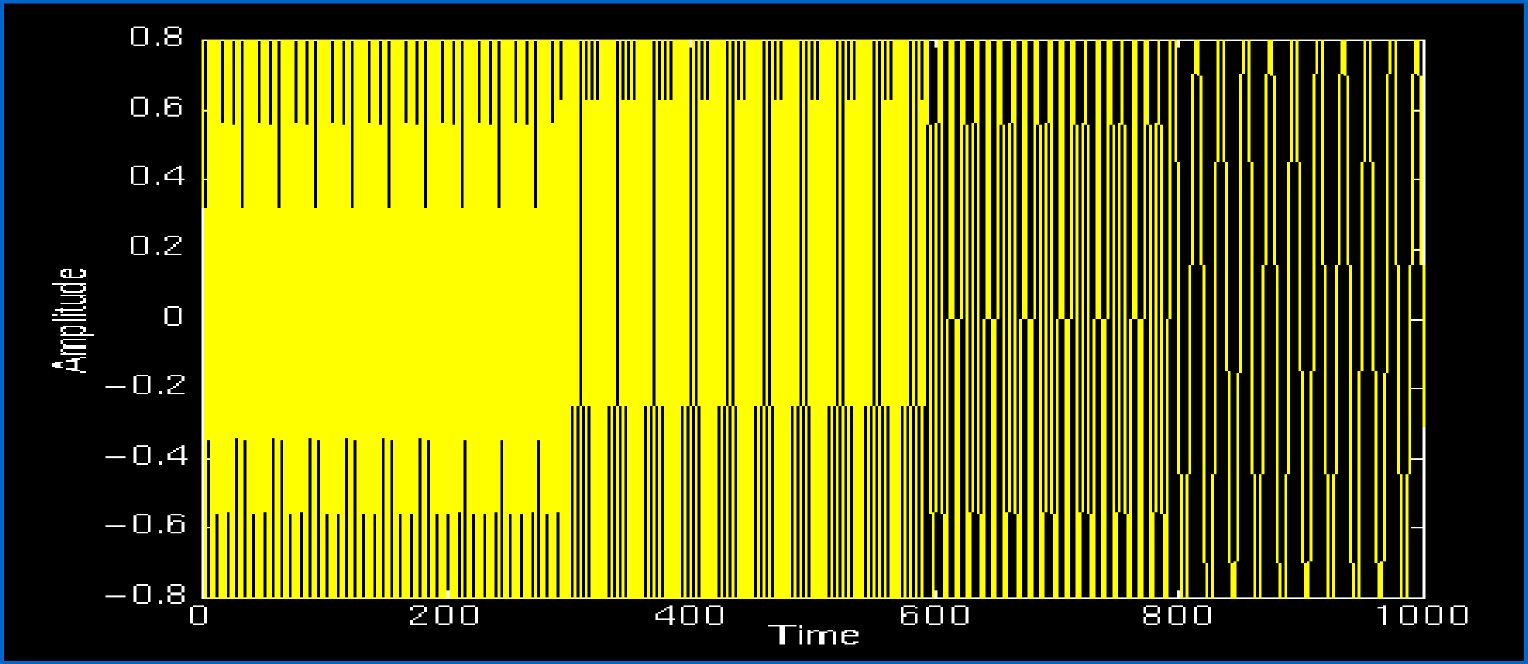
STFT of f(t): computed for each window centered at t=t'

Windowing function

Centered at t=t'

# Example

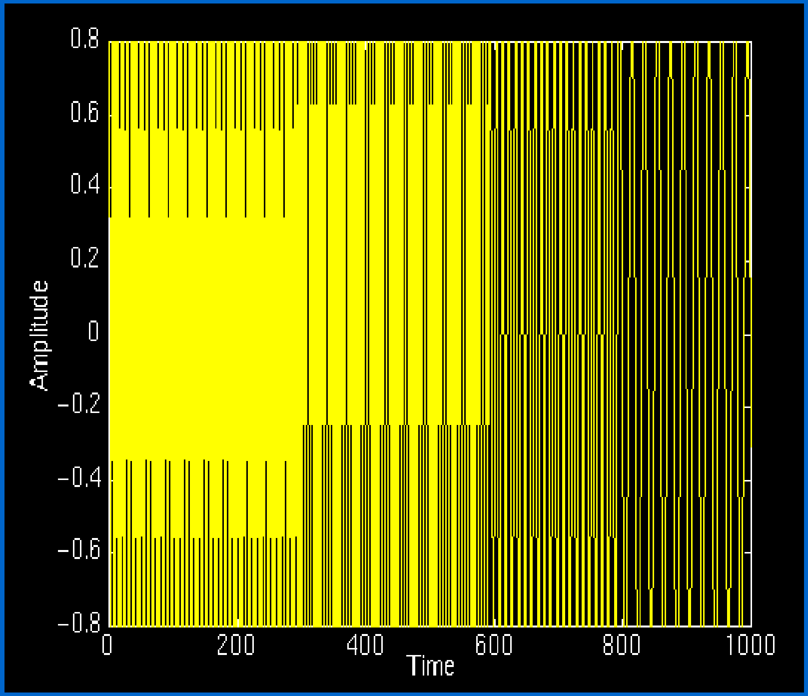
f(t)



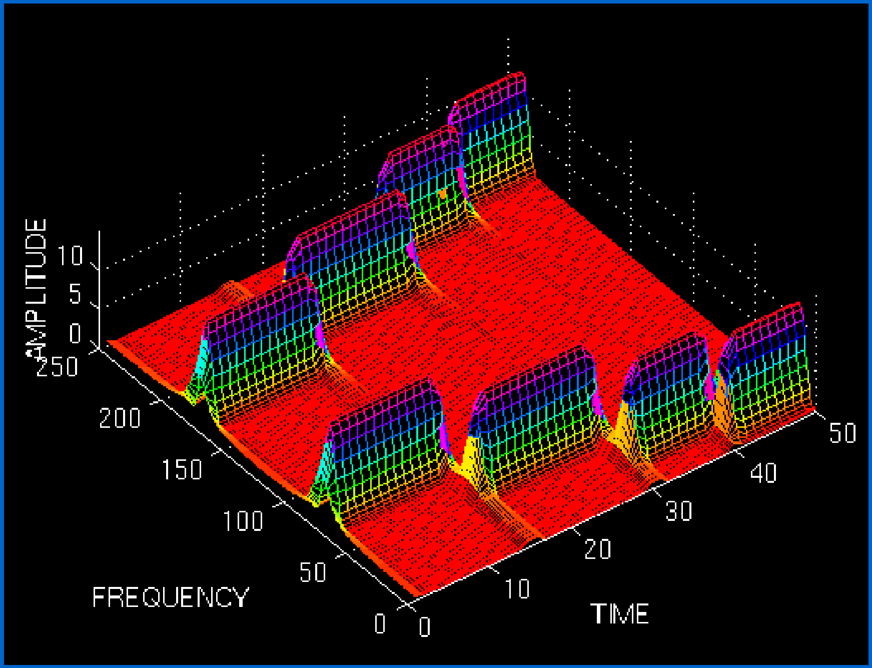

[0 – 300] ms → 75 Hz sinusoid
[300 – 600] ms → 50 Hz sinusoid
[600 – 800] ms → 25 Hz sinusoid
[800 – 1000] ms →10 Hz sinusoid

# Example

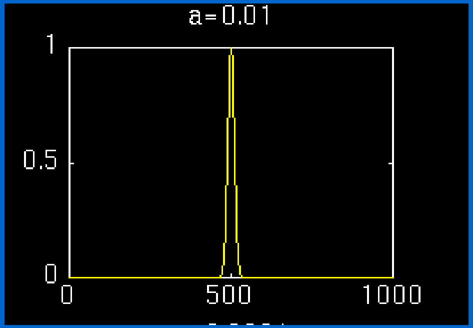
f(t)




$$STFT_f^u(t',u)$$




W(t)




scaled: t/20

# Choosing Window W(t)

- What shape should it have?
  – Rectangular, Gaussian, Elliptic …

- How wide should it be?
  – Window should be narrow enough to ensure that the portion of the signal falling within the window is stationary.
  – But … very narrow windows do not offer good localization in the frequency domain.

# STFT Window Size

$$STFT_f^u(t',u) = \int_t \left[ f(t) \cdot W(t-t') \right] \cdot e^{-j2\pi ut} dt$$

***W(t)* infinitely long**: $\boxed{W(t)=1}$ → STFT turns into FT, providing excellent frequency localization, but no time localization.

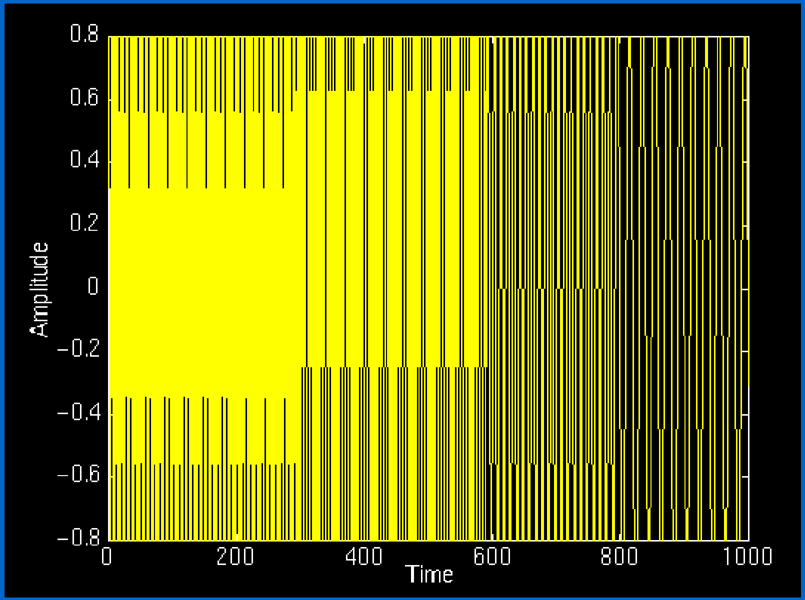
***W(t)* infinitely short**: $\boxed{W(t)=\delta(t)}$ → results in the time signal (with a phase factor), providing excellent time localization but no frequency localization.

$$STFT_f^u(t',u) = \int_t \left[ f(t) \cdot \delta(t-t') \right] \cdot e^{-j2\pi ut} dt = f(t') \cdot e^{-jut'}$$

# STFT Window Size (cont'd)

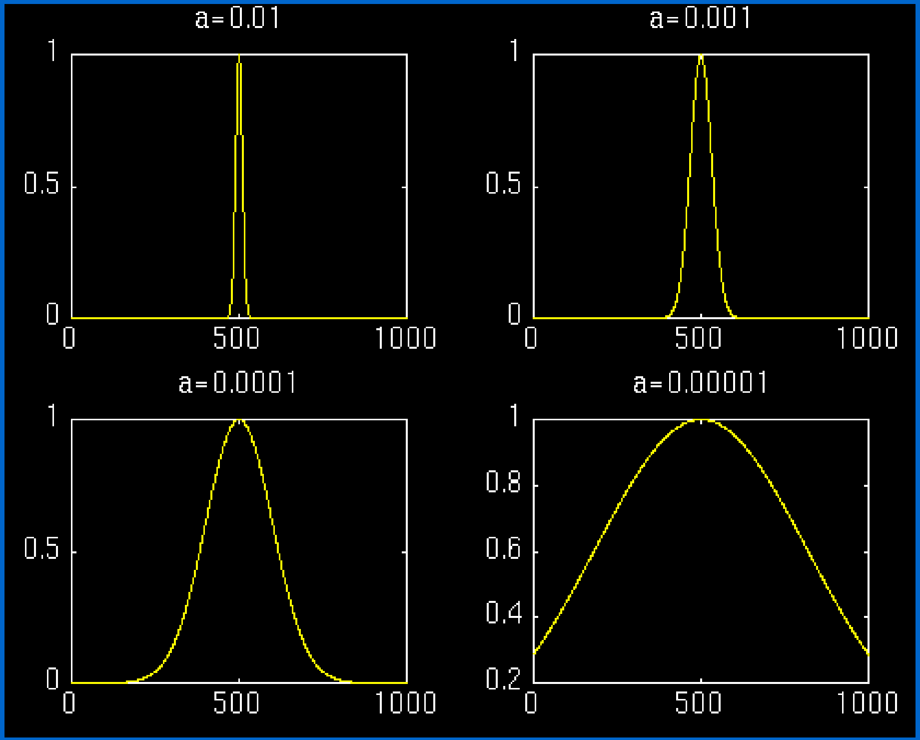
- Wide window → good frequency resolution, poor time resolution.
- Narrow window → good time resolution, poor frequency resolution.
- Wavelets (later): use multiple window sizes.

# Example



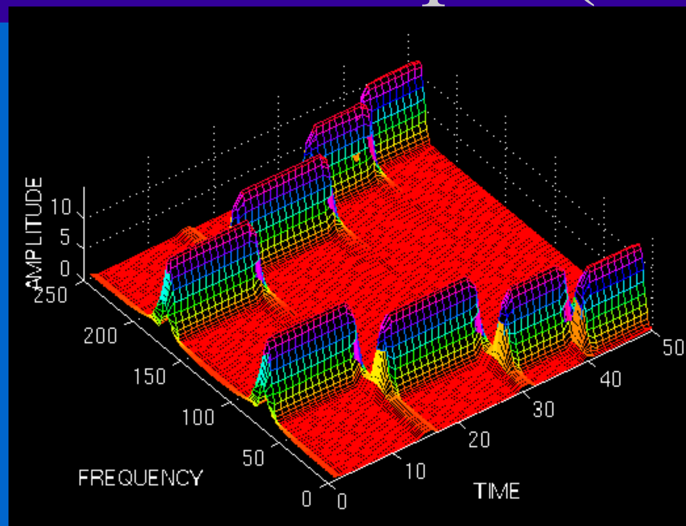

(four frequencies, non-stationary)

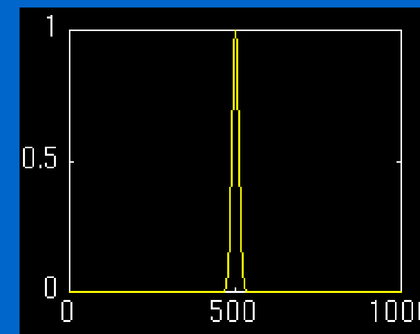
different size windows



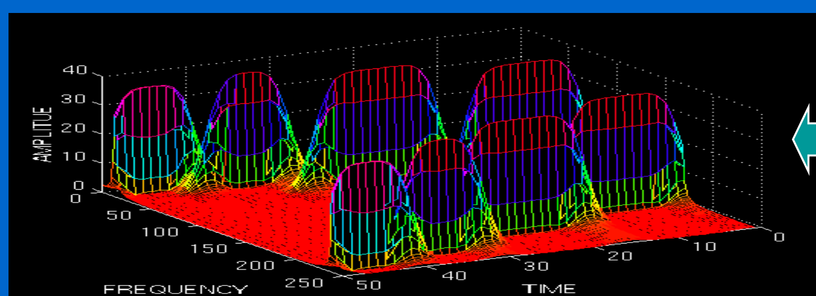
# Example (cont’d)

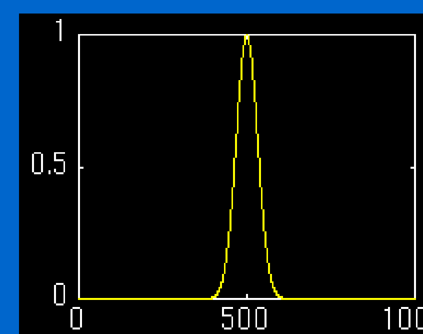
$$STFT_f^u(t',u)$$




$$STFT_f^u(t',u)$$

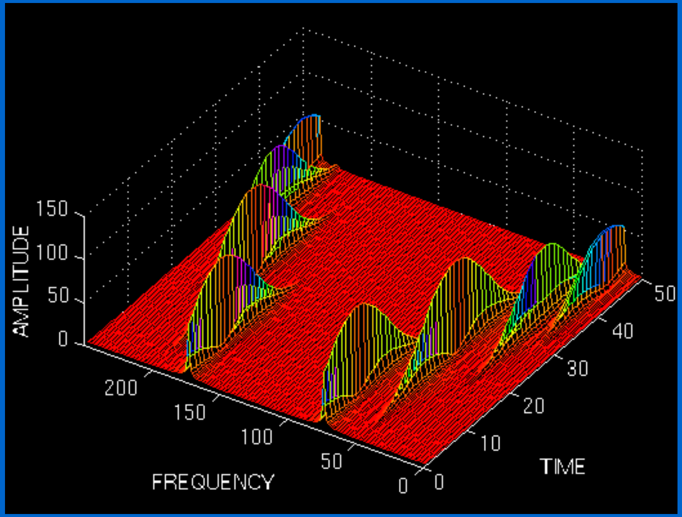


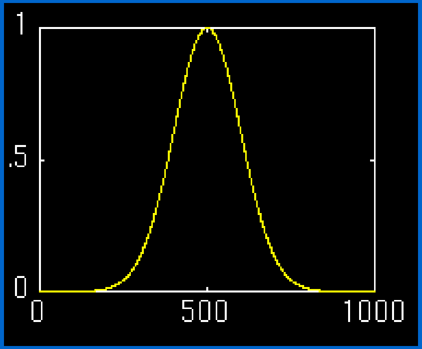




scaled: t/20

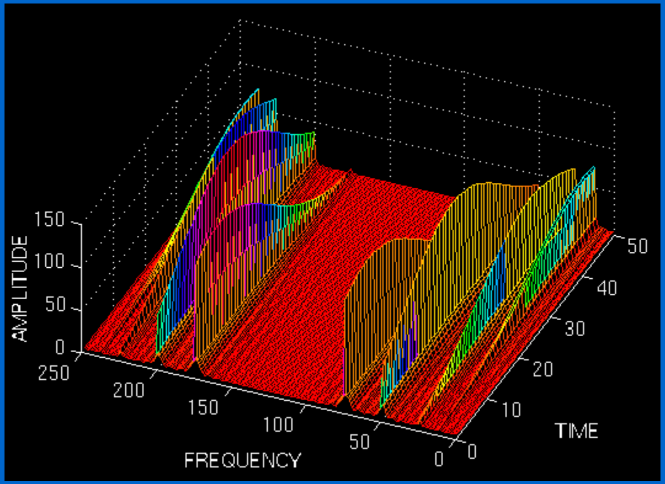
# Example (cont'd)

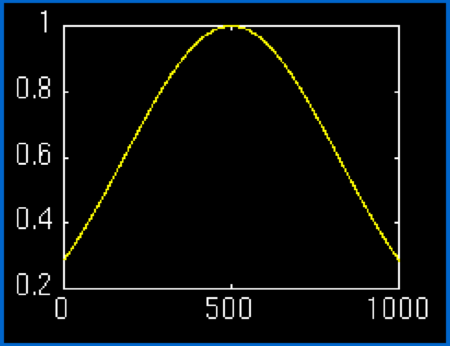
$STFT_f^u(t',u)$



$STFT_f^u(t',u)$



scaled: t/20

# Heisenberg (or Uncertainty) Principle

$$\Delta t \cdot \Delta f \geq \frac{1}{4\pi}$$

**Time resolution:** How well two spikes in time can be separated from each other in the frequency domain.

**Frequency resolution:** How well two spectral components can be separated from each other in the time domain

$$\Delta t \textit{ and } \Delta f \textit{ cannot be made arbitrarily small!}$$

# Heisenberg (or Uncertainty) Principle

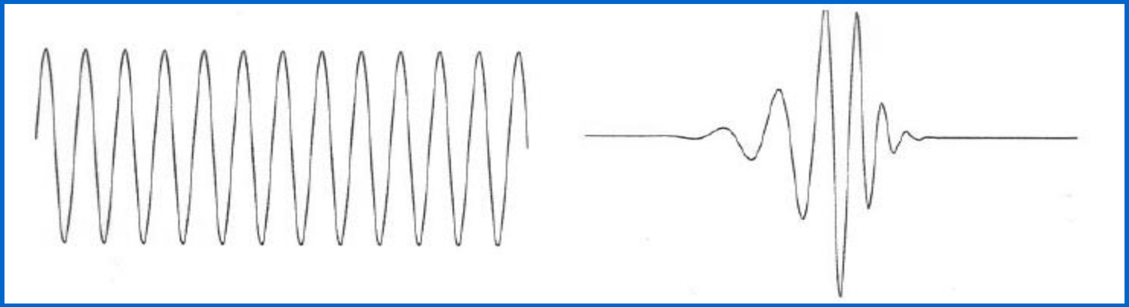
- We cannot know the exact time-frequency representation of a signal.
- We can only know what *interval of frequencies* are present in which *time intervals*.

# Wavelets

# What is a wavelet?

- A function that "waves" above and below the x-axis with the following properties:
  - Varying frequency
  - Limited duration
  - Zero average value
- This is in contrast to sinusoids, used by FT, which have infinite duration and constant frequency.

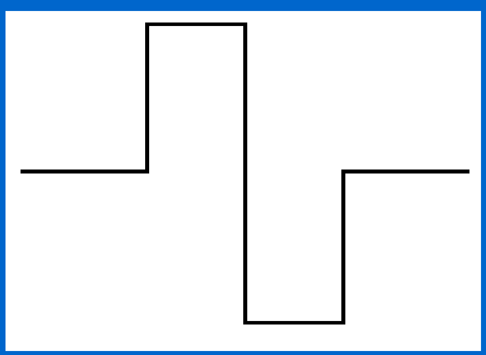
Sinusoid Wavelet



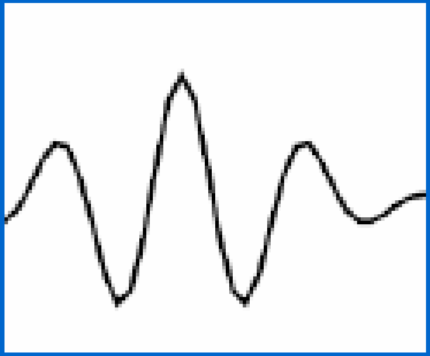
# Types of Wavelets

- There are many different wavelets, for example:

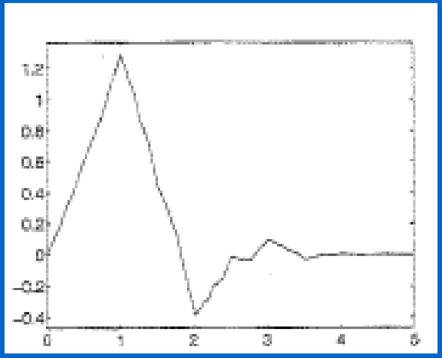
Haar



Morlet



Daubechies



# Basis Functions Using Wavelets

- Like sin( ) and cos( ) functions in the Fourier Transform, wavelets can define a set of basis functions $\psi_k(t)$:

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

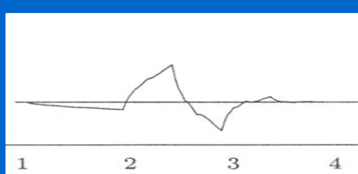
- Span of $\psi_k(t)$: vector space S containing all functions f(t) that can be represented by $\psi_k(t)$.

# Basis Construction – "Mother" Wavelet

The basis can be constructed by applying translations and scalings (stretch/compress) on the "mother" wavelet $\psi(t)$:

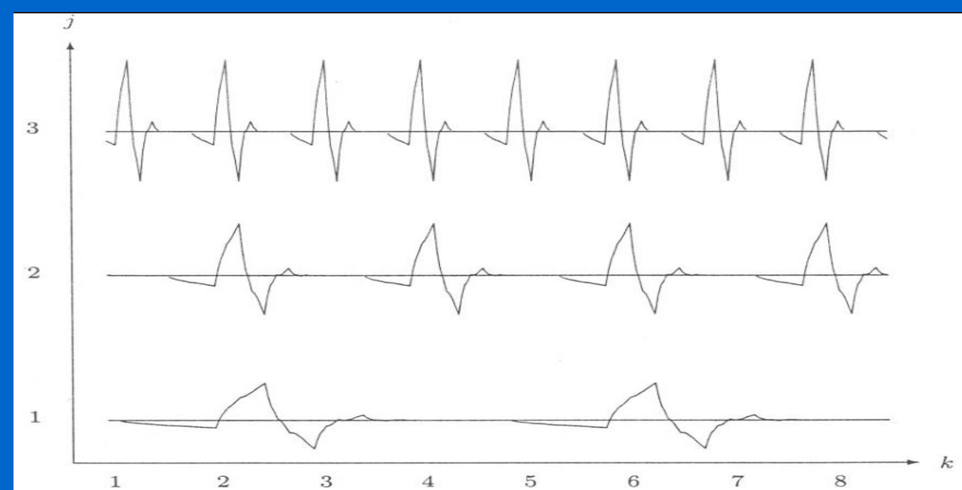
$$\psi(s,\tau,t)=\frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right)$$

Example:




$\psi(t)$

scale




translate

# Basis Construction - Mother Wavelet

- It is convenient to take special values for s and τ in defining the wavelet basis: $\mathbf{s = 2^{-j}}$ and $\boldsymbol{\tau} \mathbf{= k \, . \, 2^{-j}}$

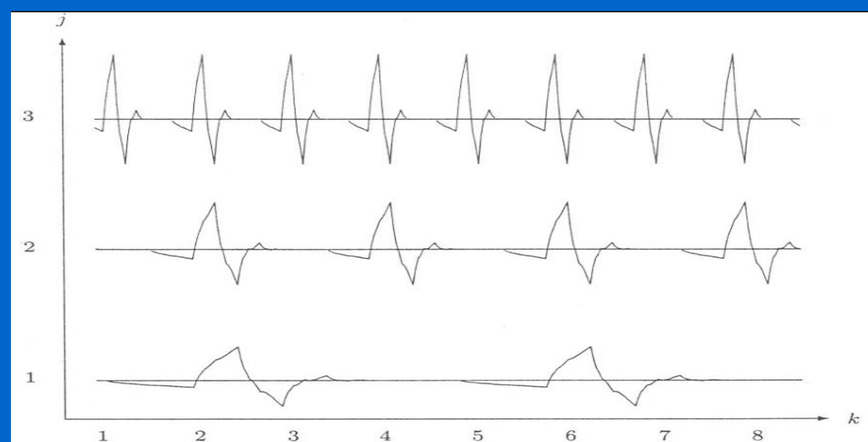
(dyadic/octave grid)

$$\psi(s,\tau,t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right) = \frac{1}{\sqrt{2^{-j}}}\psi\left(\frac{t-k.2^{-j}}{2^{-j}}\right) = 2^{\frac{j}{2}}\psi(2^{j}t-\mathrm{k}) \boxed{= \psi_{jk}(t)}$$

scale $=1/2^{j}$

(1/frequency)

↑ j




k ⟶

# Continuous Wavelet Transform (CWT)

translation parameter
(measure of time)

scale parameter
(measure of frequency)

scale $=1/2^{j}$
(1/frequency)

Forward CWT:

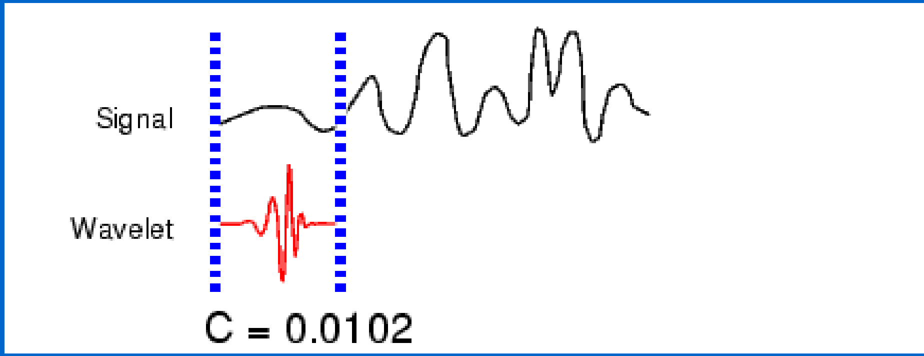
$$C(\tau, s) = \frac{1}{\sqrt{s}} \int_t f(t)\, \psi^*\left(\frac{t-\tau}{s}\right) dt$$

normalization constant

mother wavelet (i.e., window function)

# Illustrating CWT

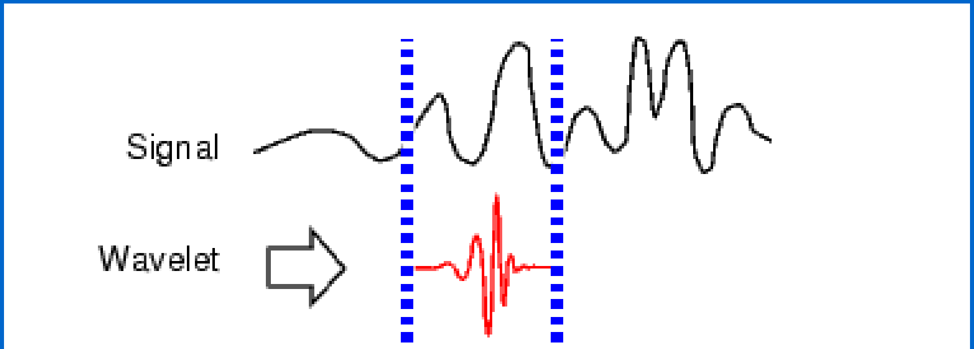
1. Take a wavelet and compare it to a section at the start of the original signal.
2. Calculate a number, C, that represents how closely correlated the wavelet is with this section of the signal. The higher C is, the more the similarity.




$$C(\tau, s) = \frac{1}{\sqrt{s}} \int_t f(t) \psi^*\left(\frac{t-\tau}{s}\right) dt$$

# Illustrating CWT (cont'd)

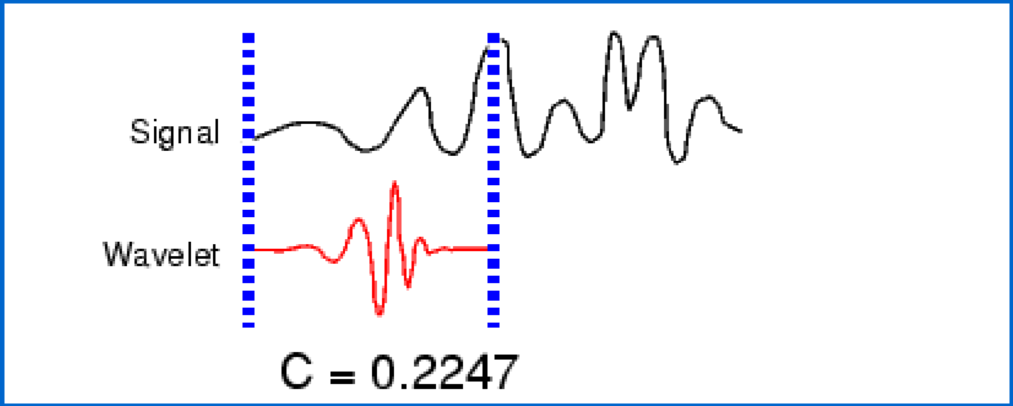
3. Shift the wavelet to the right and repeat step 2 until you've covered the whole signal.




$$C(\tau, s) = \frac{1}{\sqrt{s}} \int_t f(t)\,\psi^*\left(\frac{t-\tau}{s}\right) dt$$

# Illustrating CWT (cont'd)

4. Scale the wavelet and go to step 1.

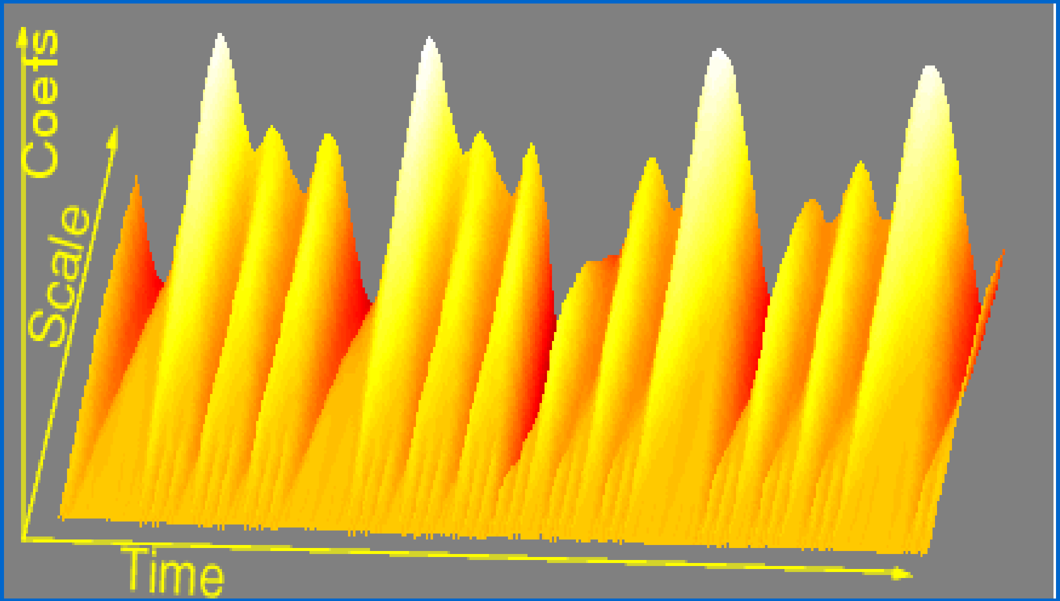



$$C(\tau, s) = \frac{1}{\sqrt{s}} \int_t f(t) \psi^* \left( \frac{t-\tau}{s} \right) dt$$

5. Repeat steps 1 through 4 for all scales.

# Visualize CTW Transform

- Wavelet analysis produces a time-scale view of the input signal or image.




$$C(\tau, s) = \frac{1}{\sqrt{s}} \int_t f(t)\psi^*\left(\frac{t-\tau}{s}\right) dt$$

# Continuous Wavelet Transform (cont'd)

Forward CWT:

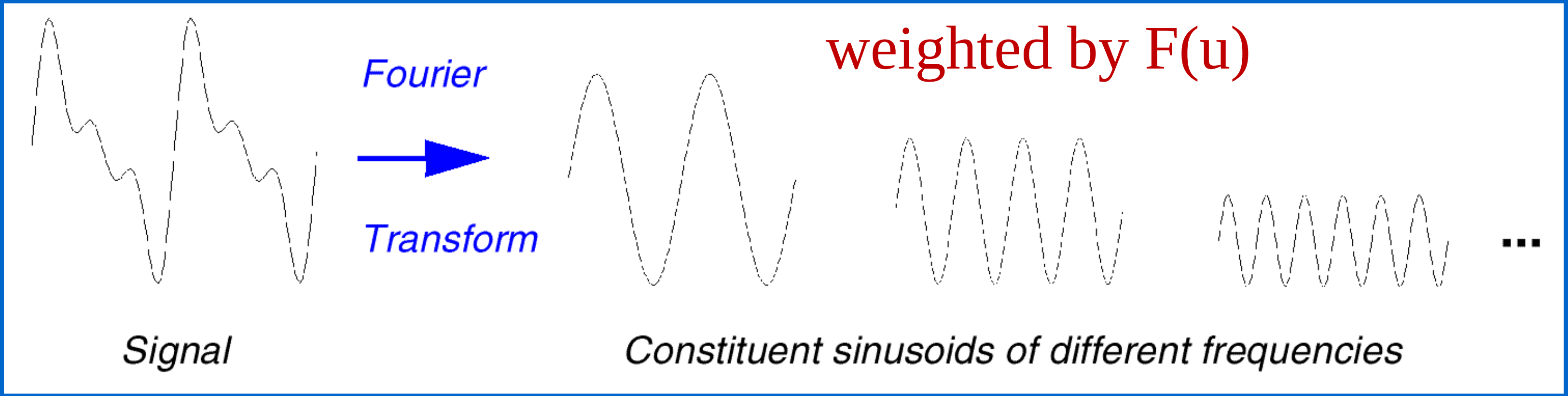
$$C(\tau, s) = \frac{1}{\sqrt{s}} \int_t f(t) \psi^* \left( \frac{t - \tau}{s} \right) dt$$

Inverse CWT:

$$f(t) = \frac{1}{\sqrt{s}} \int_\tau \int_s C(\tau, s) \psi \left( \frac{t - \tau}{s} \right) d\tau ds$$

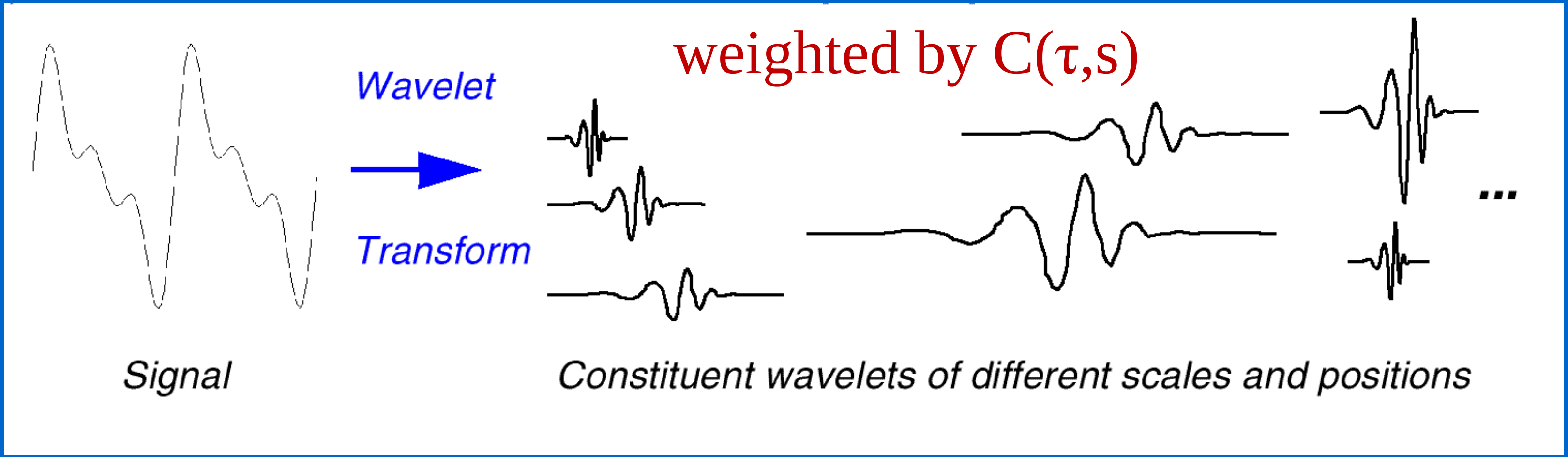
Note the double integral!

# Fourier Transform vs Wavelet Transform




$$f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du$$

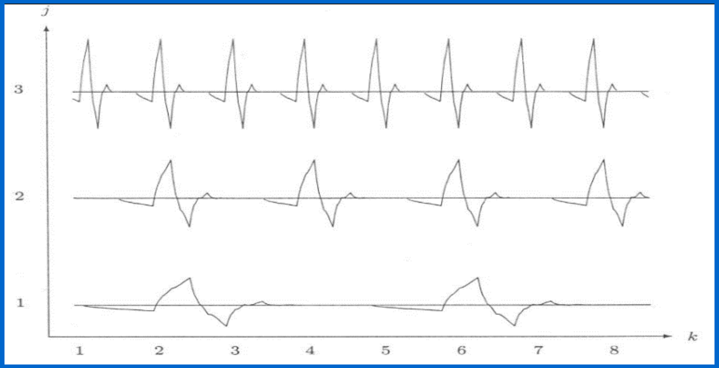
# Fourier Transform vs Wavelet Transform

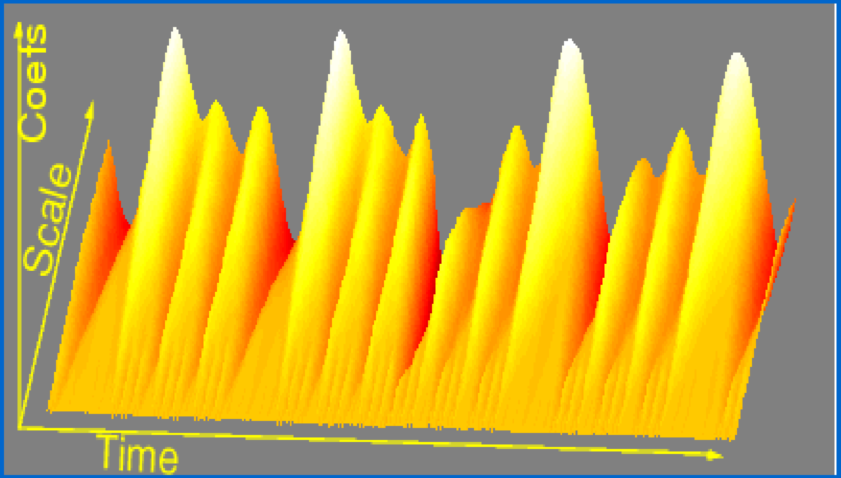



$$f(t) = \frac{1}{\sqrt{s}} \int_{\tau} \int_{s} C(\tau, s) \psi\left(\frac{t-\tau}{s}\right) d\tau ds$$

# Properties of Wavelets

- Simultaneous localization in time and scale
  - The location of the wavelet allows to explicitly represent the location of events in time.
  - The shape of the wavelet allows to represent different detail or resolution.



# Properties of Wavelets (cont'd)

- Sparsity: for functions typically found in practice, many of the coefficients in a wavelet representation are either zero or very small.

$$f(t) = \frac{1}{\sqrt{s}} \int_{\tau} \int_{s} C(\tau, s) \psi\left(\frac{t-\tau}{s}\right) d\tau ds$$

# Properties of Wavelets (cont'd)

$$f(t) = \frac{1}{\sqrt{s}} \int_{\tau} \int_{s} C(\tau, s) \psi\left(\frac{t-\tau}{s}\right) d\tau ds$$

- Adaptability: Can represent functions with discontinuities or corners more efficiently.

- Linear-time complexity: many wavelet transformations can be accomplished in O(N) time.

# Discrete Wavelet Transform (DWT)

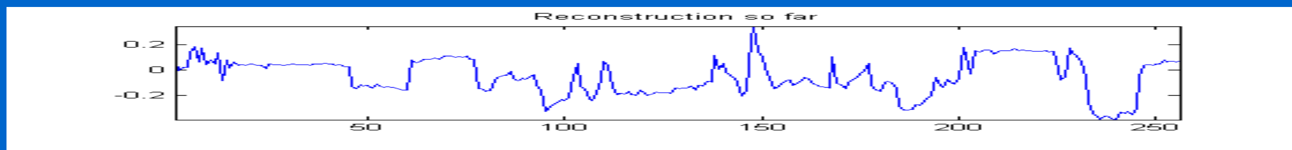
$$a_{jk} = \sum_{t} f(t)\psi_{jk}^{*}(t)$$ (forward DWT)

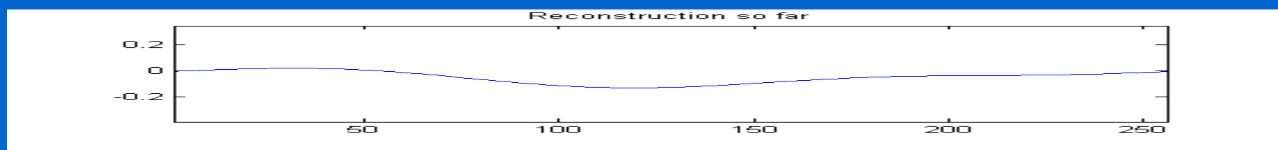
$$f(t) = \sum_{k}\sum_{j} a_{jk}\psi_{jk}(t)$$ (inverse DWT)

where $$\psi_{jk}(t) = 2^{j/2}\psi\left(2^{j}t - k\right)$$

# Multiresolution Representation Using Wavelets

$f(t)$

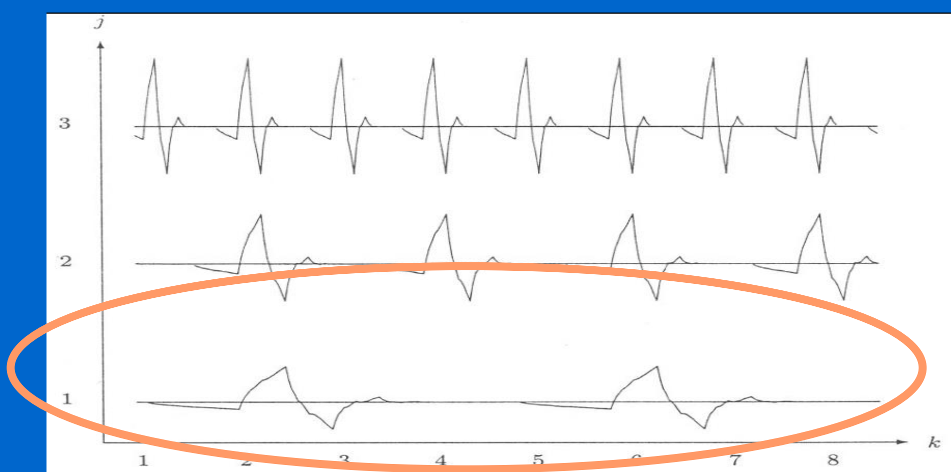



fine details

wider, large translations

$$f(t) = \sum_k \sum_j a_{jk} \psi_{jk}(t)$$

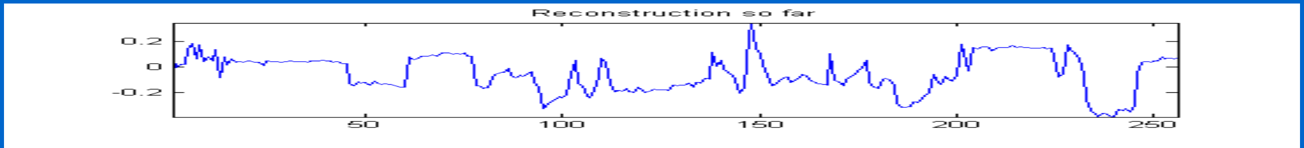


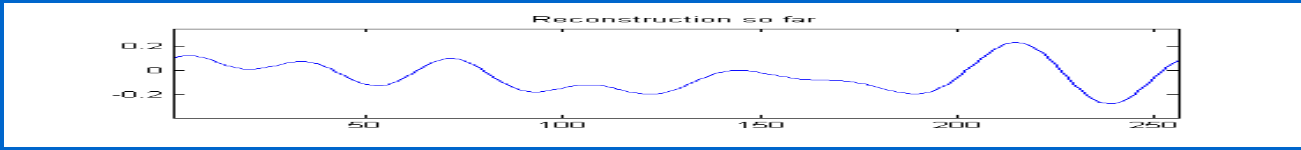

j

coarse details

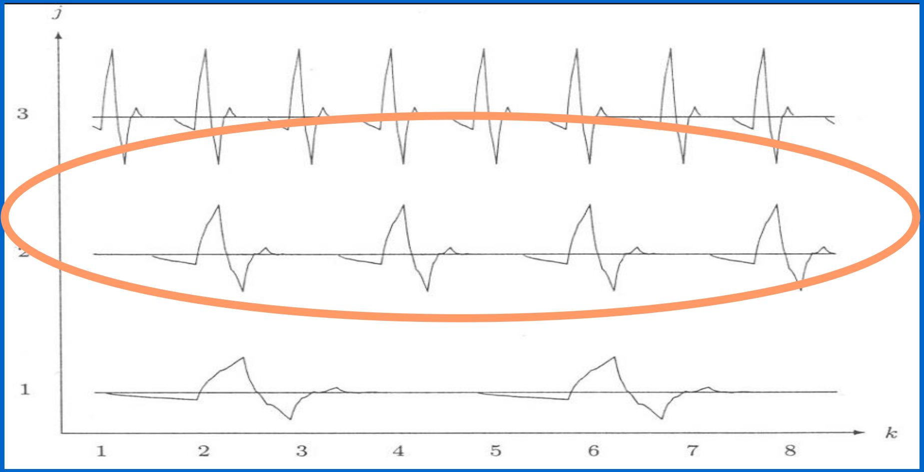
# Multiresolution Representation Using Wavelets

$f(t)$




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



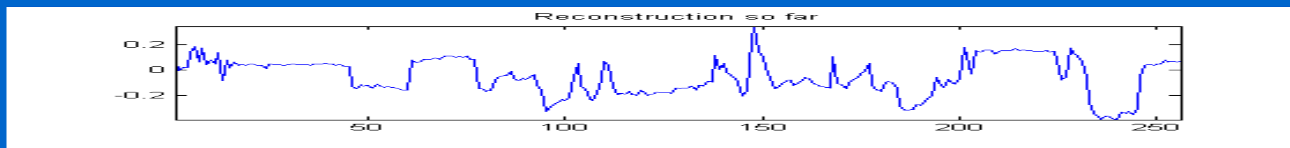

fine details

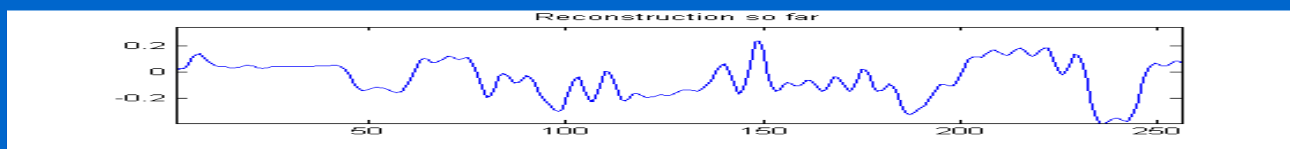
j

coarse details

# Multiresolution Representation Using Wavelets

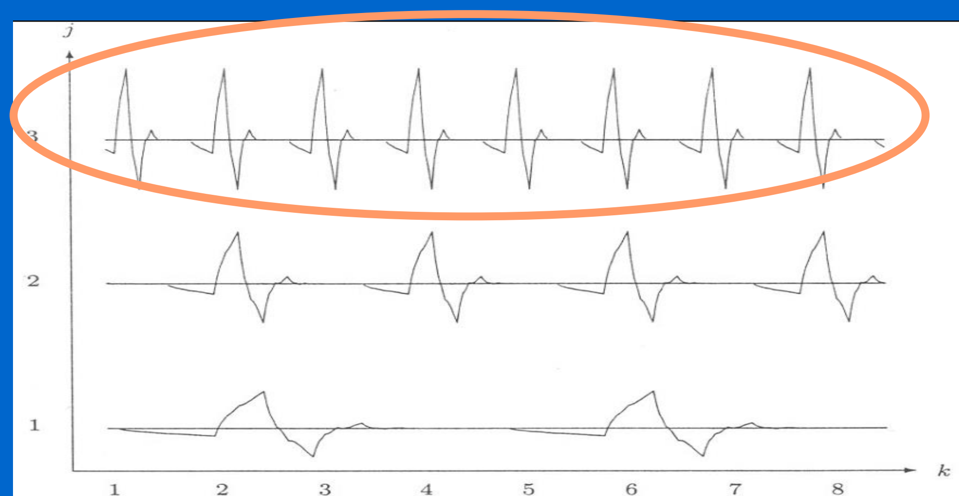
$f(t)$




fine details

narrower, small translations



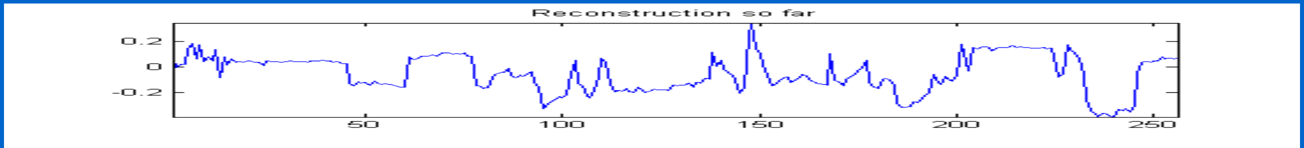

j

$$f(t)=\sum_k \sum_j a_{jk}\psi_{jk}(t)$$

coarse details

# Multiresolution Representation Using Wavelets

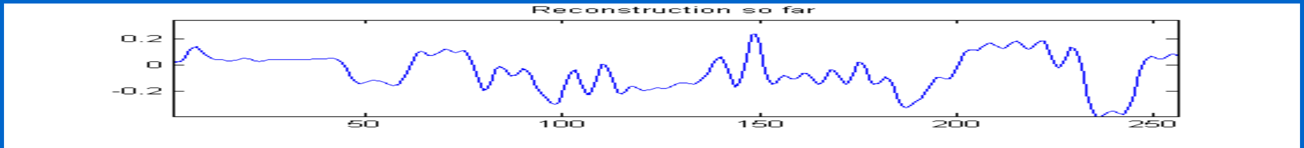
$f(t)$




high resolution

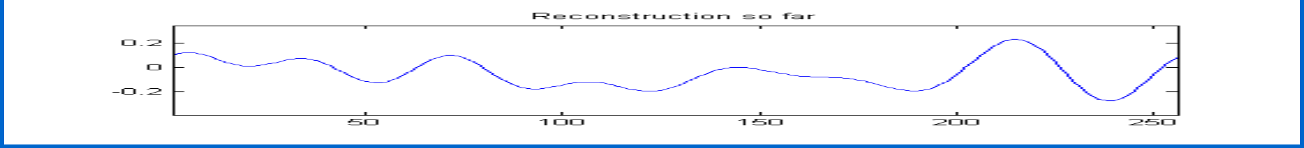
(more details)

$\hat{f}_1(t)$



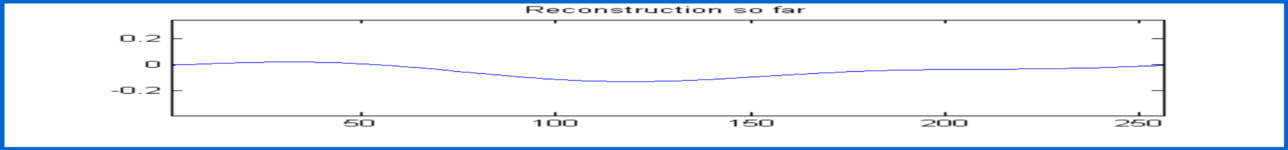

j

$\hat{f}_2(t)$




• • •

$\hat{f}_s(t)$



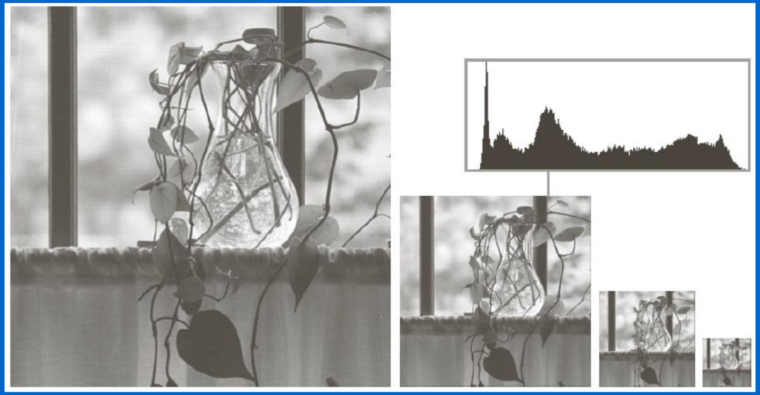

low resolution

(less details)

$$f(t) = \sum_k \sum_j a_{jk} \psi_{jk}(t)$$

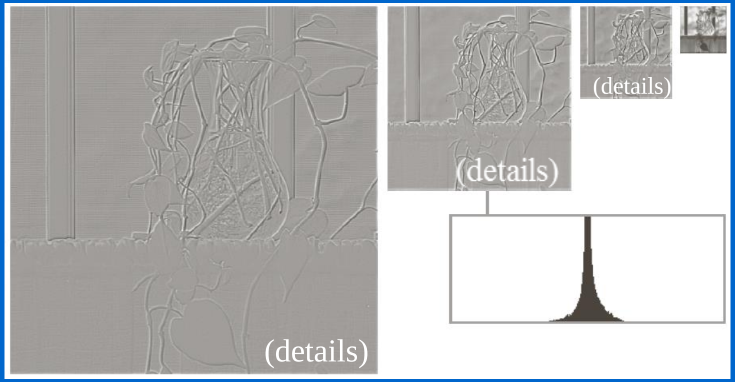
# Pyramidal Coding - Revisited

Approximation Pyramid



(with sub-sampling)

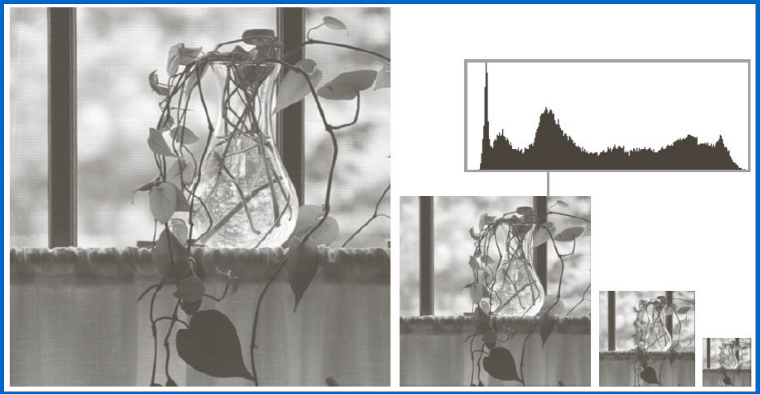
# Pyramidal Coding - Revisited




(with sub-sampling)

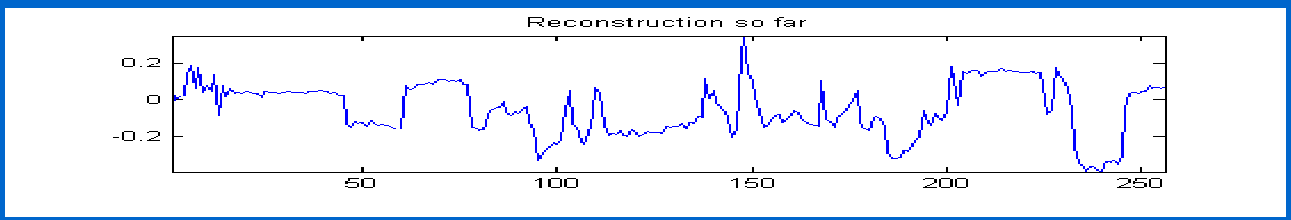
Prediction Residual Pyramid

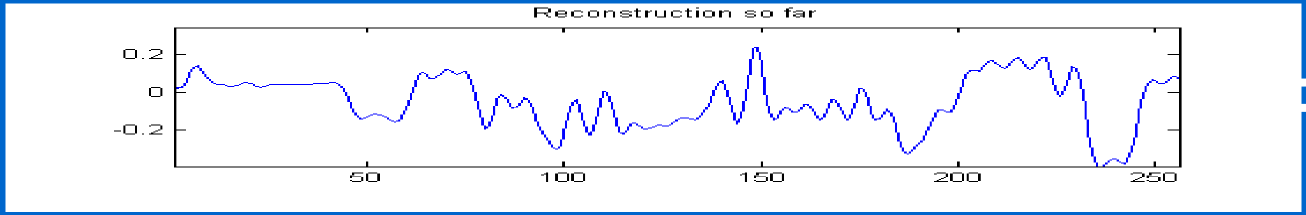
reconstruct



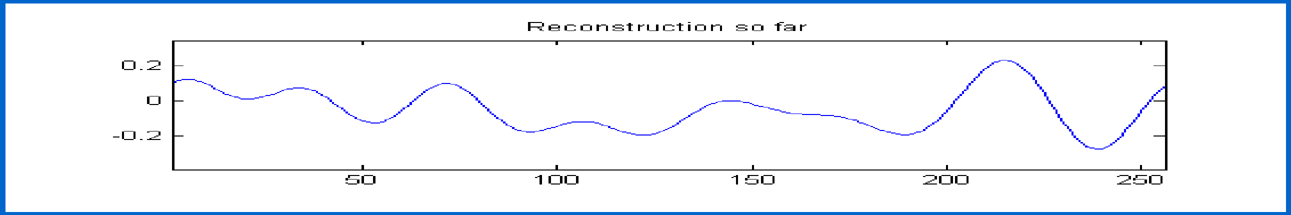
Approximation Pyramid

# Efficient Representation Using "Details"

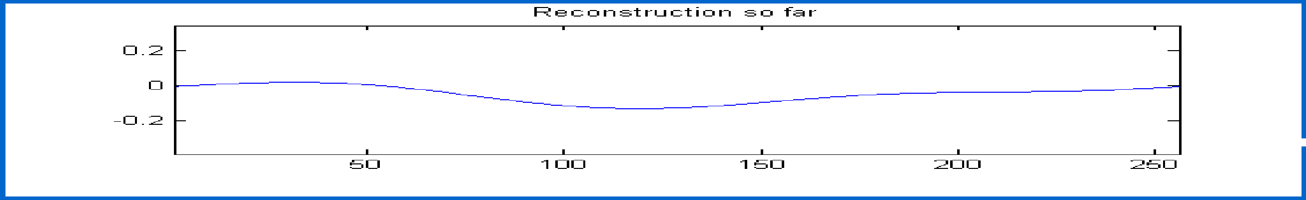



details $D_3$




details $D_2$

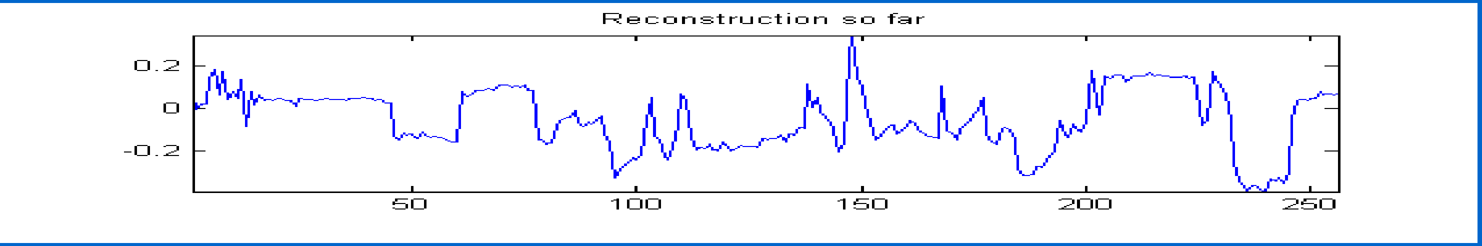



details $D_1$

$L_0$

(without sub-sampling)

# Efficient Representation Using Details (cont'd)



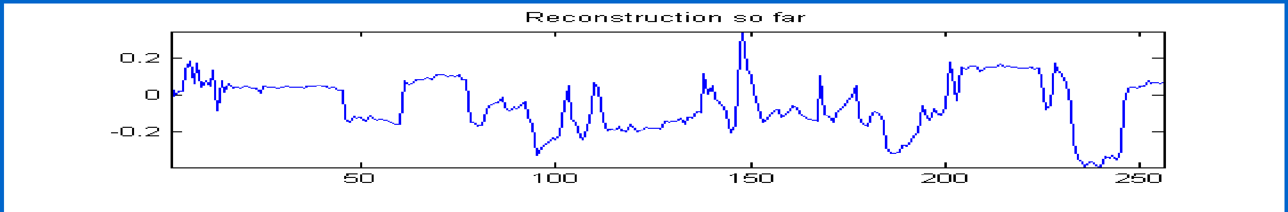

representation: $L_0 \; D_1 \; D_2 \; D_3$

in general: $L_0 \; D_1 \; D_2 \; D_3 \ldots D_J$

**A wavelet representation of a function consists of**
**(1) a coarse overall approximation**
**(2) detail coefficients that influence the function at various scales**

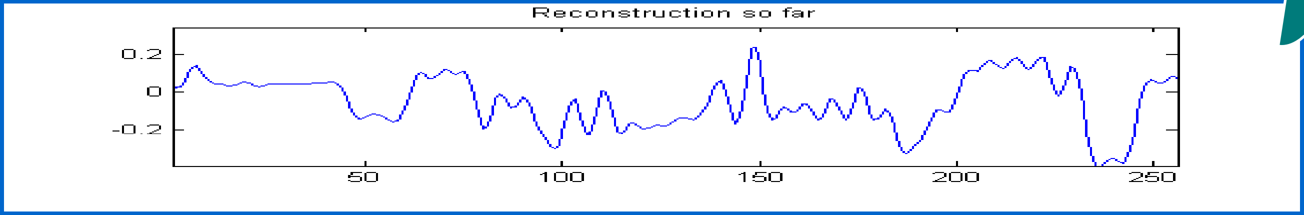
# Reconstruction (synthesis)

$H_3 = H_2$ & $D_3$

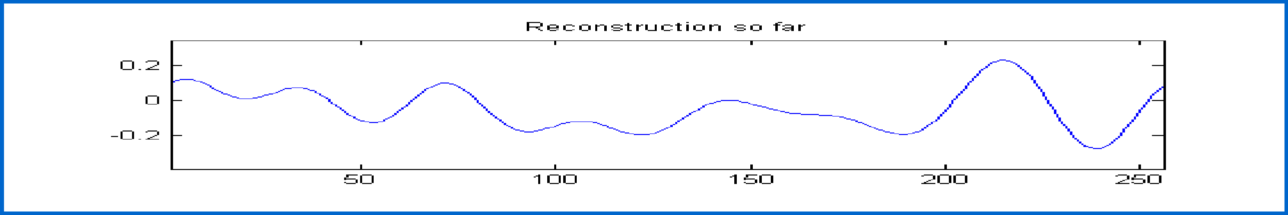



details $D_3$

$H_2 = H_1$ & $D_2$



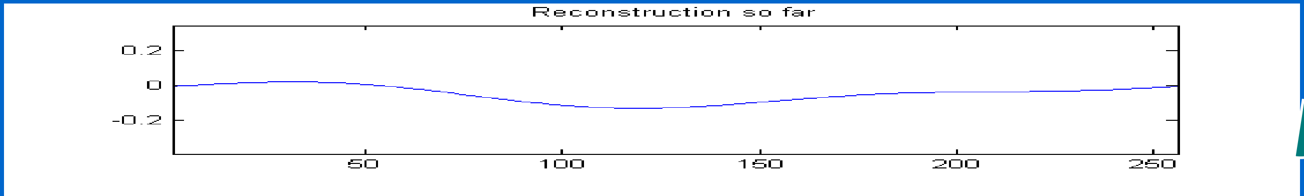

details $D_2$




$H_1 = L_0$ & $D_1$

details $D_1$

$L_0$




(without sub-sampling)

# Example - Haar Wavelets

- Suppose we are given a 1D "image" with a resolution of 4 pixels:

$$[9\ 7\ 3\ 5]$$

- The Haar wavelet transform is the following:

$$[6\quad 2\quad 1\quad -1]$$ (with sub-sampling)

$L_0\ D_1\ D_2\ D_3$

# Example - Haar Wavelets (cont'd)

- Start by averaging and subsampling the pixels together (pairwise) to get a new lower resolution image:

$$[9\ 7\ 3\ 5] \rightarrow [8\ \ 4]$$

- To recover the original four pixels from the two averaged pixels, store some *detail coefficients*.

| *Resolution* | *Averages* | *Detail Coefficients* |
| --- | --- | --- |
| 1 | $[9\ \ 7\ \ 3\ \ 5]$ | $[]$ |
| 2 | $[8\ \ 4]$ | $[1\ \ -1]$ |

# Example - Haar Wavelets (cont'd)

- Repeating this process on the averages (i.e., low resolution image) gives the full decomposition:

| *Resolution* | *Averages* | *Detail Coefficients* |
|---|---|---|
| 1 | $[9 \;\; 7 \;\; 3 \;\; 5]$ | $[]$ |
| 2 | $[8 \;\; 4]$ | $[1 \;\; -1]$ |
| 4 | $[6]$ | $[2]$ |

Haar decomposition: $[6 \;\; 2 \;\; 1 \;\; -1]$

# Example - Haar Wavelets (cont'd)

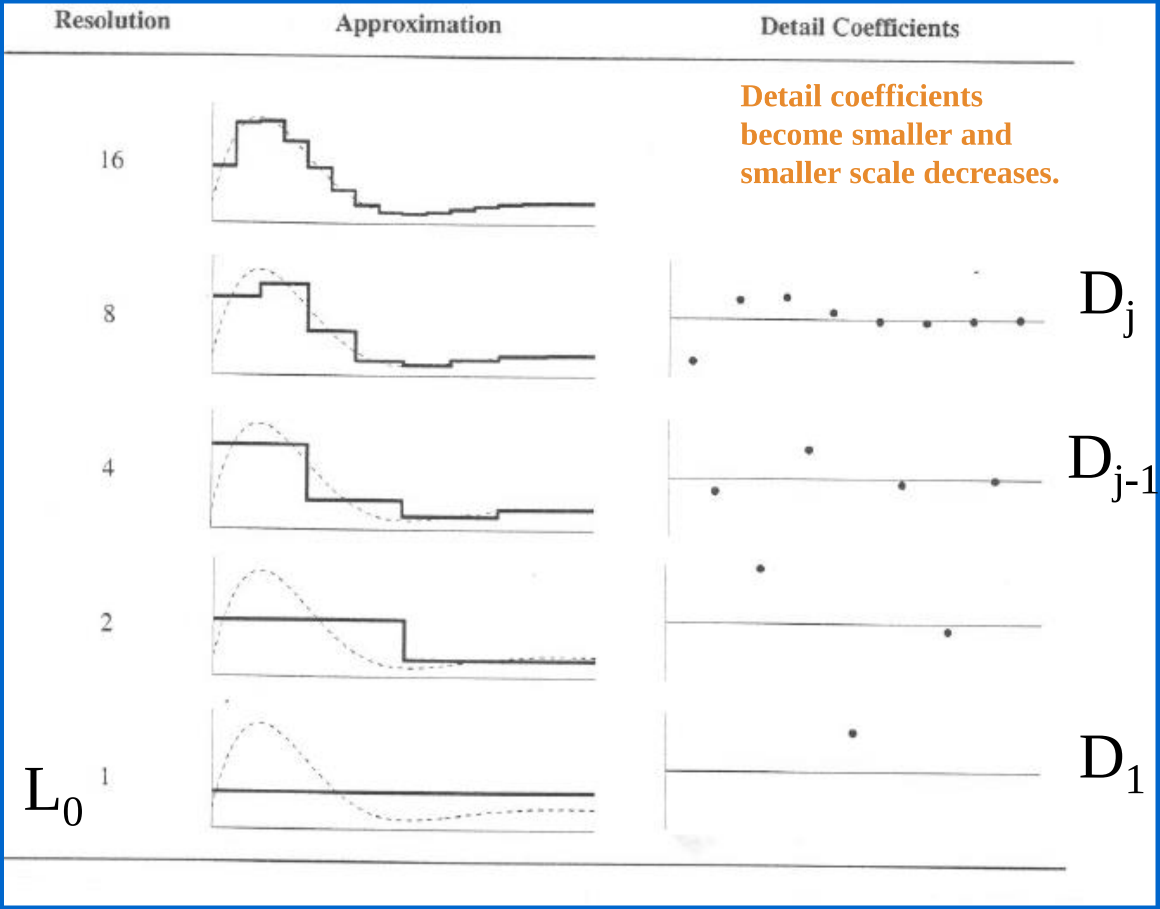
- The original image can be reconstructed by adding or subtracting the detail coefficients from the lower-resolution representations.

$$[6 \;\; 2 \;\; 1 \;\; -1]$$

$L_0 \; D_1 \; D_2 \; D_3$

$$[6] \xrightarrow{2} [8 \;\; 4] \xrightarrow{1 \; -1} [9 \;\; 7 \;\; 3 \;\; 5]$$

# Example - Haar Wavelets (cont'd)



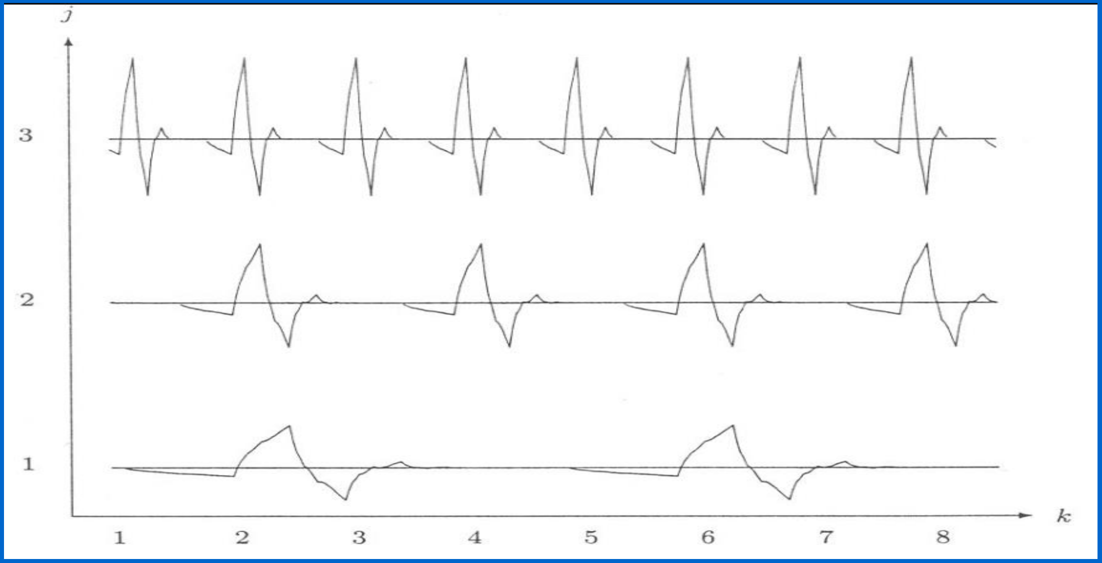

How should we compute the detail coefficients $D_j$ ?

# Multiresolution Conditions

- If a set of functions V can be represented by a weighted sum of $\psi(2^j t - k)$, then a **larger set**, including V, can be represented by a weighted sum of $\psi(2^{j+1} t - k)$.

$\psi(2^{j+1} t - k)$

$\psi(2^j t - k)$



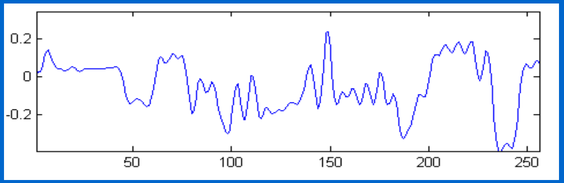

high resolution

j

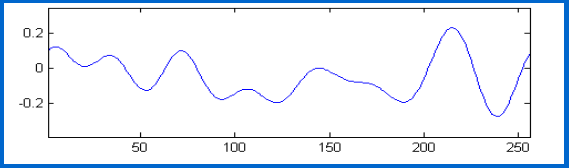
low resolution

# Multiresolution Conditions (cont'd)

$V_{j+1}$: span of $\psi(2^{j+1}t - k)$: $$f_{j+1}(t) = \sum_k b_k \psi_{(j+1)k}(t)$$




$V_j$: span of $\psi(2^j t - k)$: $$f_j(t) = \sum_k a_k \psi_{jk}(t)$$

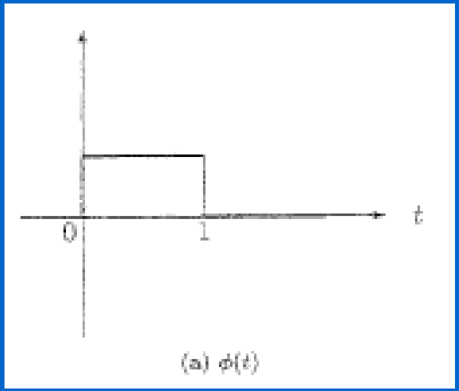


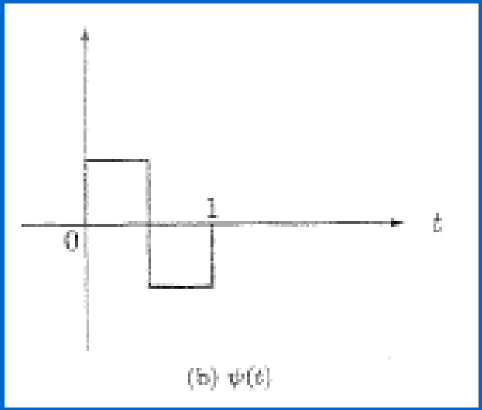

$$V_j \subseteq V_{j+1}$$

# 1D Haar Wavelets

- Haar scaling and wavelet functions:

$\varphi(t)$ $\psi(t)$




computes average (low pass)

computes details (high pass)

# 1D Haar Wavelets (cont'd)

Let's consider the spaces corresponding to different resolution 1D images:

| | | | |
|---|---|---|---|
| $V_0$ | ● | 1-pixel | (j=0) |
| $V_1$ | ● ● | 2-pixel | (j=1) |
| $V_2$ | ● ● ● ● | 4-pixel | (j=2) |

etc.

# 1D Haar Wavelets (cont'd)

j=0

- $V_0$ represents the space of 1-pixel ($2^0$-pixel) images

- Think of a 1-pixel image as a function that is constant over [0,1)

Example:

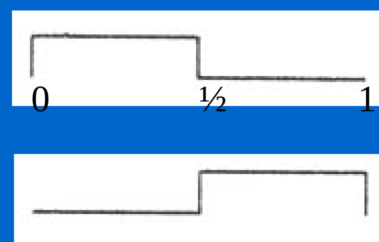

width: 1

# 1D Haar Wavelets (cont'd)

j=1

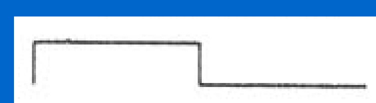
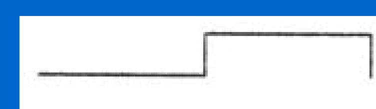
- $V_1$ represents the space of all 2-pixel ($2^1$-pixel) images
- Think of a 2-pixel image as a function having $2^1$ equal-sized constant pieces over the interval [0, 1).

Example:




width: 1/2

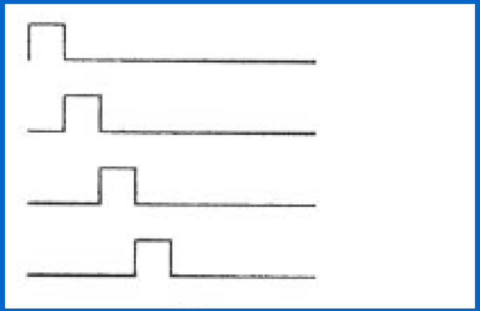
Note that: $\boxed{V_0 \subset V_1}$

e.g.,  =  + 

# 1D Haar Wavelets (cont'd)

- $V_j$ represents all the $2^j$-pixel images
- Functions having $2^j$ equal-sized constant pieces over interval [0,1).

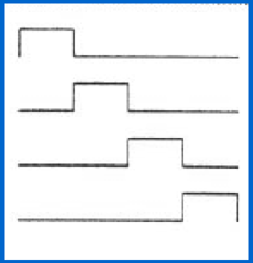
Example:



width: $1/2^j$

Note that:

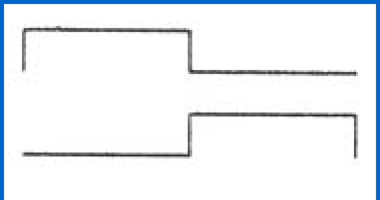
$V_{j-1}$



$\in V_j$

width: $1/2^{j-1}$

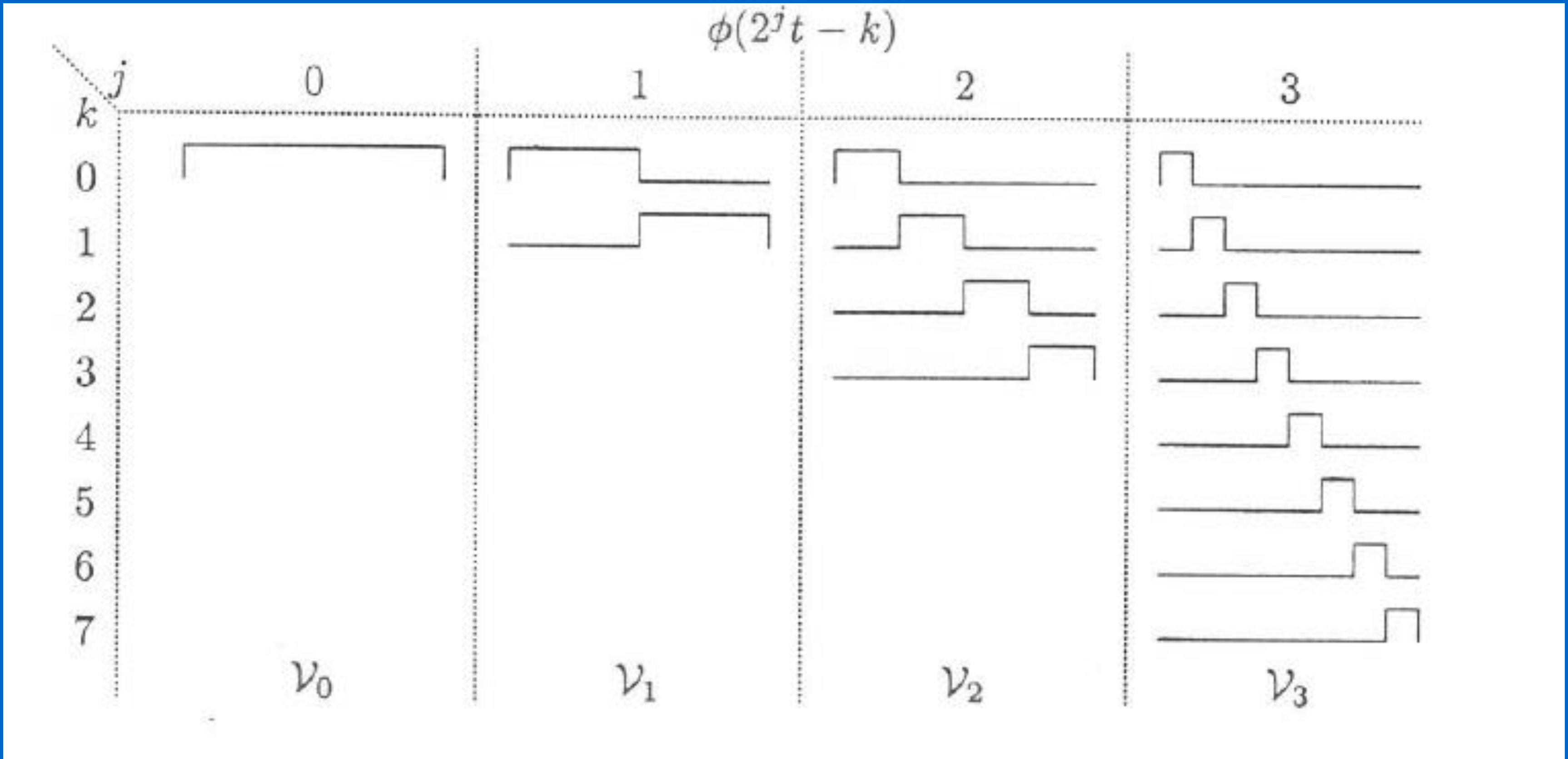
$V_1$



$\in V_j$

width: 1/2

# Define a basis for $V_j$ (cont'd)

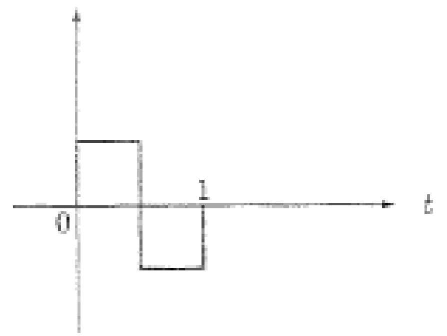



width: $1/2^0$ width: $1/2$ width: $1/2^2$ width: $1/2^3$

# Define a basis for $W_j$

- Wavelet function:

$$\psi(x) = \begin{cases} 1 & \text{if } 0 \le x < 1/2 \\ -1 & \text{if } 1/2 \le x < 1 \\ 0 & otherwise \end{cases}$$



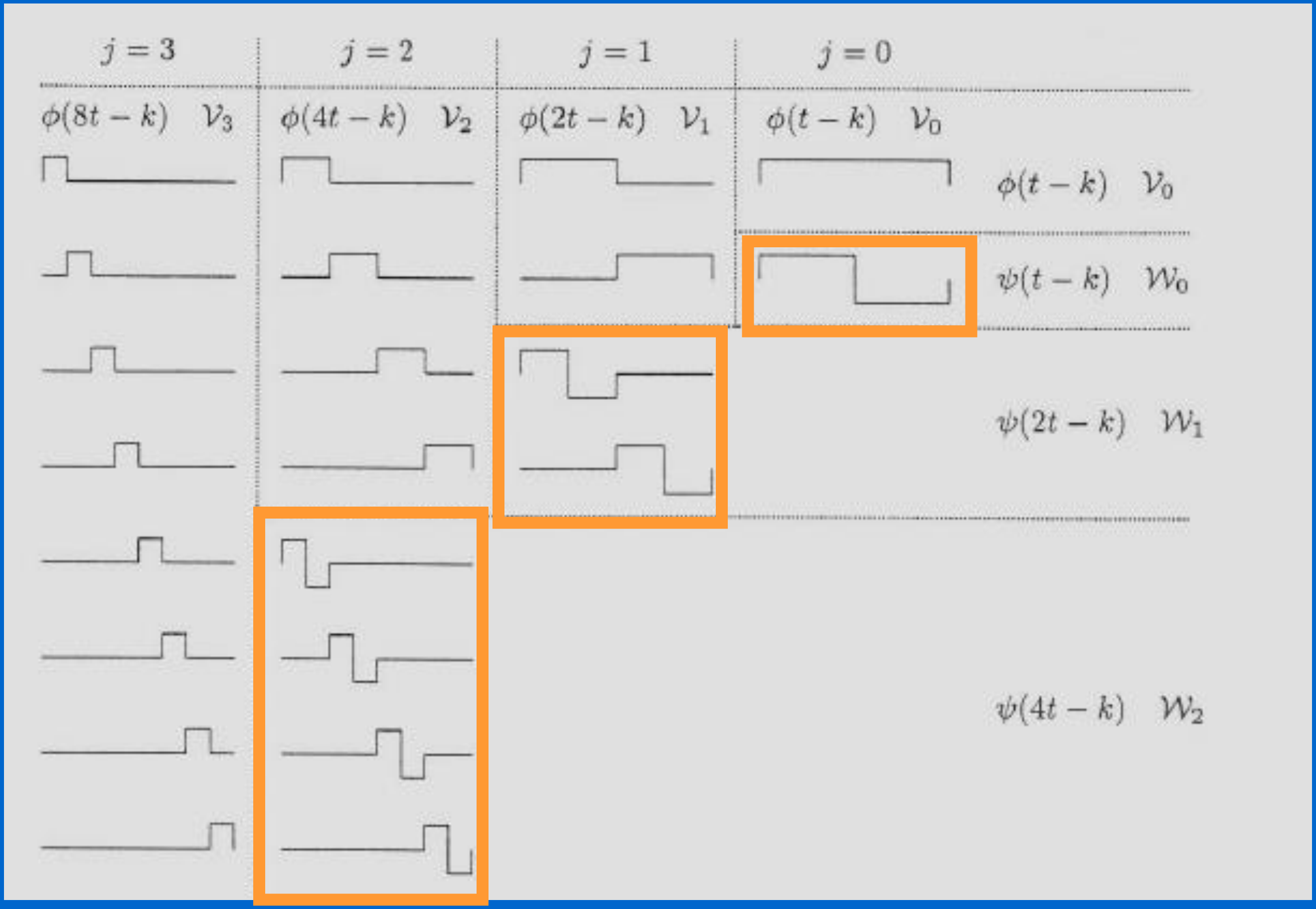
(b) $\psi(t)$

- Let's define a basis $\psi^j_i$ for $W_j$ :

$$\psi_i^j(x) := \psi(2^j x - i), \quad i = 0, 1, \ldots, 2^j - 1$$

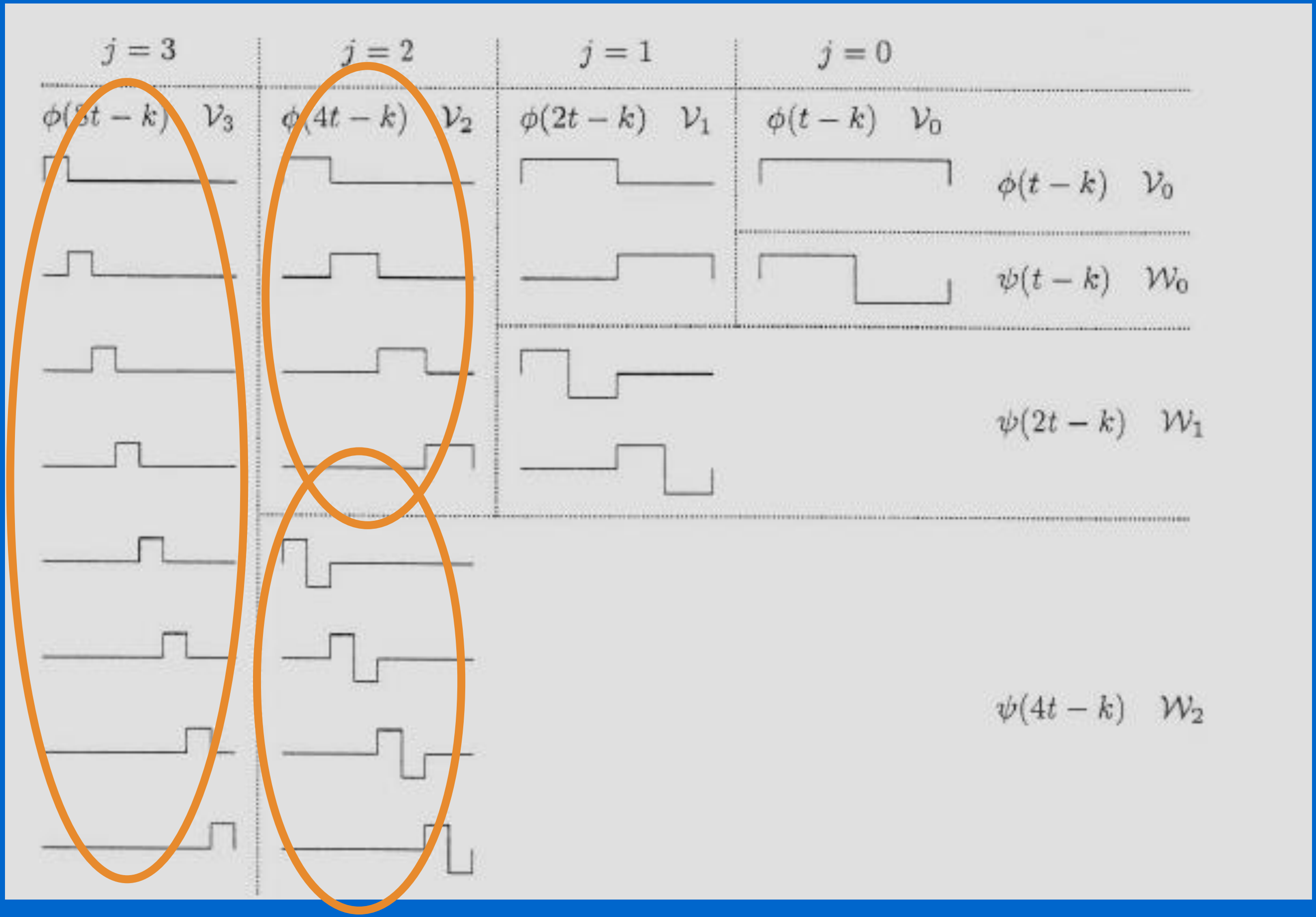
Note new notation: $\psi_i^j(x) \equiv \psi_{ji}(x)$

# Define basis for $W_j$ (cont'd)



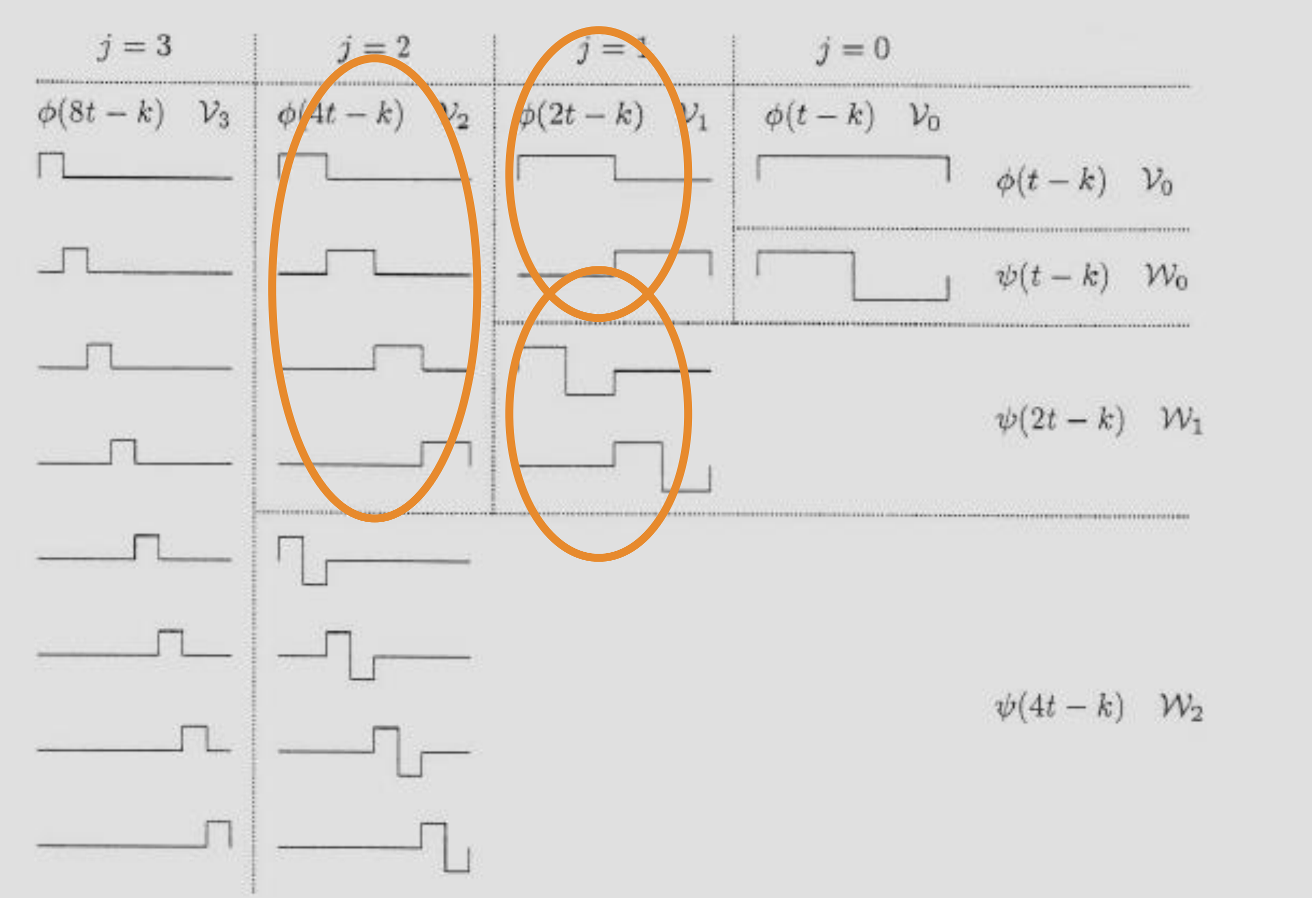

Note that the dot product between basis functions in $V_j$ and $W_j$ is zero!

# Define a basis for $W_j$ (cont'd)



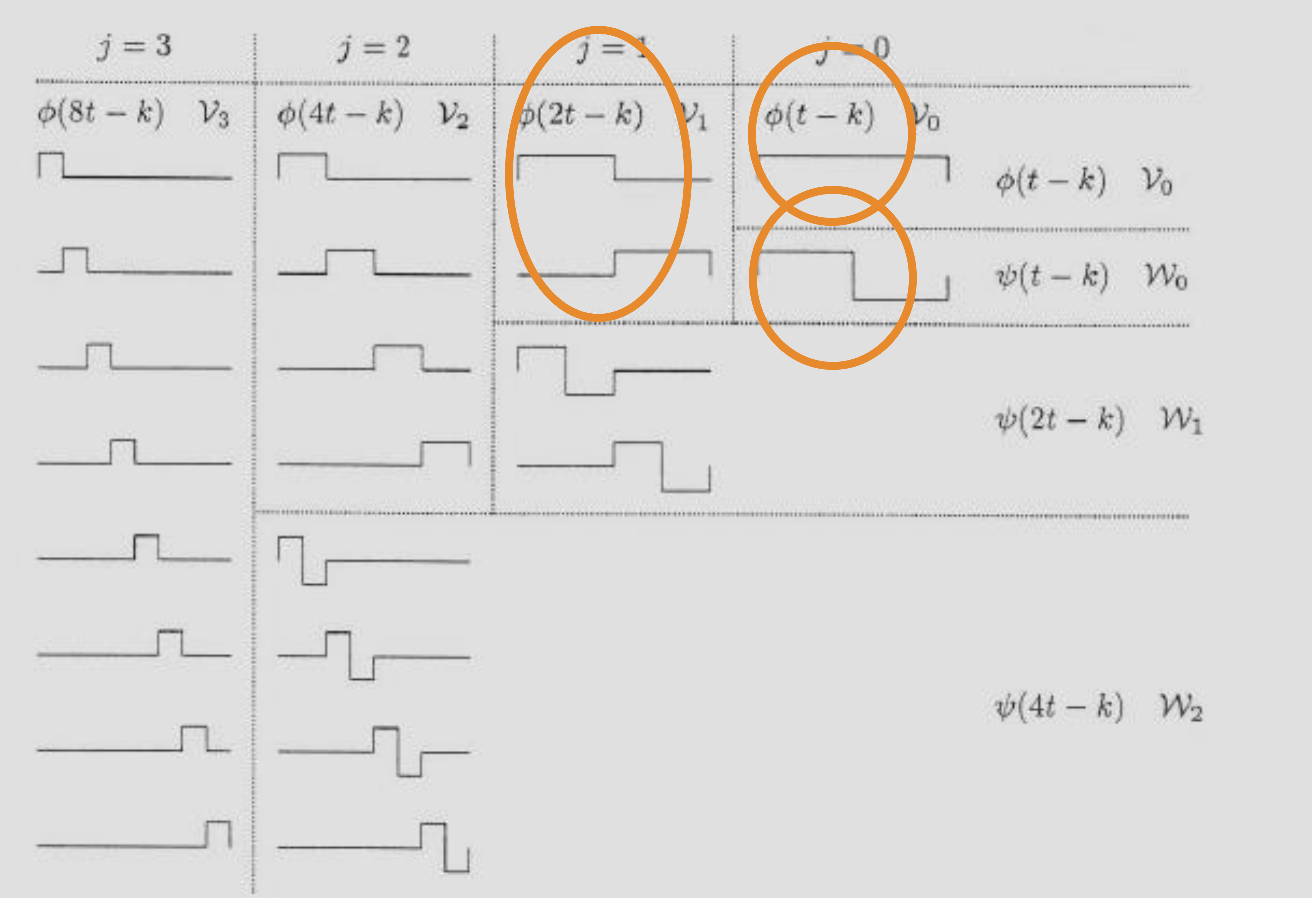

$V_3 = V_2 + W_2$

# Define a basis for $W_j$ (cont'd)




$V_2 = V_1 + W_1$

# Define a basis for $W_j$ (cont'd)



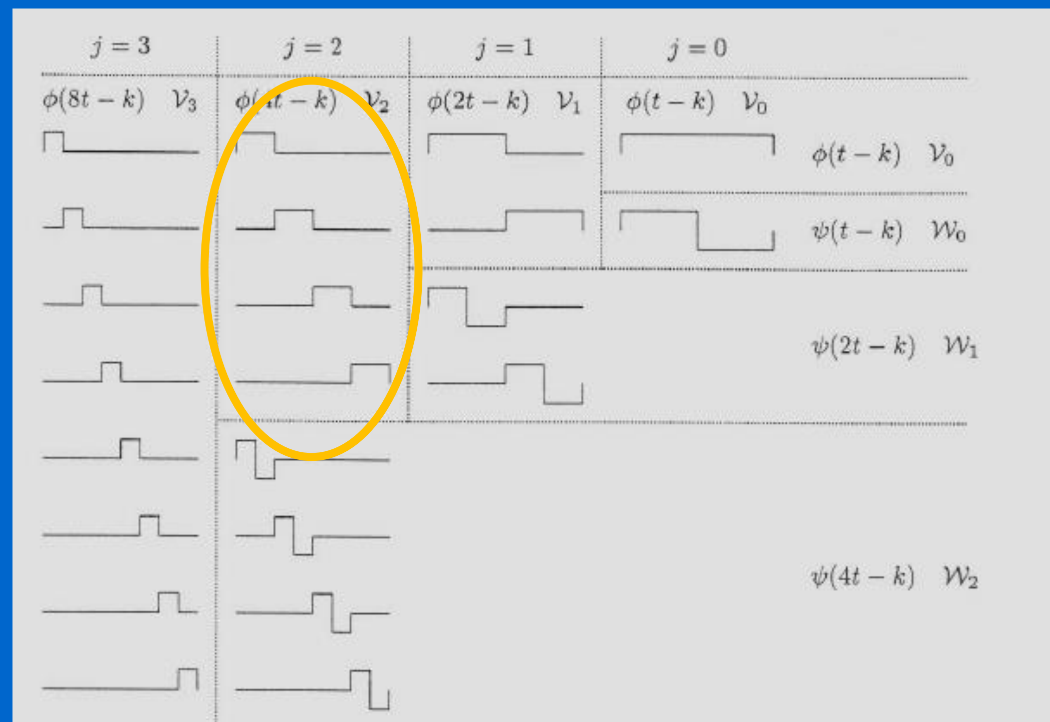

$V_1 = V_0 + W_0$

# Example - Revisited

| *Resolution* | *Averages* | *Detail Coefficients* |
|---|---|---|
| 4 | [9 7 3 5] | [] |
| 2 | [8 4] | [1 −1] |
| 4 | [6] | [2] |

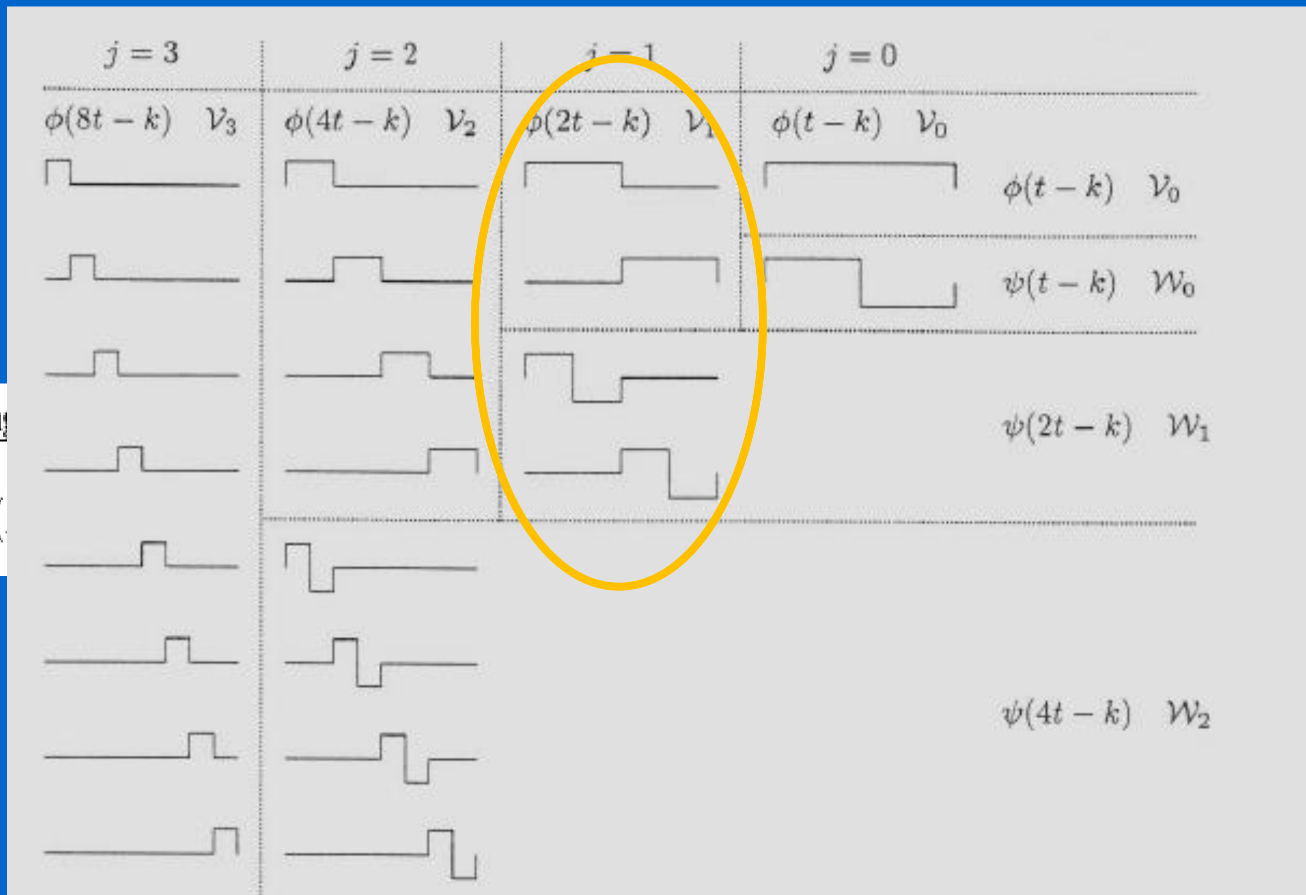
f(x)= [9 7 3 5]

$V_2$



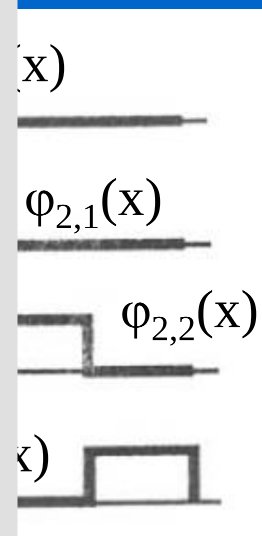
# Example (cont’d)

f(x)=




using

$f(x) = c_0^2\phi_0^2($



# Example (cont'd)

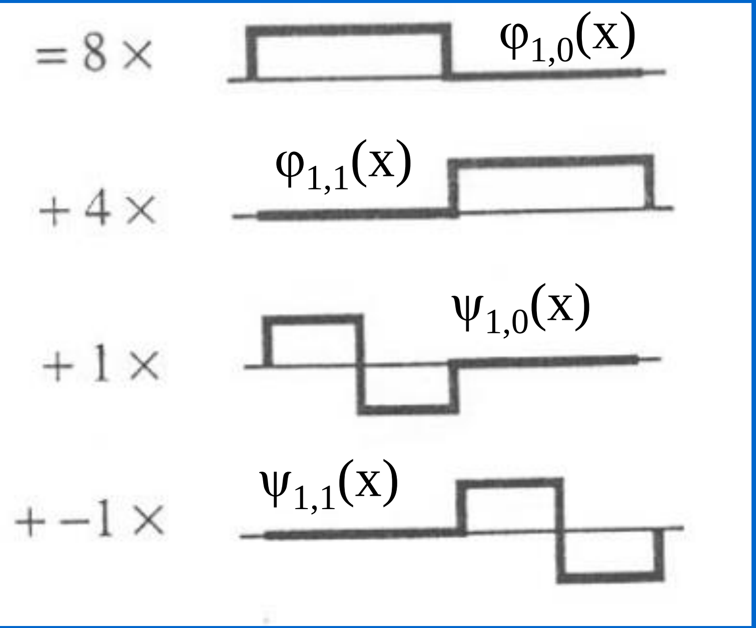
(divide by 2 for normalization)

using the basis functions in $V_1$ and $W_1$

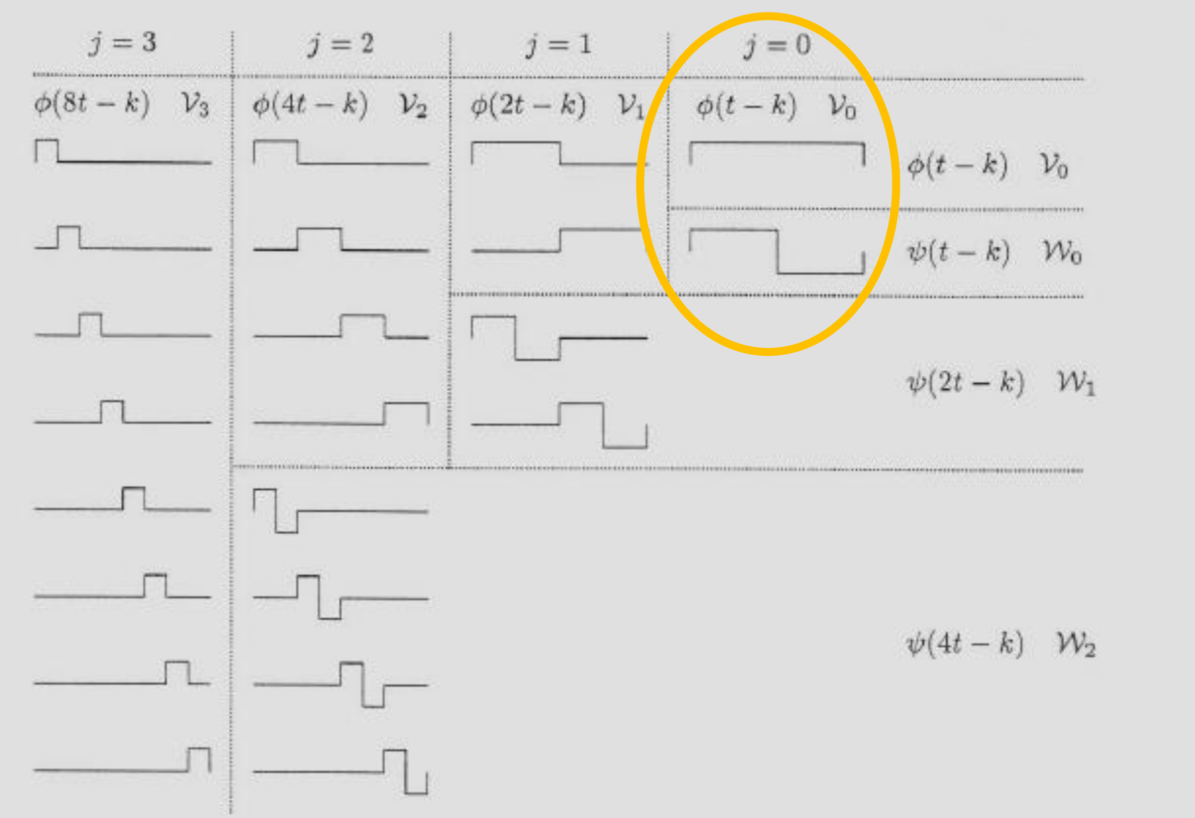
$$V_2 = V_1 + W_1$$

$$f(x) = c_0^1\phi_0^1(x) + c_1^1\phi_1^1(x) + d_0^1\psi_0^1(x) + d_1^1\psi_1^1(x)$$

| *Resolution* | *Averages* | *Detail Coefficients* |
|---|---|---|
| 4 | [9 7 3 5] | [] |
| 2 | [8 4] | [1 −1] |
| 4 | [6] | [2] |



# Example (cont'd)



# Example (cont'd)

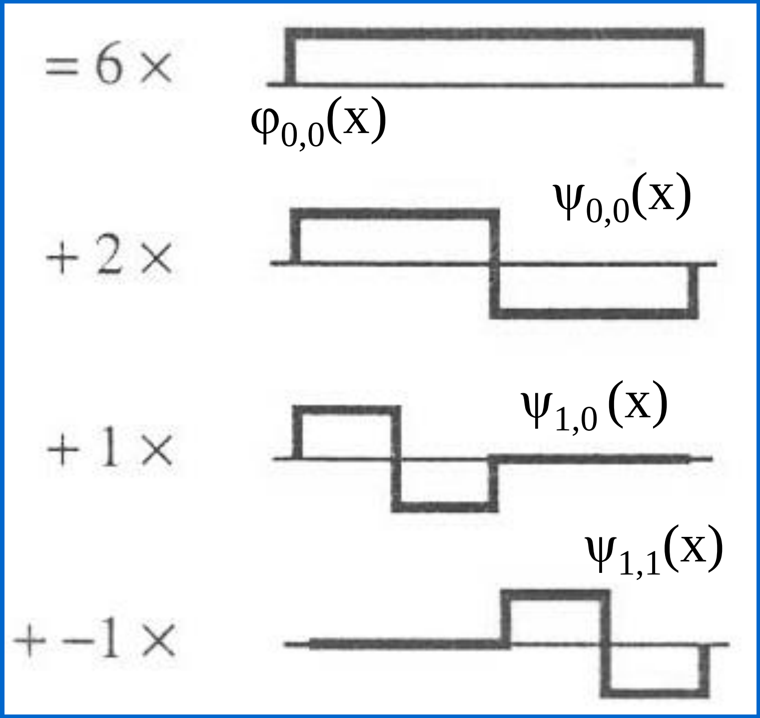
using the basis functions in $V_0$, $W_0$ and $W_1$

$$V_2 = V_1 + W_1 = V_0 + W_0 + W_1$$

$$f(x) = c_0^0 \phi_0^0(x) + d_0^0 \psi_0^0(x) + d_0^1 \psi_0^1(x) + d_1^1 \psi_1^1(x)$$

| *Resolution* | *Averages* | *Detail Coefficients* |
|---|---|---|
| 4 | [9 7 3 5] | [] |
| 2 | [8 4] | [1 – 1] |
| 4 | [6] | [2] |

(divide by 2 for normalization)

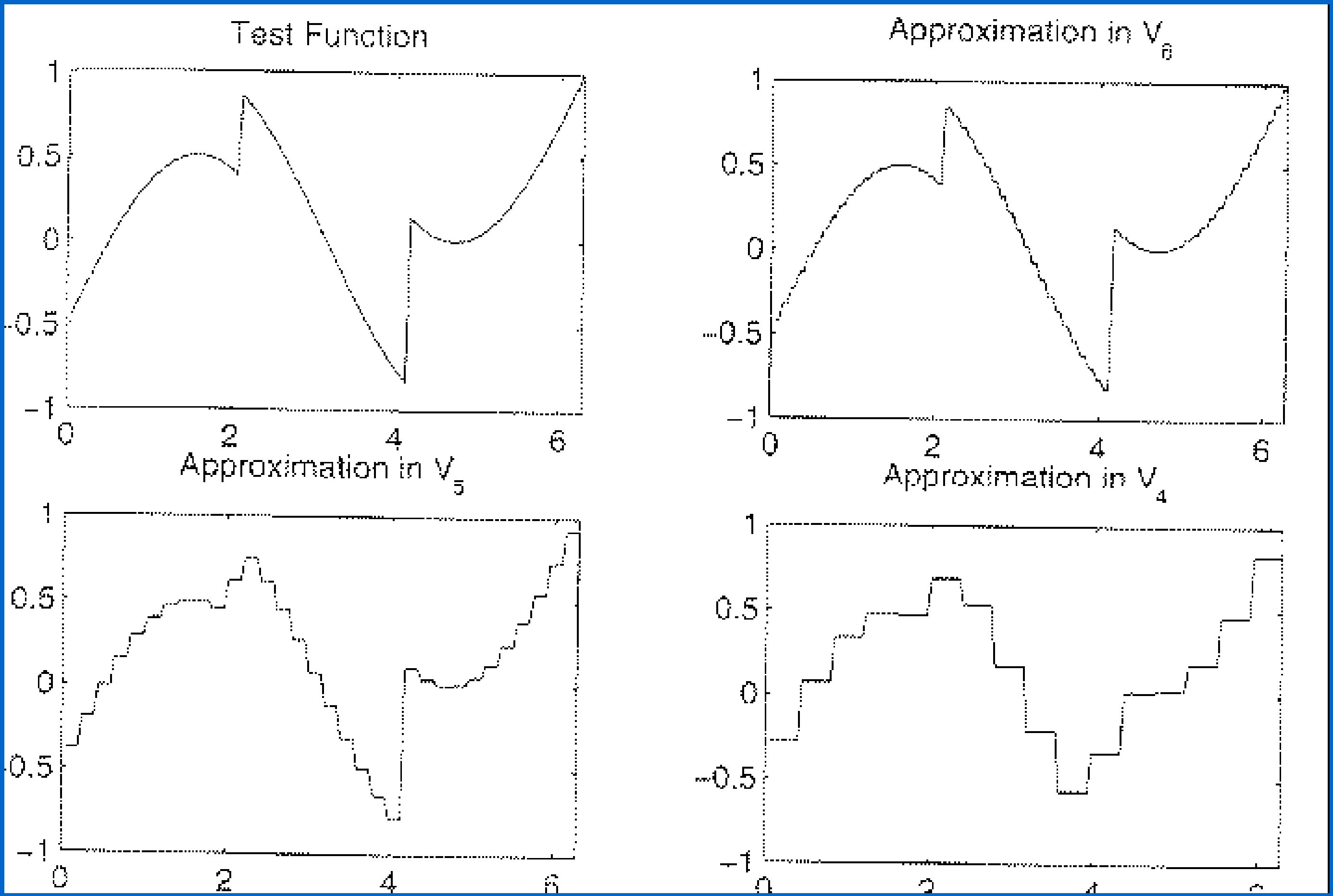



$$f(t) = \sum_k c_k \varphi(t-k) + \sum_k \sum_j d_{jk} \psi(2^j t - k)$$

scaling function

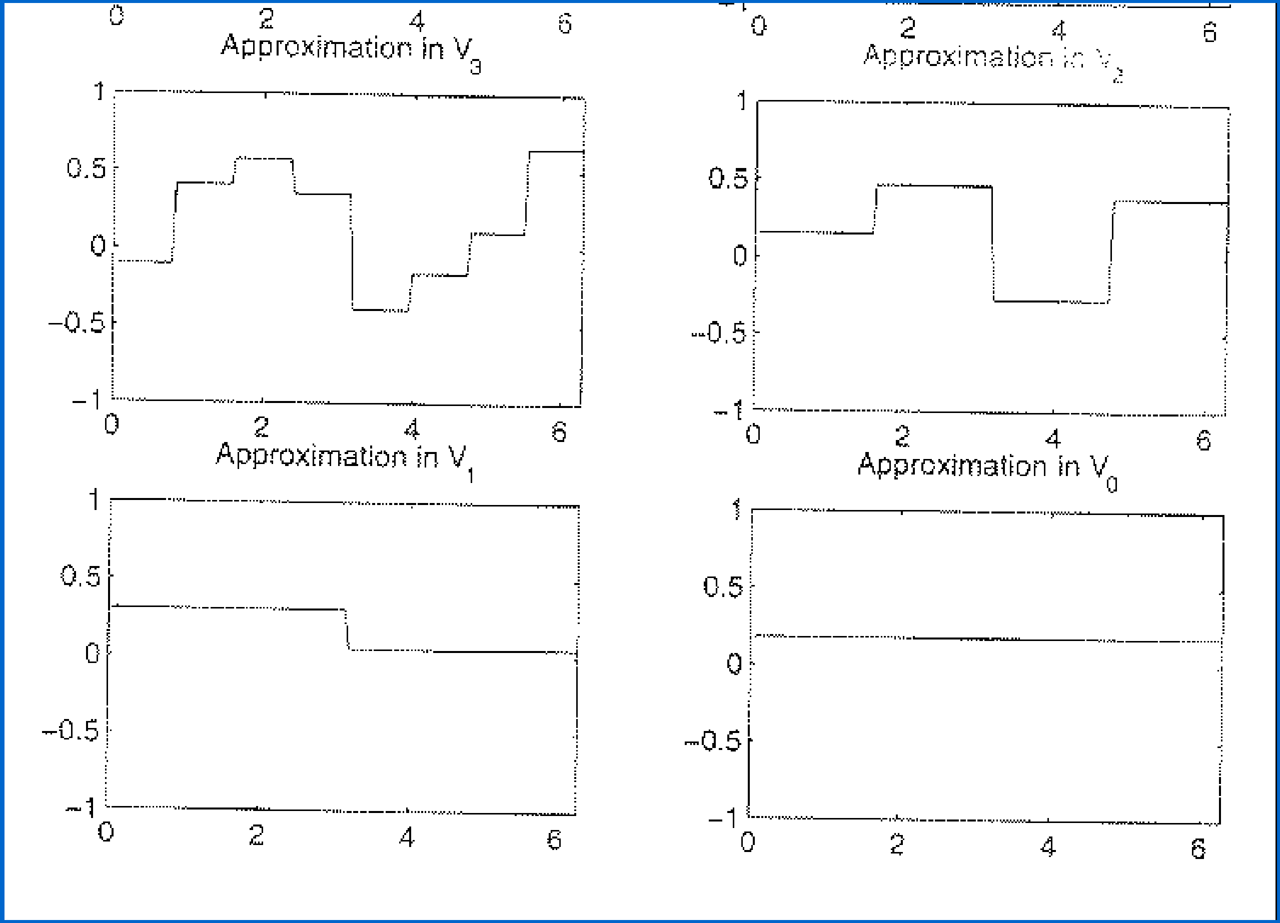
wavelet function

# Example



Test Function
Approximation in $V_6$
Approximation in $V_5$
Approximation in $V_4$

# Example (cont'd)



# Example (revisited)

[9 7 3 5]

low-pass,
down-sampling

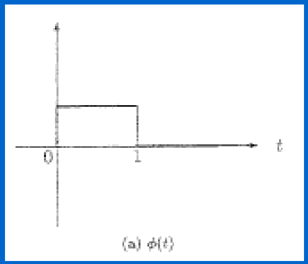
high-pass,
down-sampling

[8  4]

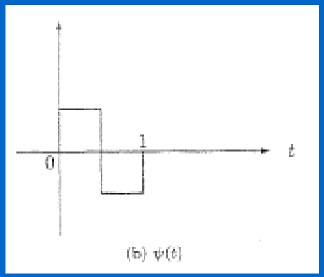
[1  −1]

(9+7)/2  (3+5)/2  (9-7)/2  (3-5)/2

LP




HP



# Example (revisited)

[9 7 3 5]

[8 4]

[1 −1]

low-pass,
down-sampling

high-pass,
down-sampling

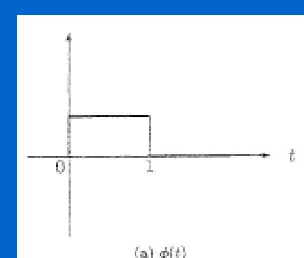
[6]

[2]

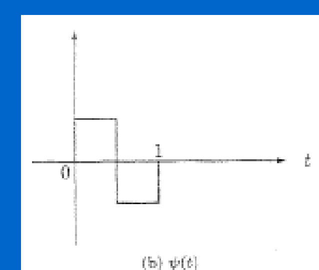
(8+4)/2

(8-4)/2

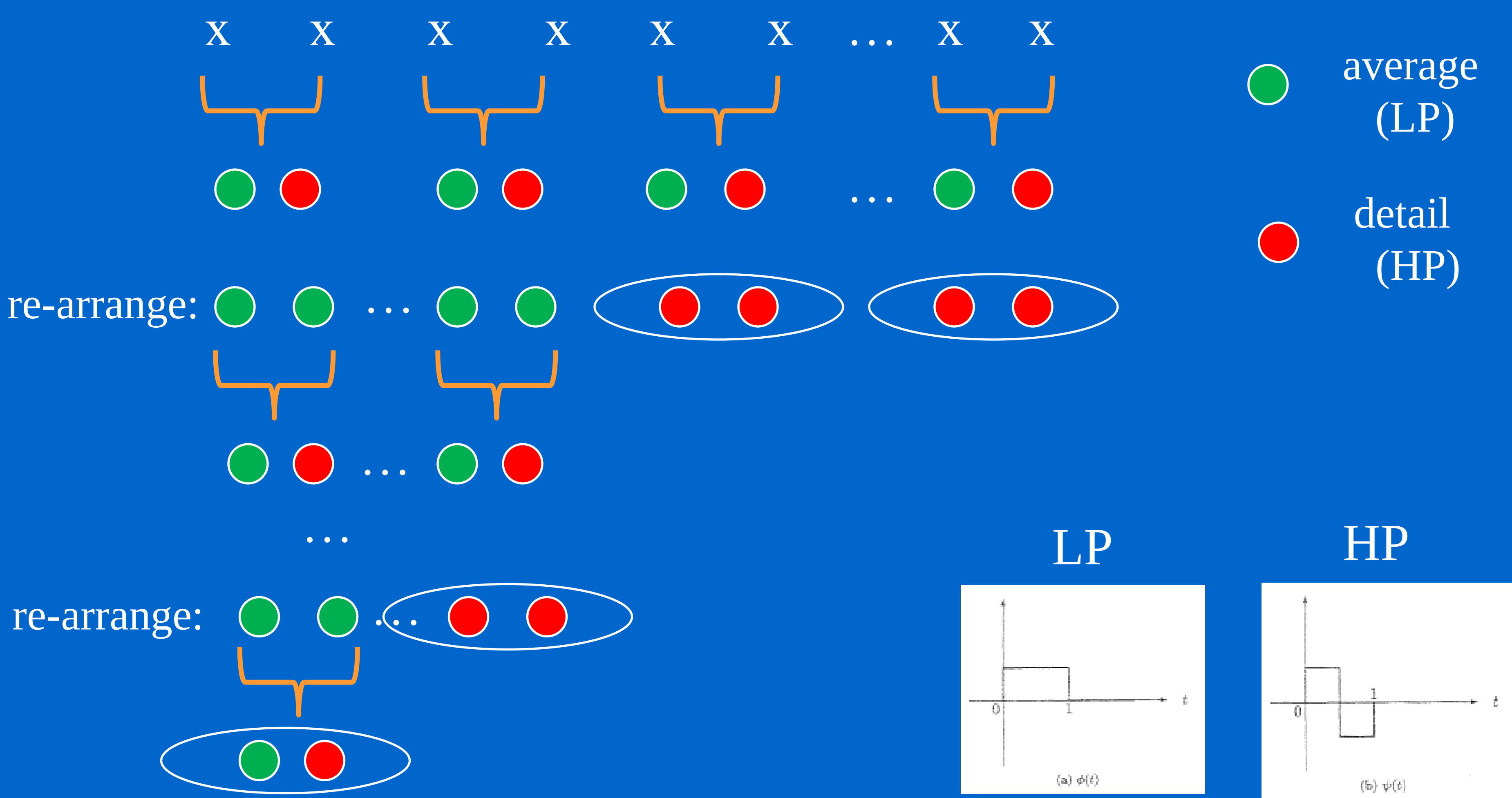
LP




HP



# Convention for illustrating 1D Haar wavelet decomposition



# Standard Haar wavelet decomposition

- Steps:

(1) Compute 1D Haar wavelet decomposition of each row of the original pixel values.

(2) Compute 1D Haar wavelet decomposition of each column of the row-transformed pixels.

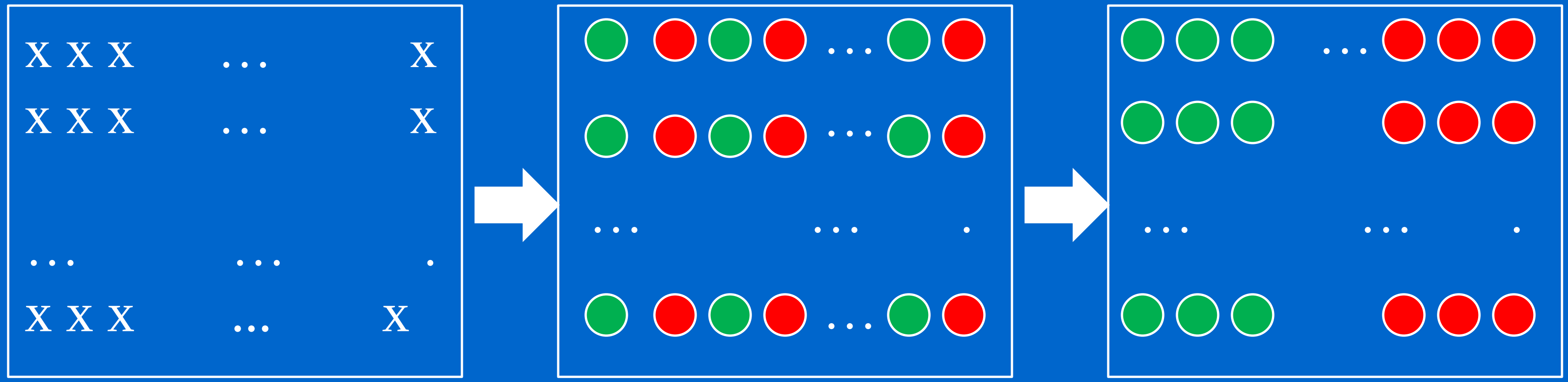
# Standard Haar wavelet decomposition (cont'd)

average

detail

(1) row-wise Haar decomposition:

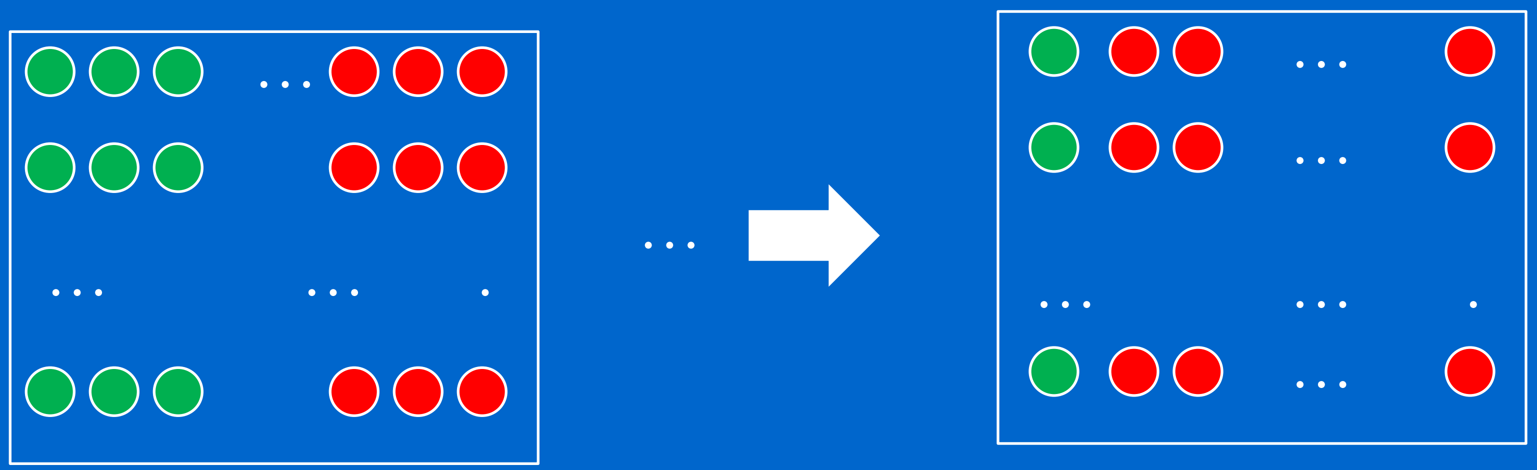
re-arrange terms



# Standard Haar wavelet decomposition (cont'd)


(1) row-wise Haar decomposition:

row-transformed result



# Standard Haar wavelet decomposition (cont'd)

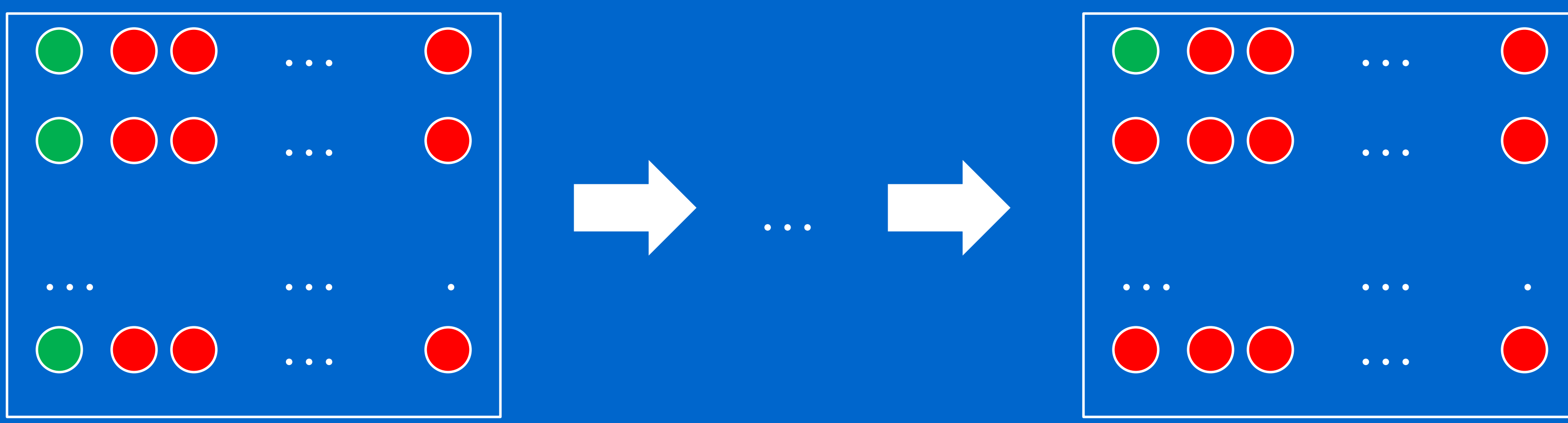
average

detail

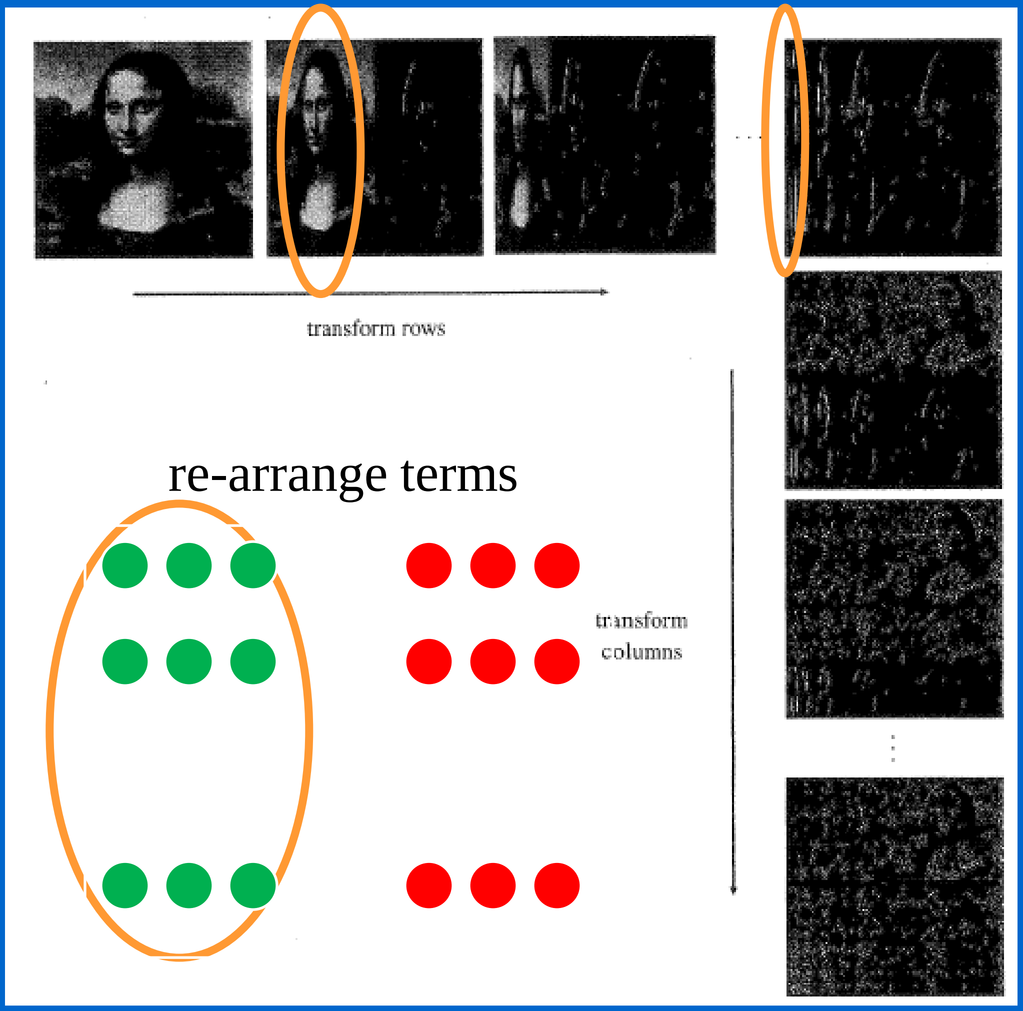
(2) column-wise Haar decomposition:

row-transformed result

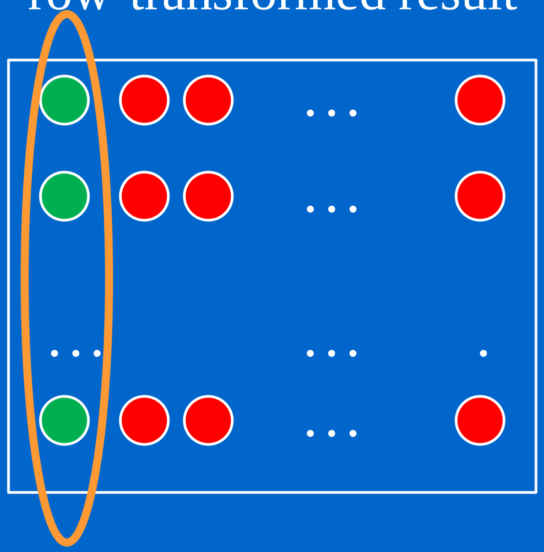
**column-transformed result**



# Example

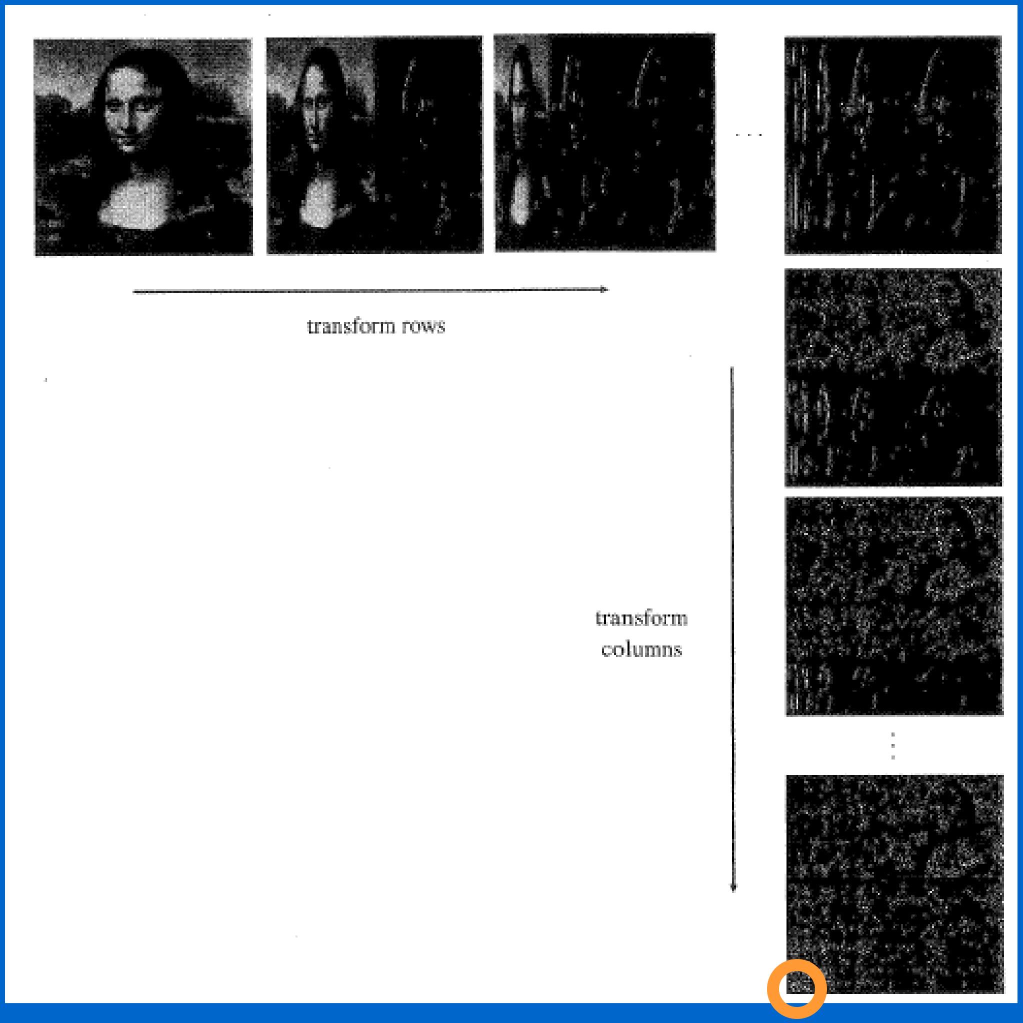


transform rows
re-arrange terms
transform columns

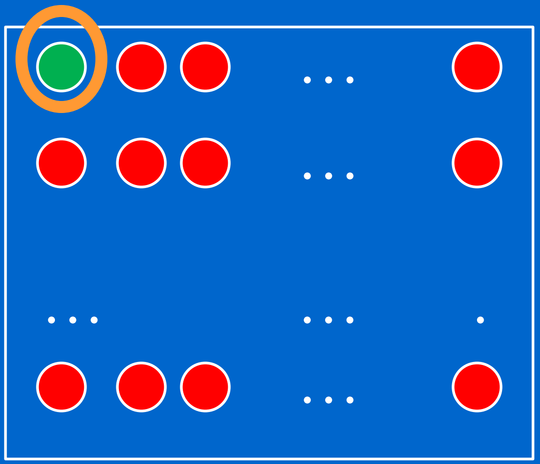



row-transformed result

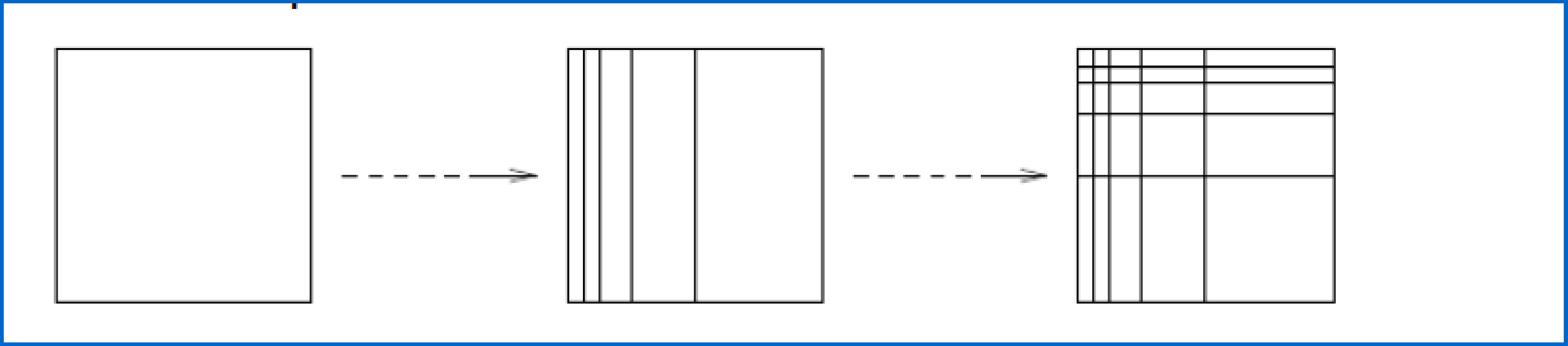
# Example (cont'd)




column-transformed result



# Standard Haar wavelet decomposition (cont'd)

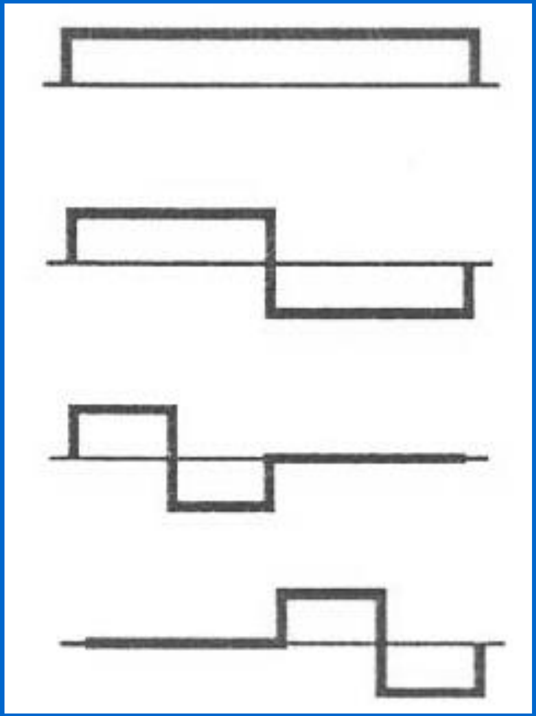


# What is the 2D Haar basis for the standard decomposition?

To construct the standard 2D Haar wavelet basis, consider all possible outer products of the1D basis functions.

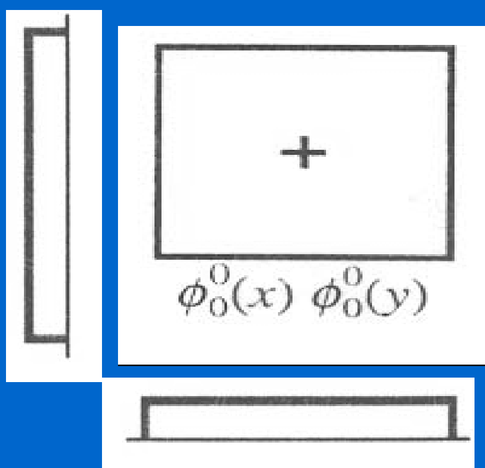
Example:

$V_2 = V_0 + W_0 + W_1$

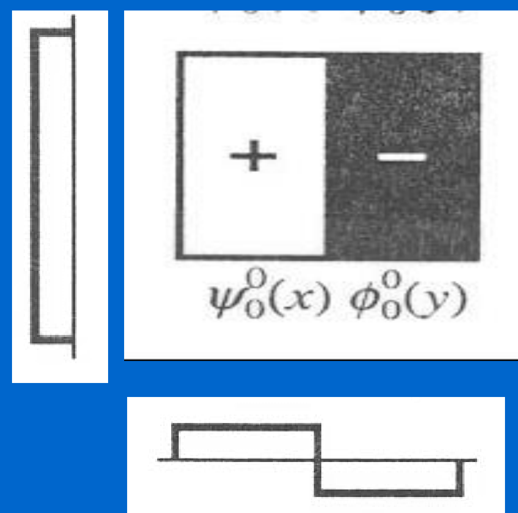


# What is the 2D Haar basis for the standard decomposition?

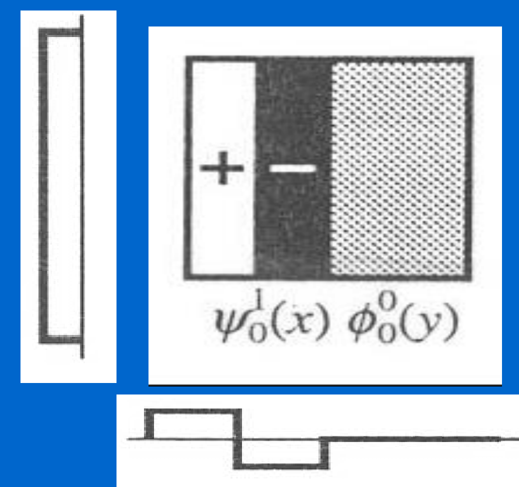
To construct the standard 2D Haar wavelet basis, consider all possible outer products of the1D basis functions.




$\varphi_{00}(x),\ \varphi_{00}(x)$



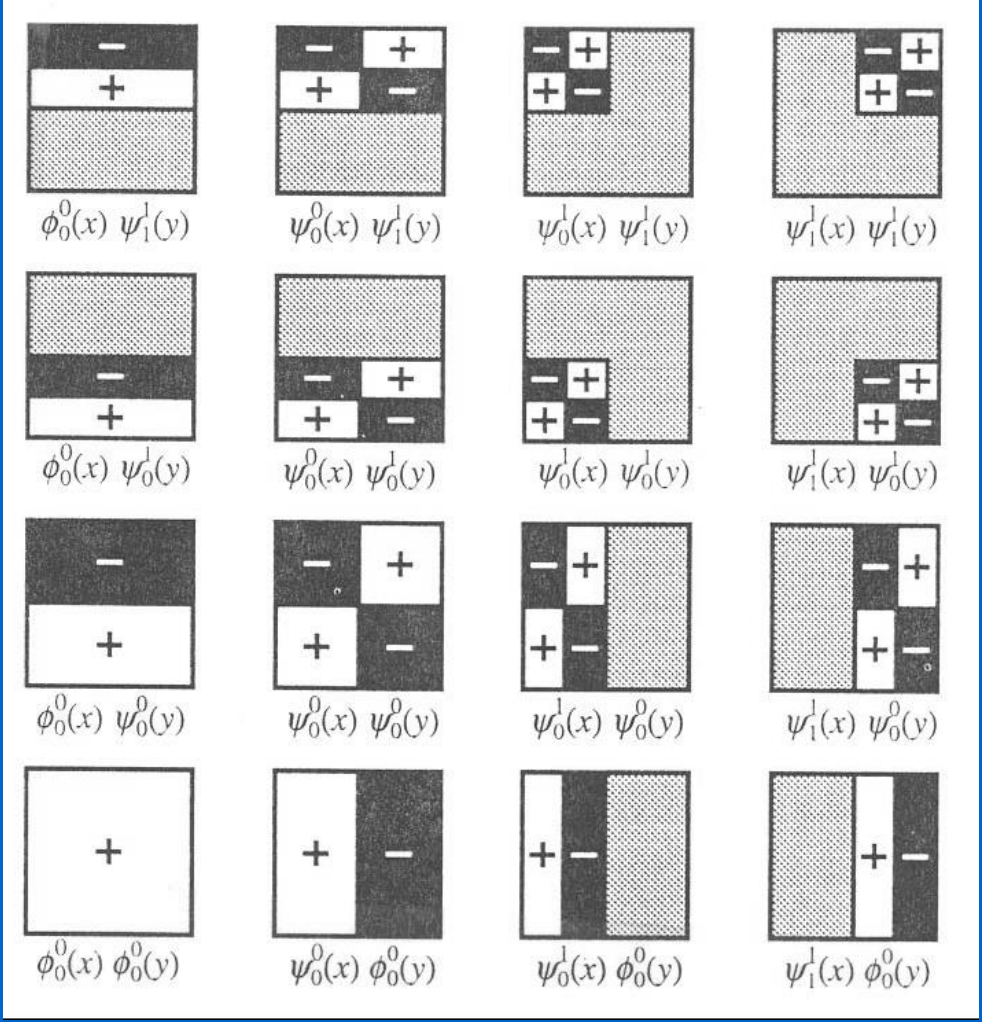

$\psi_{00}(x),\ \varphi_{00}(x)$




$\psi_{10}(x),\ \varphi_{00}(x)$

Notation: $\varphi_i^j(x) \equiv \varphi_{ji}(x)$ $\psi_i^j(x) \equiv \psi_{ji}(x)$

# What is the 2D Haar basis for the standard decomposition?




$V_2$

Notation:

$$\varphi_i^j(x) \equiv \varphi_{ji}(x)$$

$$\psi_i^j(x) \equiv \psi_{ji}(x)$$

# Non-standard Haar wavelet decomposition

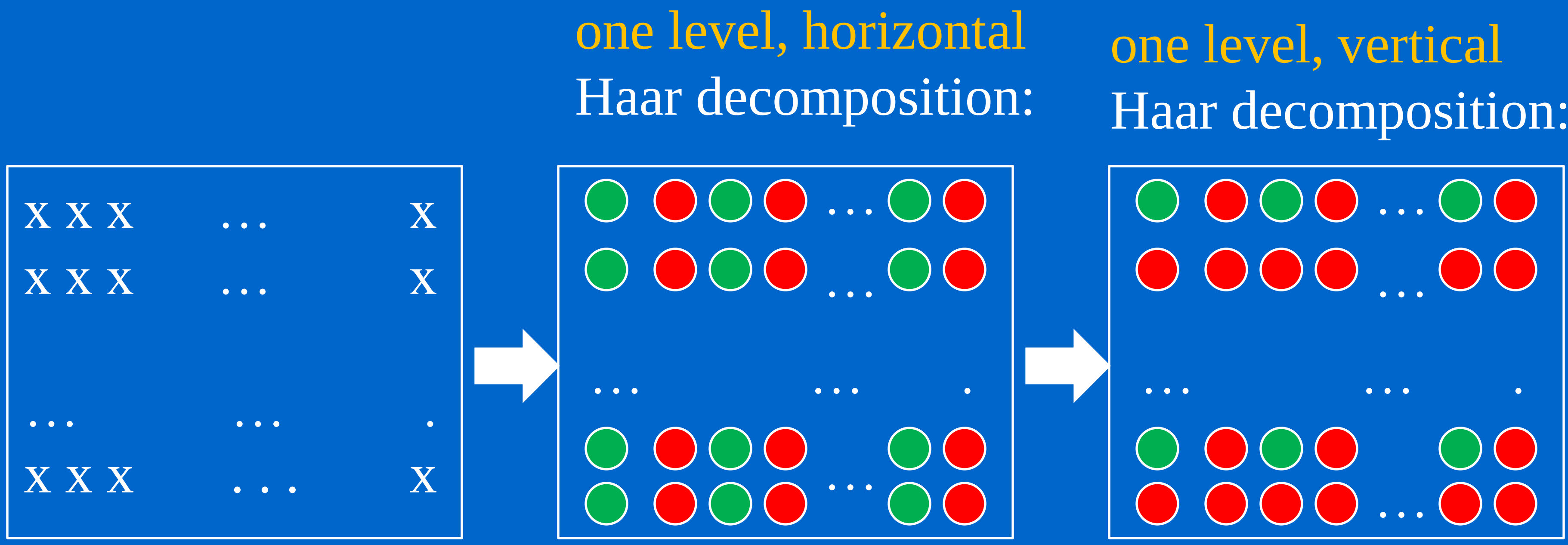
- Alternates between operations on rows and columns.

  (1) Perform one level decomposition in each row (i.e., one step of horizontal pairwise averaging and differencing).

  (2) Perform one level decomposition in each column from step 1 (i.e., one step of vertical pairwise averaging and differencing).

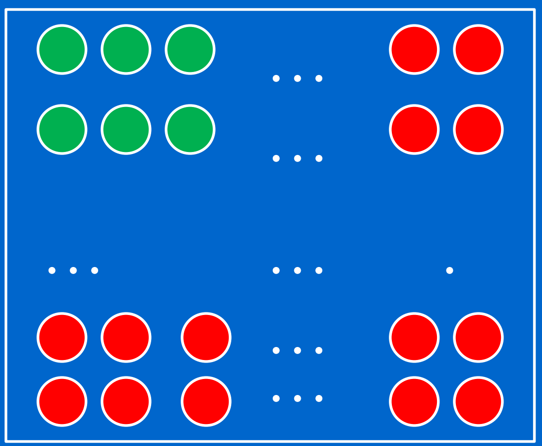
  (3) Rearrange terms and repeat the process on the quadrant containing the averages only.

# Non-standard Haar wavelet decomposition (cont'd)



# Non-standard Haar wavelet decomposition (cont'd)

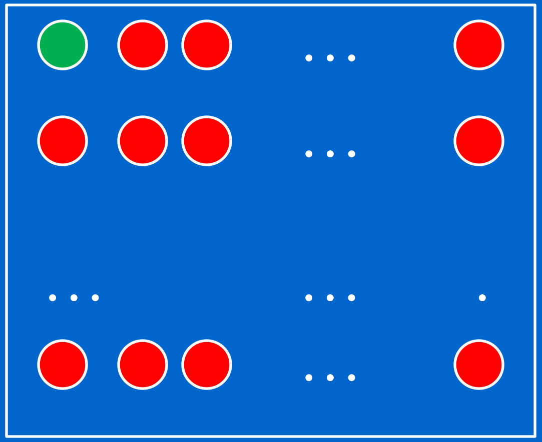
re-arrange terms



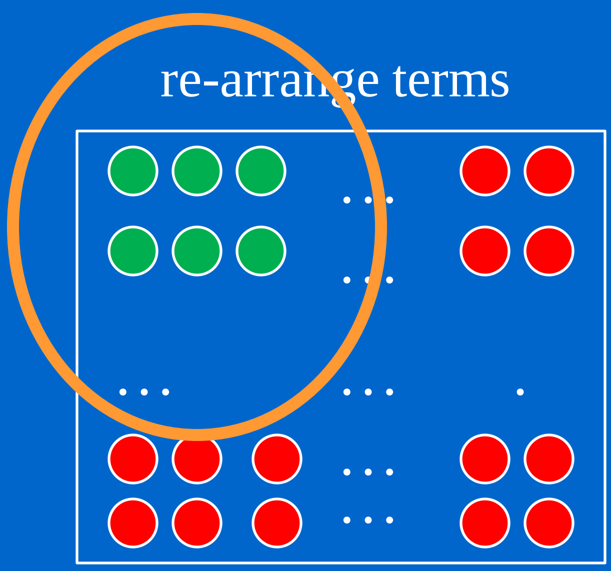
one level, horizontal Haar decomposition on "green" quadrant

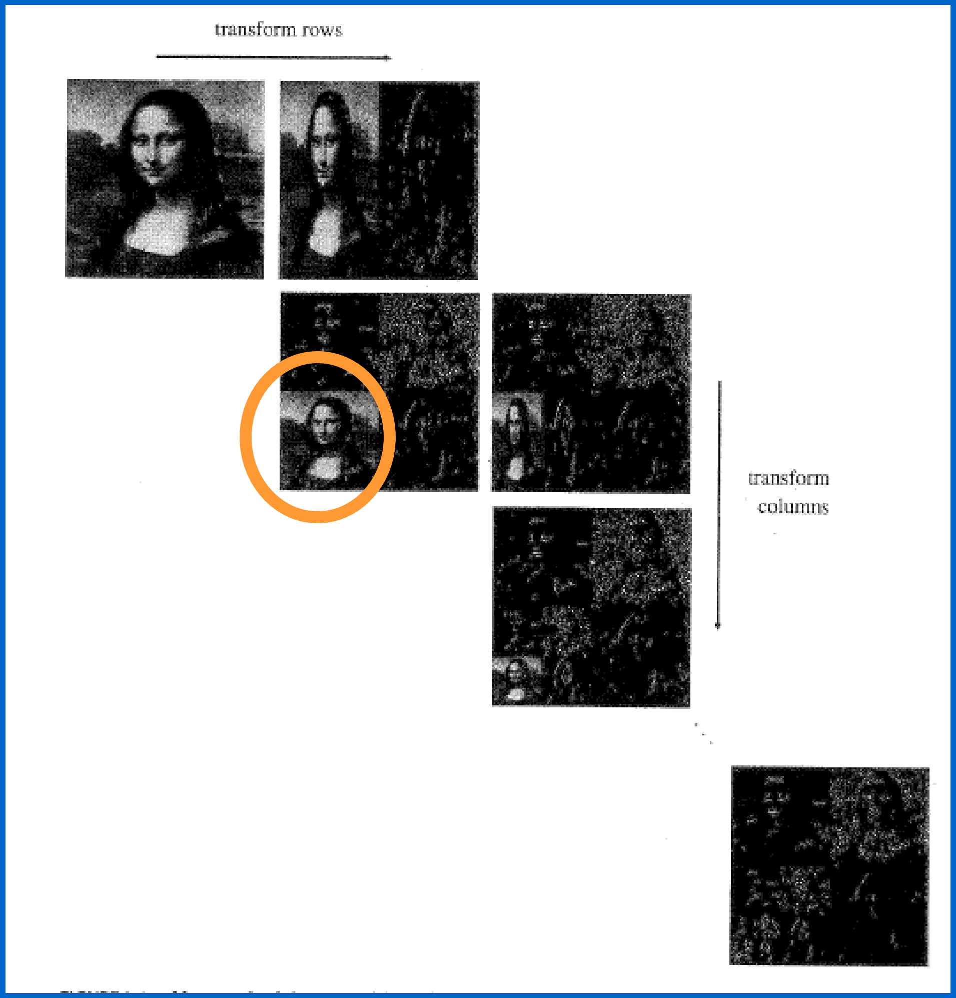
one level, vertical Haar decomposition on "green" quadrant

…

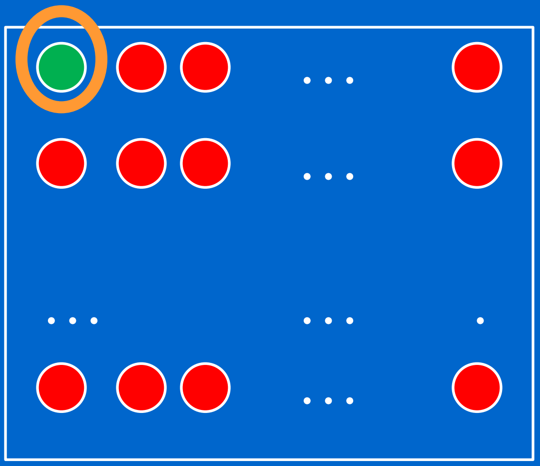


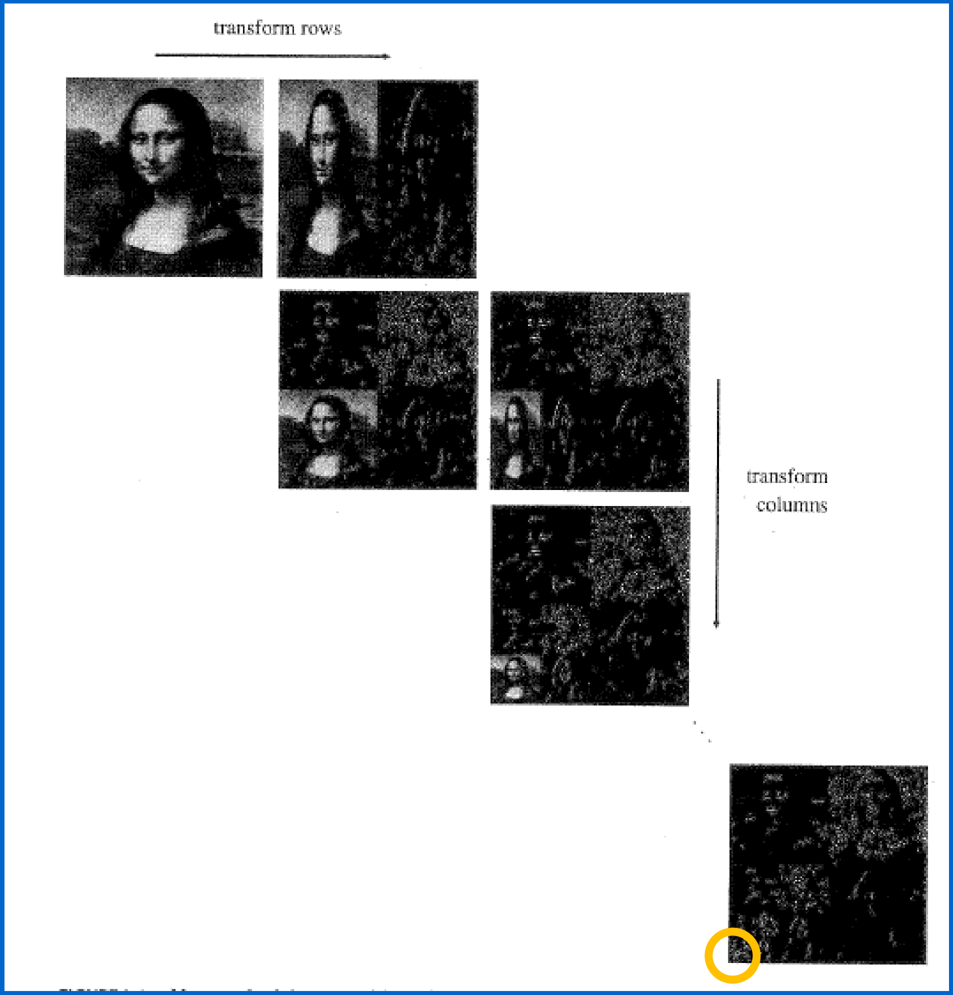
# Example



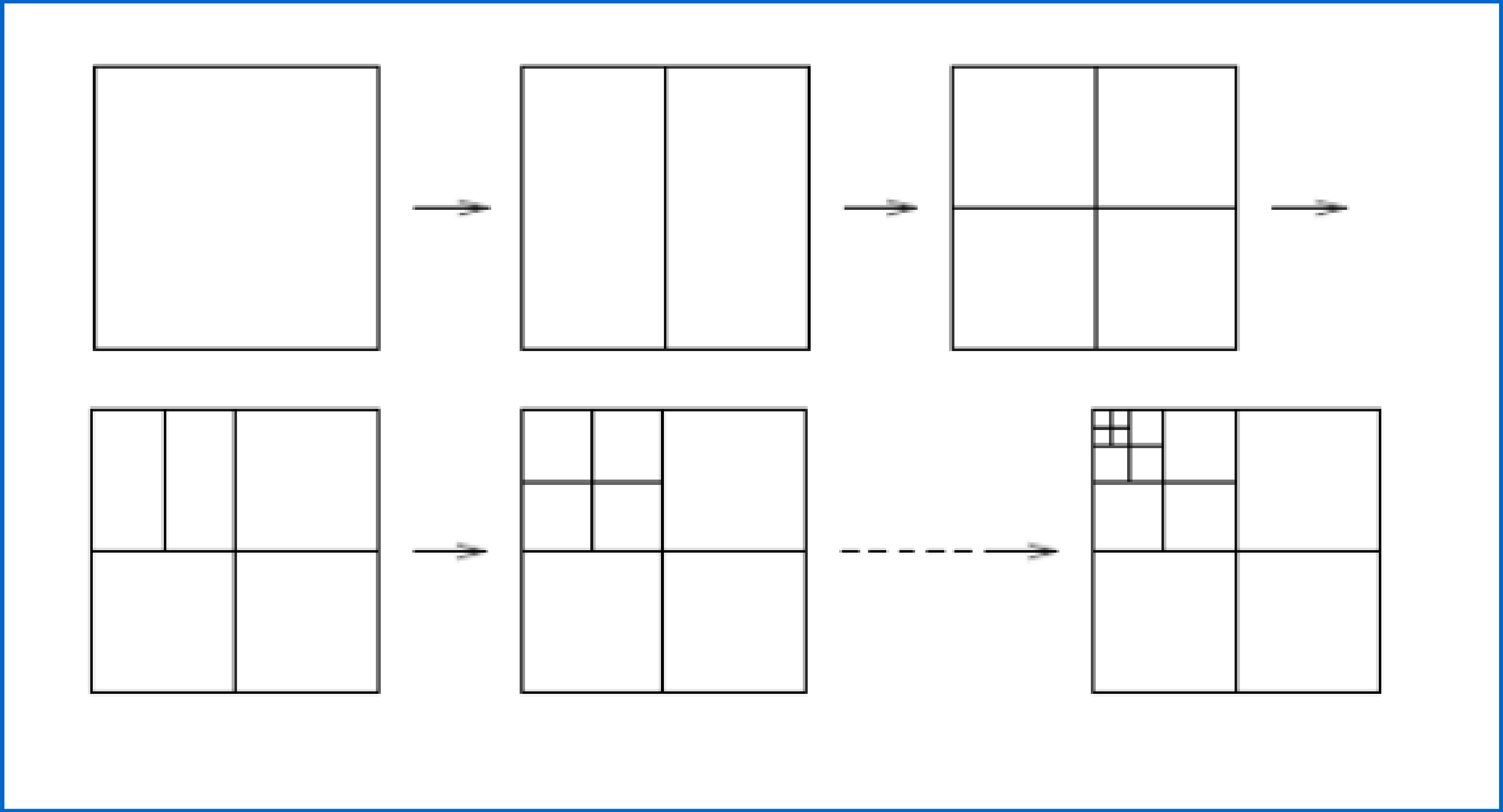



# Example (cont'd)



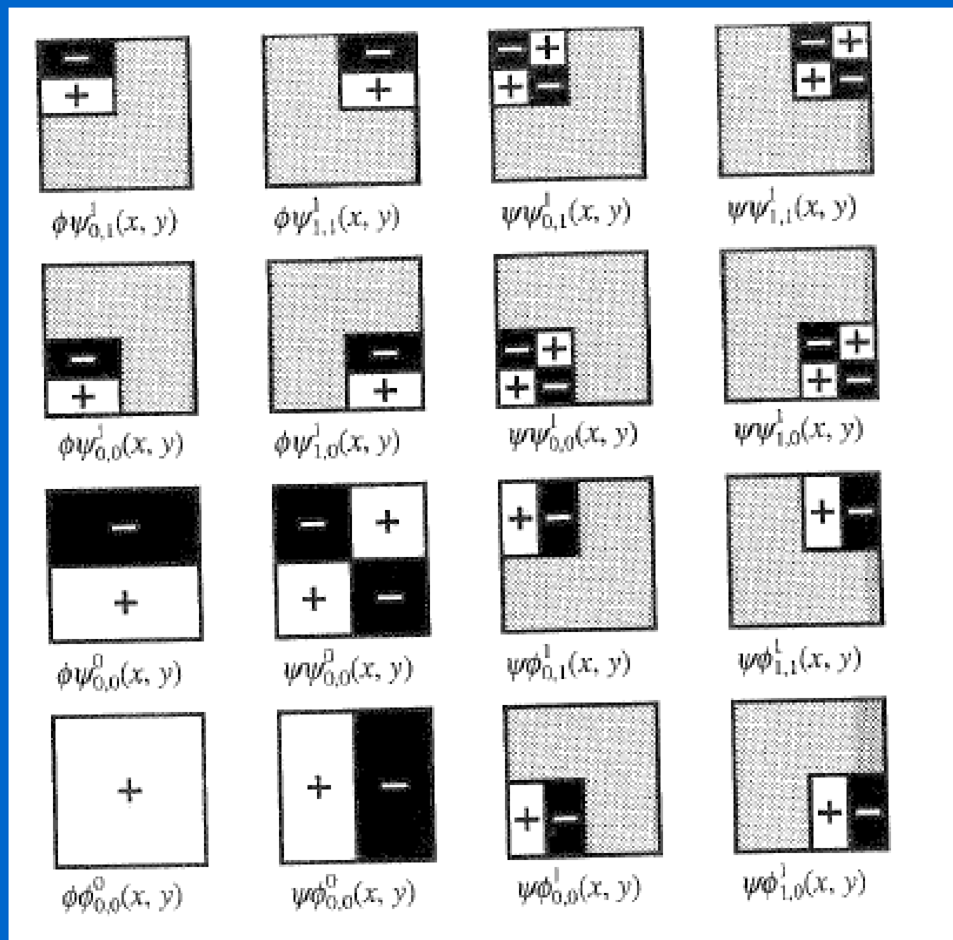


# Non-standard Haar wavelet decomposition (cont'd)



# What is the 2D Haar basis for the non-standard decomposition?

$V_2$




Notation:

$$\varphi_i^j(x) \equiv \varphi_{ji}(x)$$

$$\psi_i^j(x) \equiv \psi_{ji}(x)$$