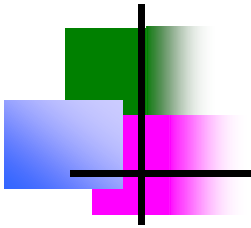


Machine Learning and Computational Intelligence
Lecture 5

Sanjeeb Prasad Panday, PhD
Associate Professor
Dept. of Electronics and Computer Engineering
Director (ICTC)
IOE, TU

Motivation

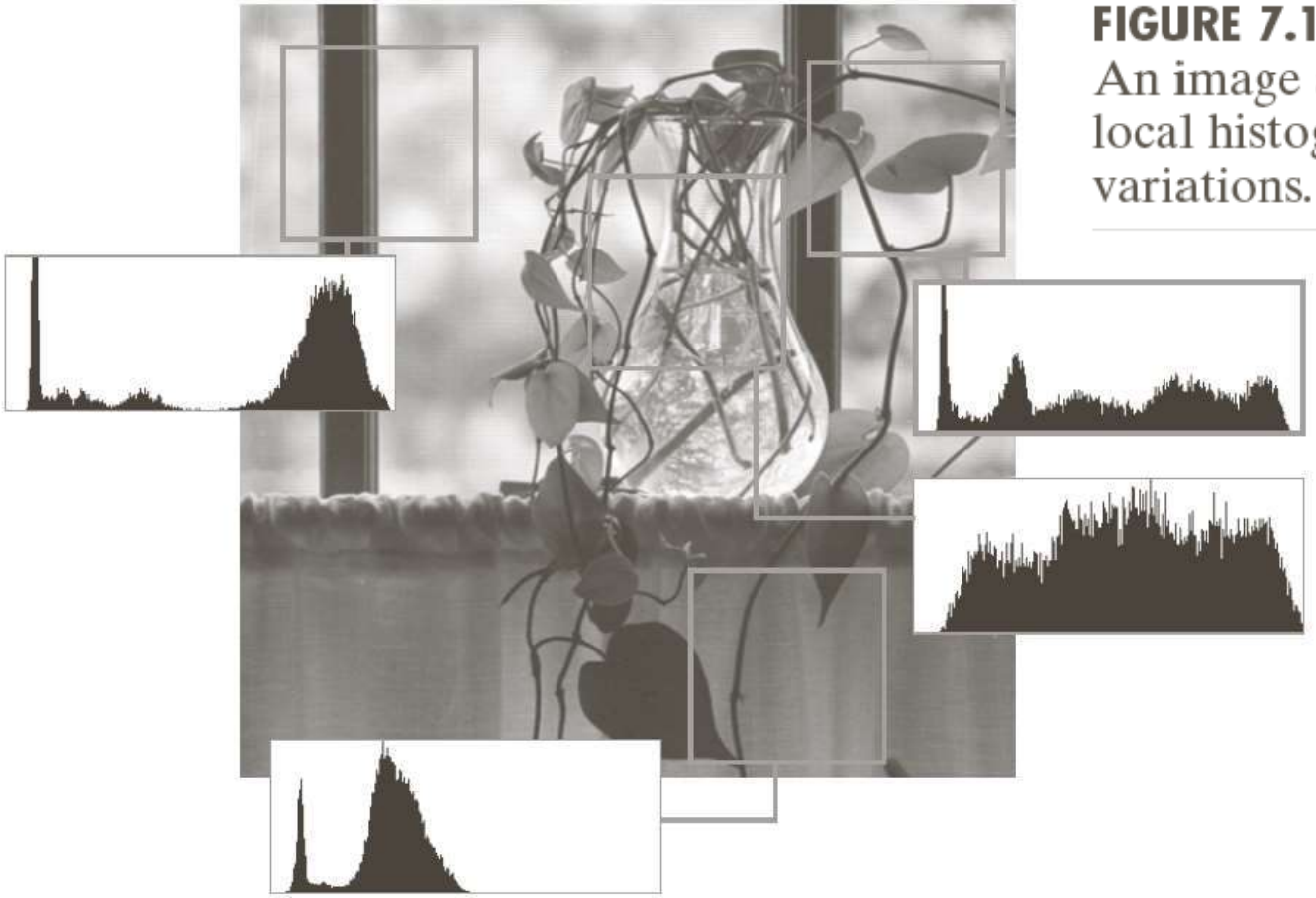


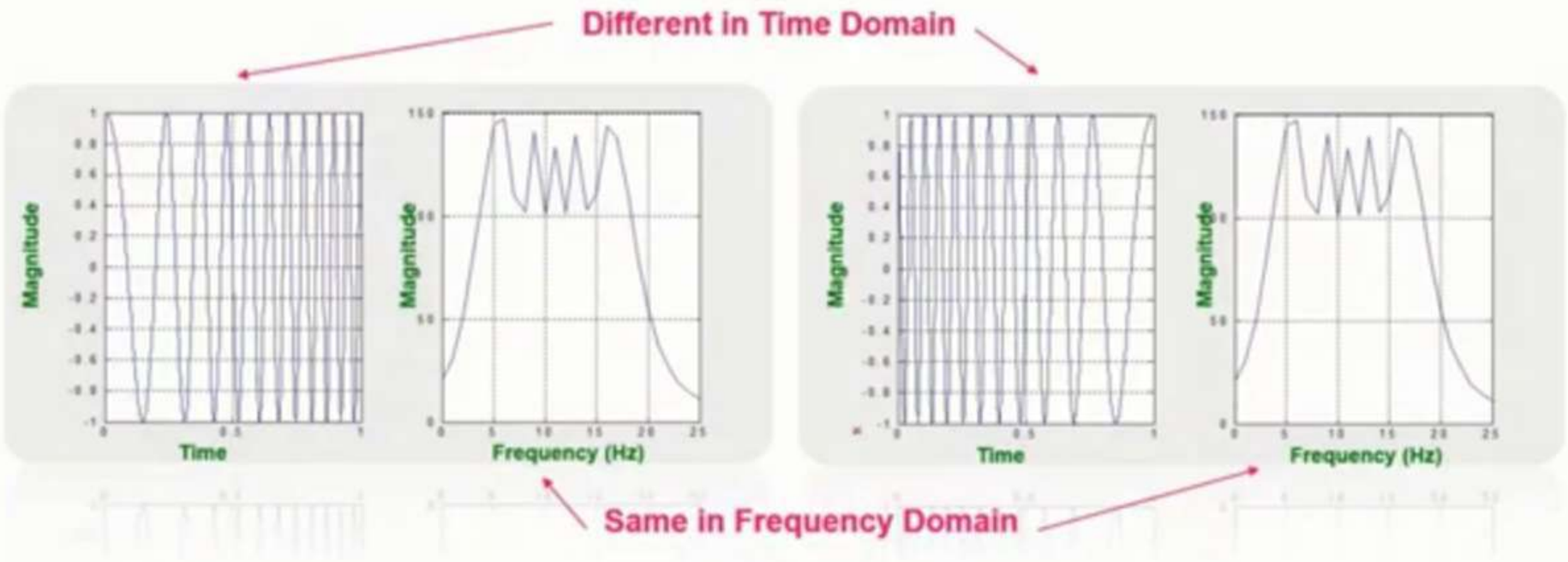
FIGURE 7.1
An image and its local histogram variations.

Problem with Fourier…

Fourier analysis -- breaks down a signal into constituent sinusoids of different frequencies.

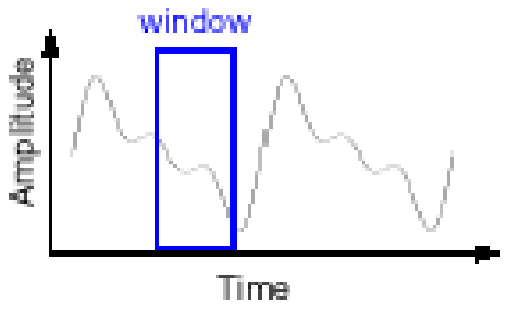
a serious drawback In transforming to the frequency domain, time information is lost. When looking at a Fourier transform of a signal, it is impossible to tell *when* a particular event took place.

The Fourier Transform and its Limitations

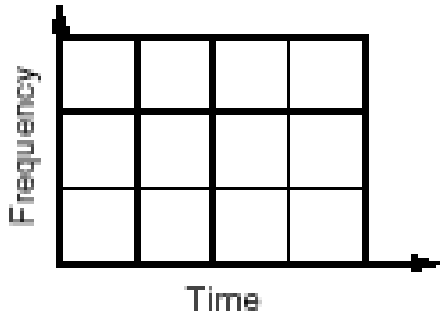


At what time the frequency components occur? FT can not tell!

Gabor's proposal



Short
Time
Fourier
Transform



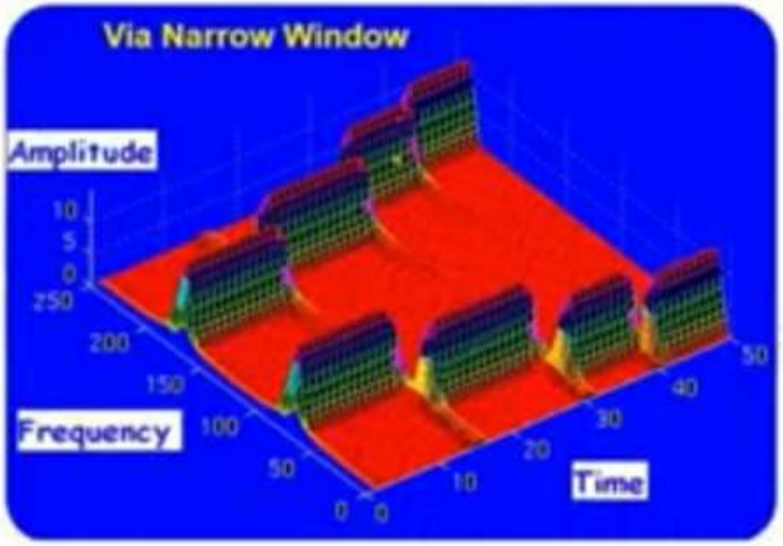
Drawbacks of STFT

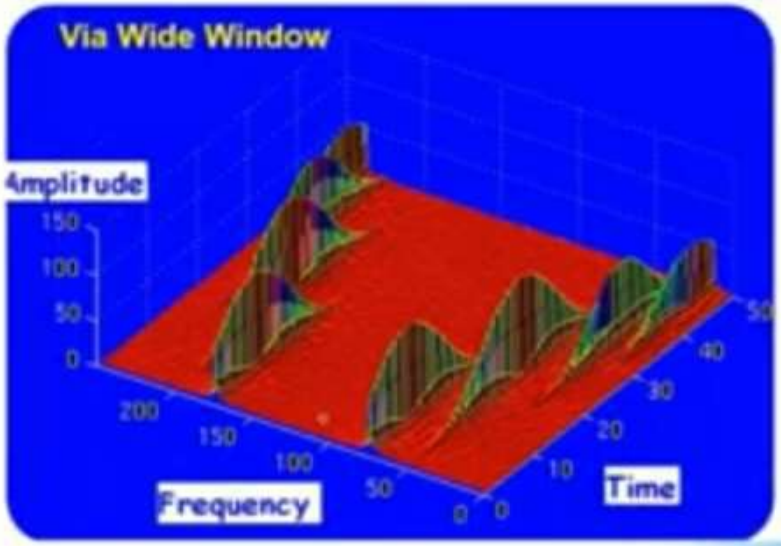
- **Unchanged Window**

- **Dilemma of Resolution**
  - Narrow window (good time resolution) -> Poor frequency resolution.
  - Wide window (poor time resolution) -> Good frequency resolution.

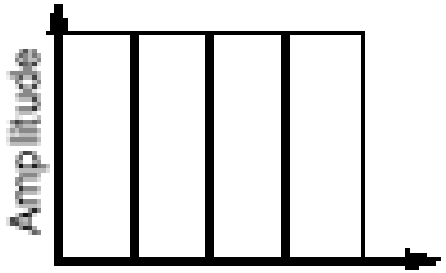
- **Uncertainty?**

  Cannot know what frequency exists at what time intervals.

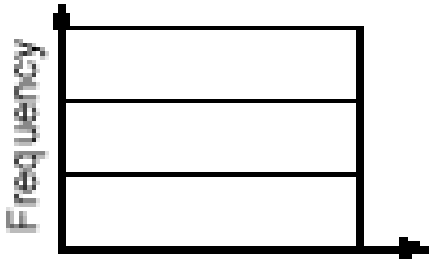




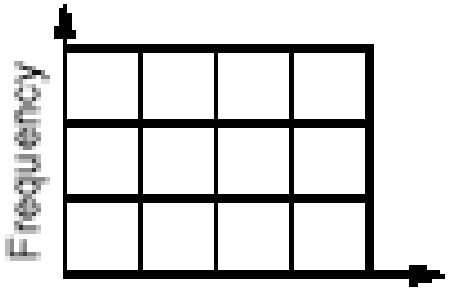
Fourier – Gabor – Wavelet



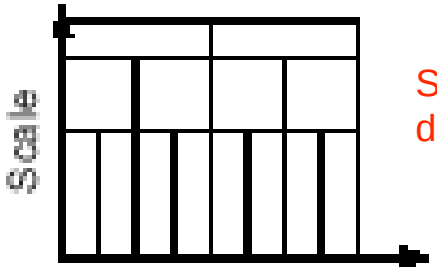
Time Domain (Shannon)



Frequency Domain (Fourier



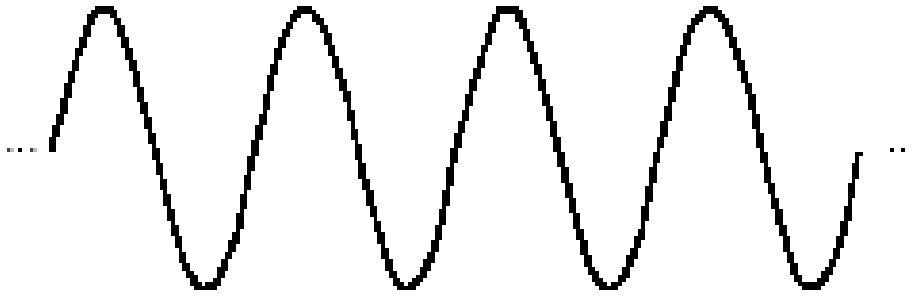
STFT (Gabor)



Wavelet Analysis

Scale-space decomposition

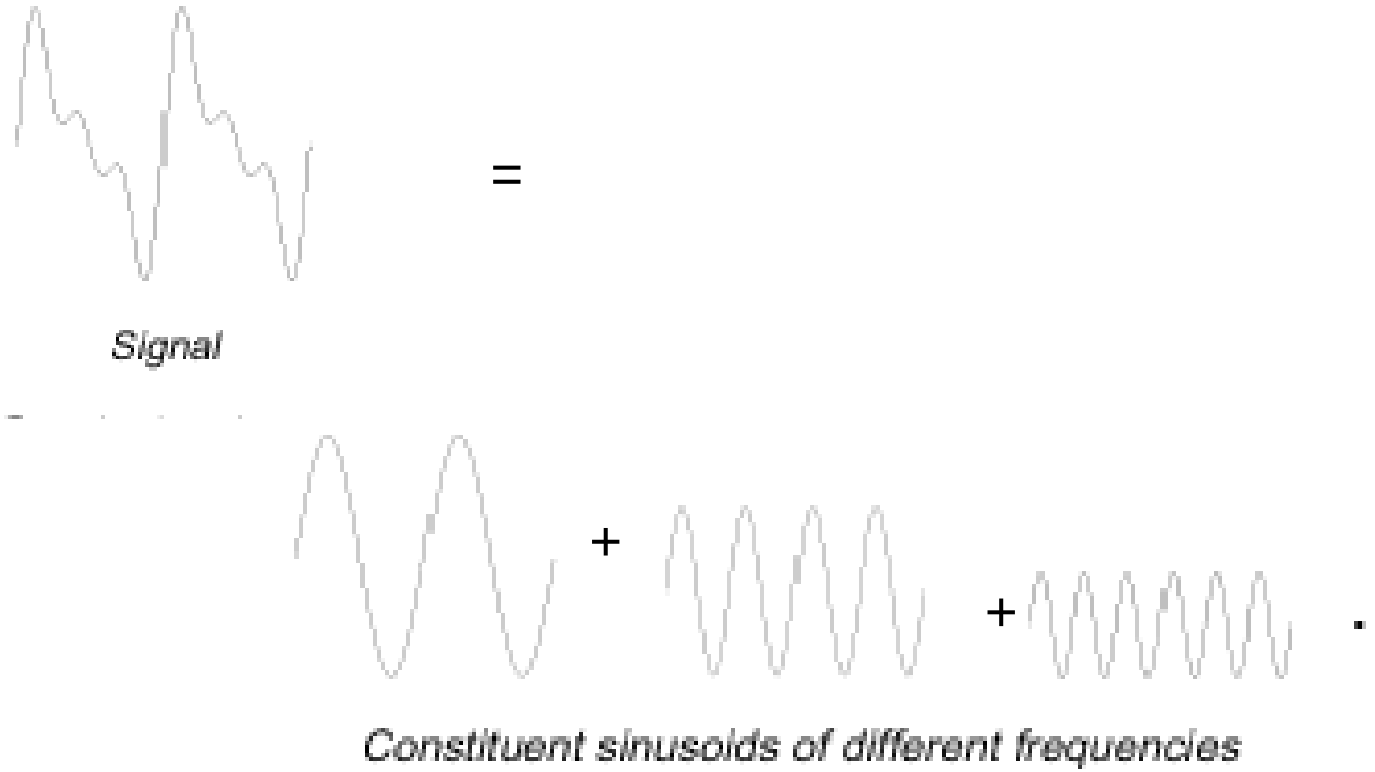
Localization (or the lack of it)



Sine Wave

Wavelet (db10)

Fourier decomposition



and the Wavelet decomposition

*Fourier transfor*m: $F(w) = \int_{-\infty}^{\infty} f(t) \exp(-jwt) dt$

Similarly, the *continuous wavelet transform* (CWT) is defined as the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet function ψ:

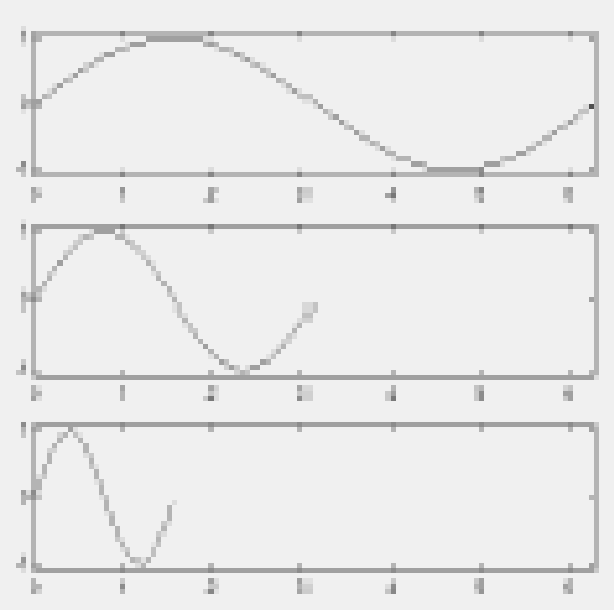
$$c(\text{scale}, \text{position}) = \int_{-\infty}^{\infty} f(t) \psi(\text{scale}, \text{position}, \text{t}) dt$$

The result of the CWT are many *wavelet coefficients* C, which are a function of scale and position.

Wavelet decomposition –contd.

Multiplying each coefficient by the appropriately scaled and shifted wavelet yields the constituent wavelets of the original signal:

What do we mean by scale?

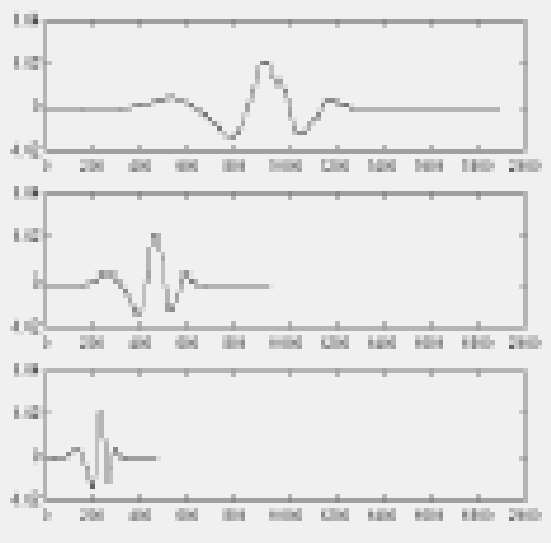


$$f(t) = \sin(t) \quad ; \quad a = 1$$

$$f(t) = \sin(2t) \quad ; \quad a = \frac{1}{2}$$

$$f(t) = \sin(4t) \quad ; \quad a = \frac{1}{4}$$

The scale factor

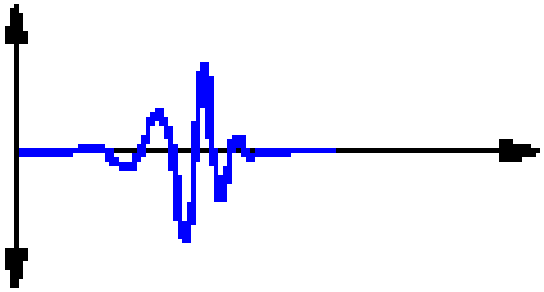


$$f(t) = \psi(t); \quad a = 1$$

$$f(t) = \psi(2t); \quad a = 1/2$$

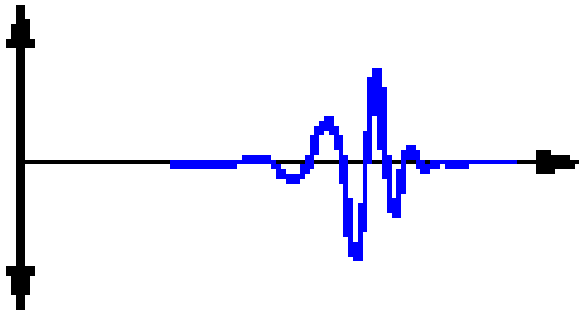
$$f(t) = \psi(4t); \quad a = 1/4$$

Shifting



Wavelet function

$\psi(t)$



Shifted Wavelet function

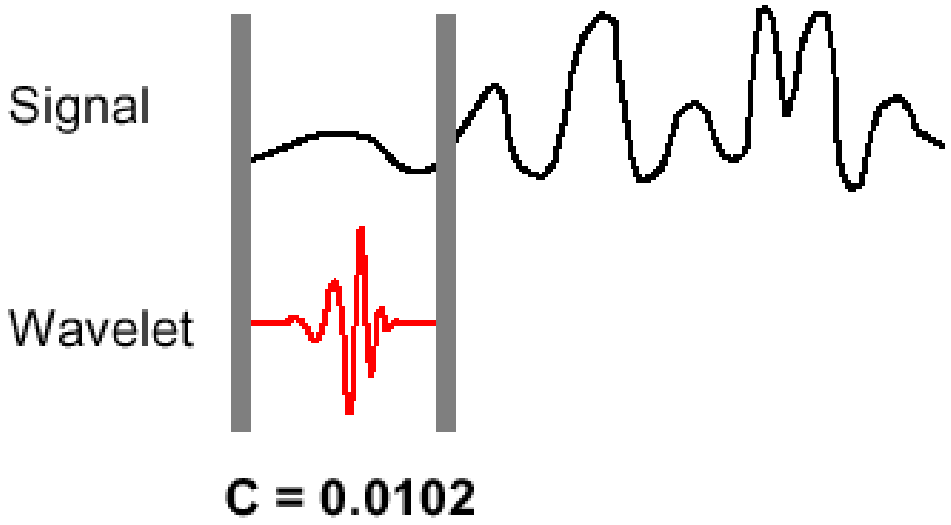
$\psi(t-k)$

Computing a wavelet transform

It's really a very simple process. In fact, here are the five steps of an easy recipe for creating a CWT:

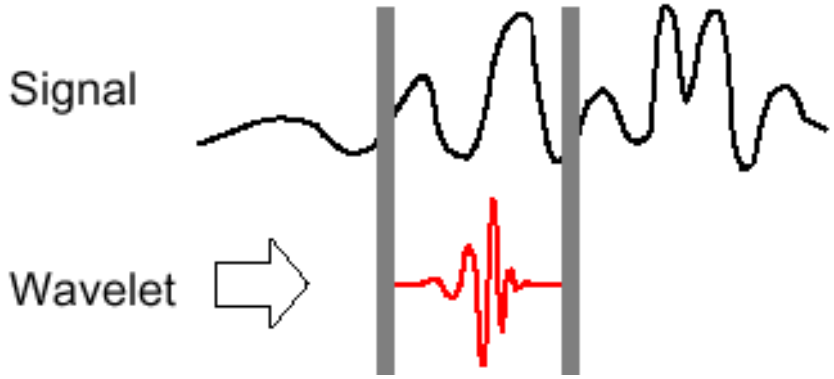
1. Take a wavelet and compare it to a section at the start of the original signal.

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. Note that the results will depend on the shape of the wavelet you choose.

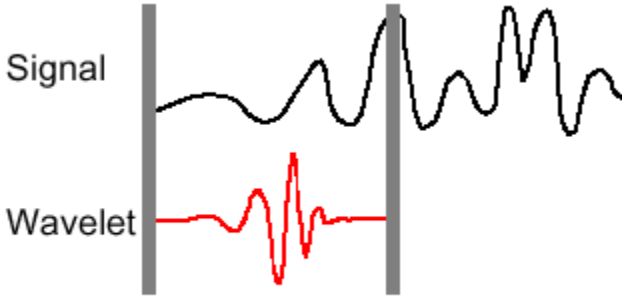


Computing the WT (2)

3 Shift the wavelet to the right and repeat steps 1 and 2 until you've covered the whole signal.



4 Scale (stretch) the wavelet and repeat steps 1 through 3.



C = 0.2247

5 Repeat steps 1 through 4 for all scales.

The discrete wavelet transform

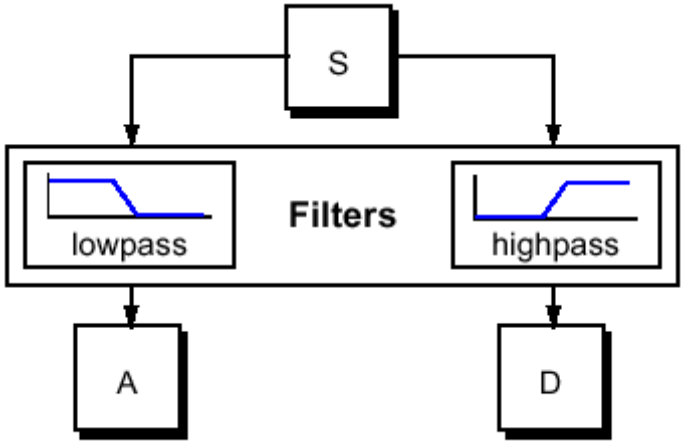
Calculating wavelet coefficients at every possible scale is a fair amount of work, and it generates an awful lot of data.

What if we choose only a subset of scales and positions at which to make our calculations?

It turns out, rather remarkably, that if we choose scales and positions based on powers of two — so-called *dyadic* scales and positions — then our analysis will be much more efficient and just as accurate. We obtain just such an analysis from the *discrete wavelet transform* (DWT).

Approximations and Details

The approximations are the high-scale, low-frequency components of the signal. The details are the low-scale, high-frequency components. The filtering process, at its most basic level, looks like this:



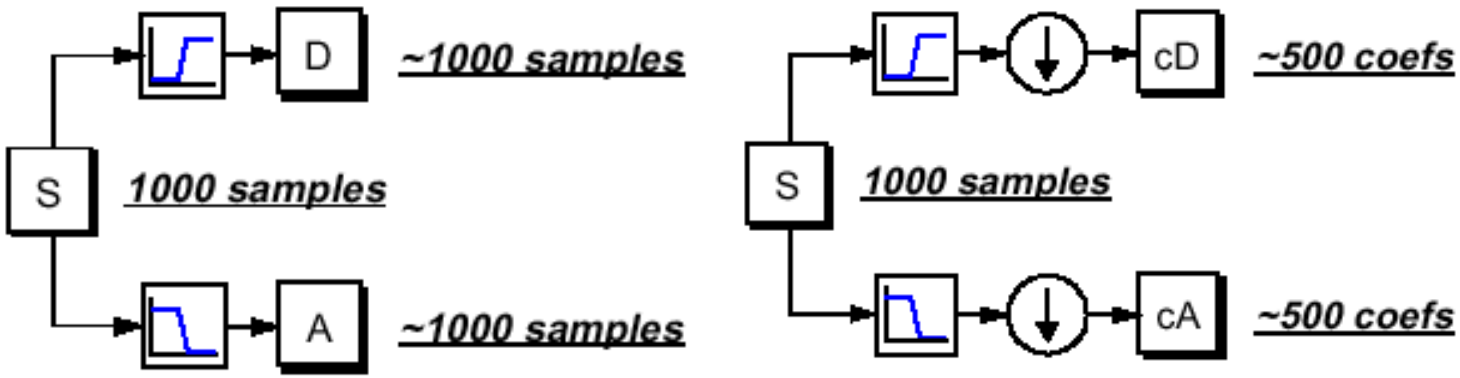
The original signal, S, passes through two complementary filters and emerges as two signals.

Downsampling

Unfortunately, if we actually perform this operation on a real digital signal, we wind up with twice as much data as we started with. Suppose, for instance, that the original signal S consists of 1000 samples of data. Then the approximation and the detail will each have 1000 samples, for a total of 2000.

To correct this problem, we introduce the notion of *downsampling*. This simply means throwing away every second data point. While doing this introduces *aliasing* in the signal components, it turns out we can account for this later on in the process.

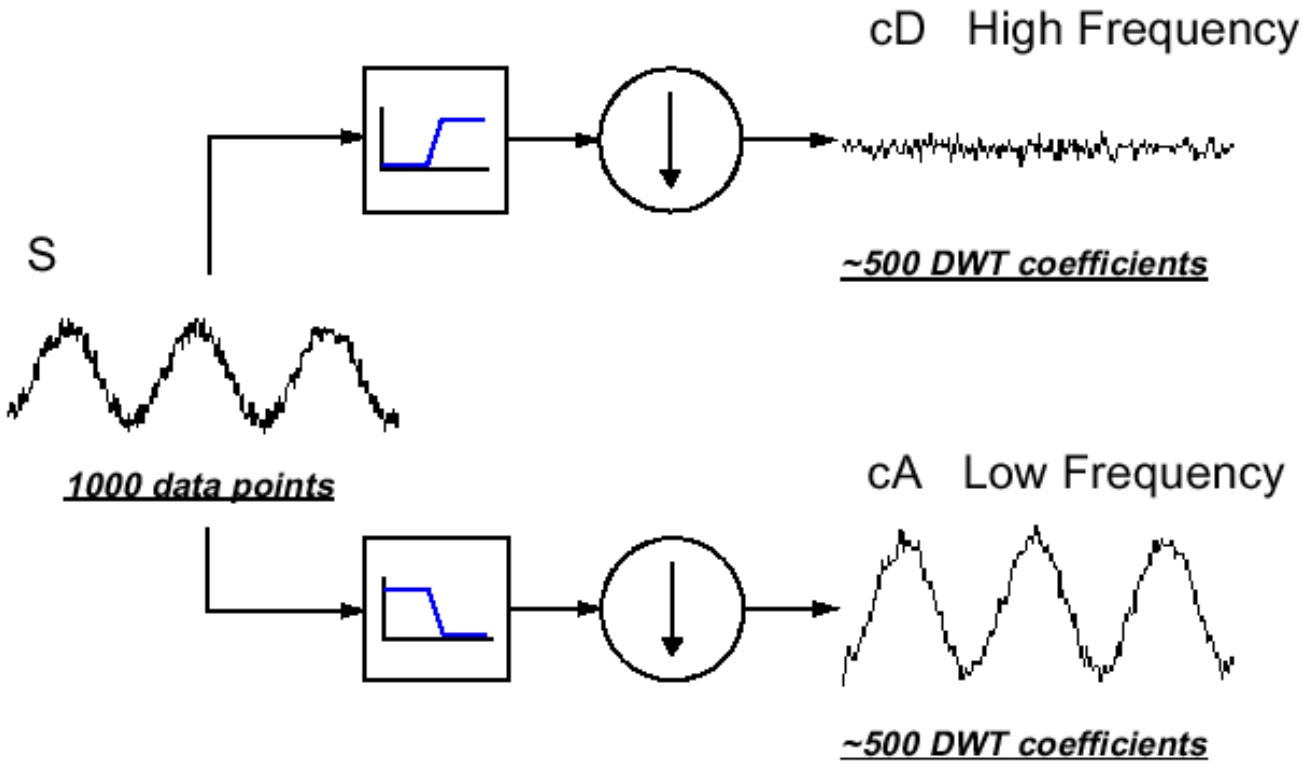
Downsampling (2)



The process on the right, which includes downsampling, produces DWT coefficients.

An example

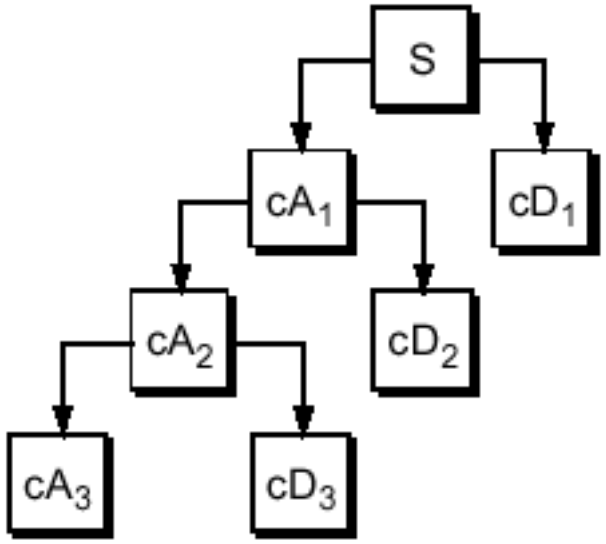
Here is our schematic diagram with real signals inserted into it:



Wavelet Decomposition

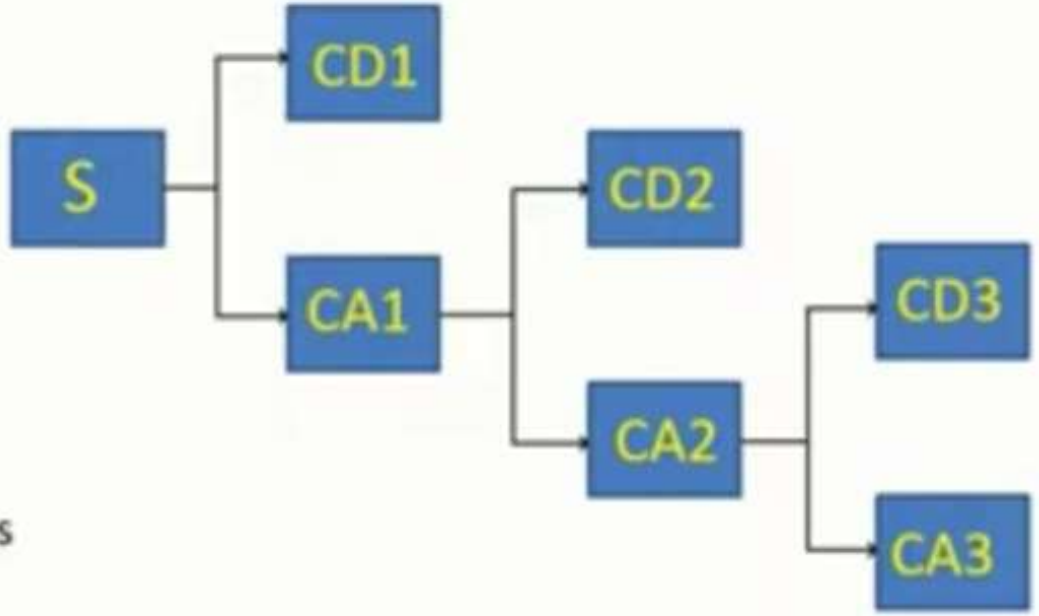
Multiple-Level Decomposition

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower-resolution components. This is called the *wavelet decomposition tree*.



Discrete Wavelet Transform (Filter Banks)

Multi Level Decomposition



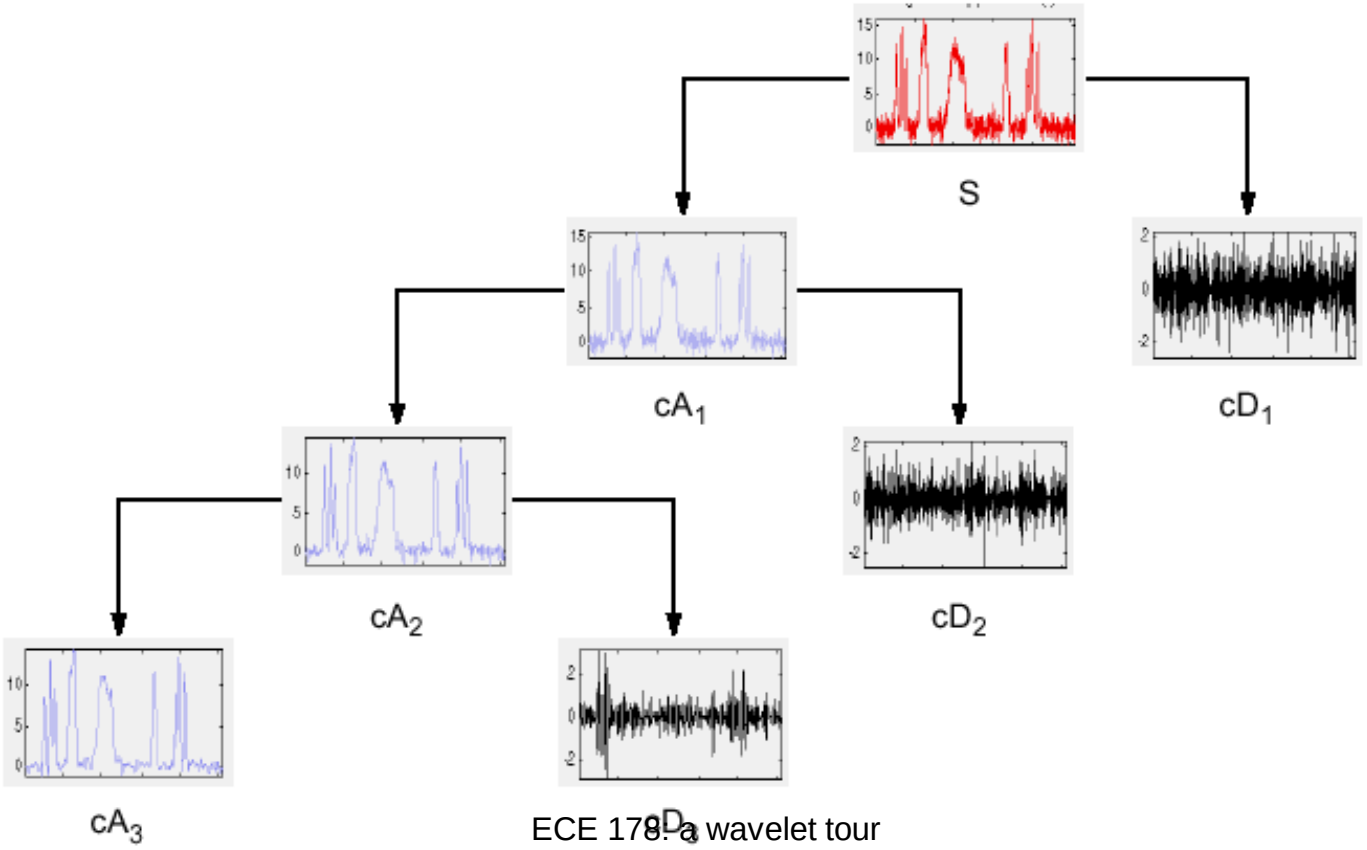
Here,
S: Signal,
CD: Detailed Coefficients
CA: Approximation Coefficients

This is called Wavelet decomposition tree.

Max. Number of decomposition levels: $\log_2 N$

Wavelet decomposition…

Looking at a signal's wavelet decomposition tree can yield valuable information.



Discrete Wavelet Transform (Filter Banks)

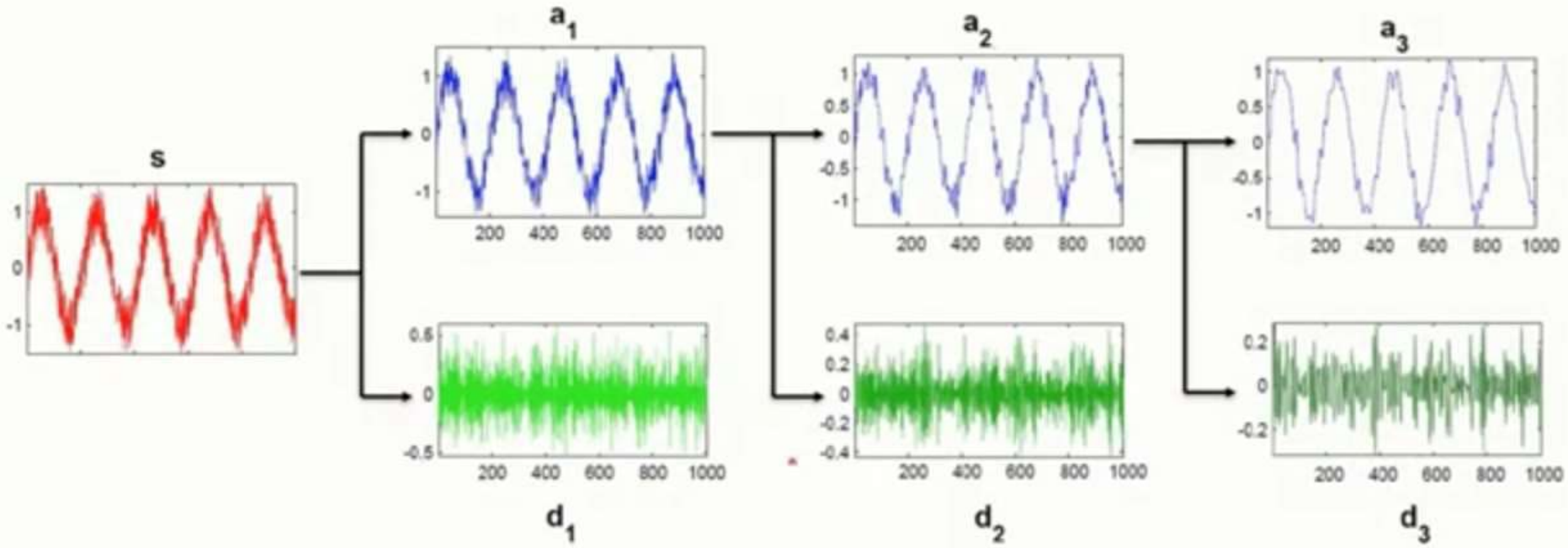
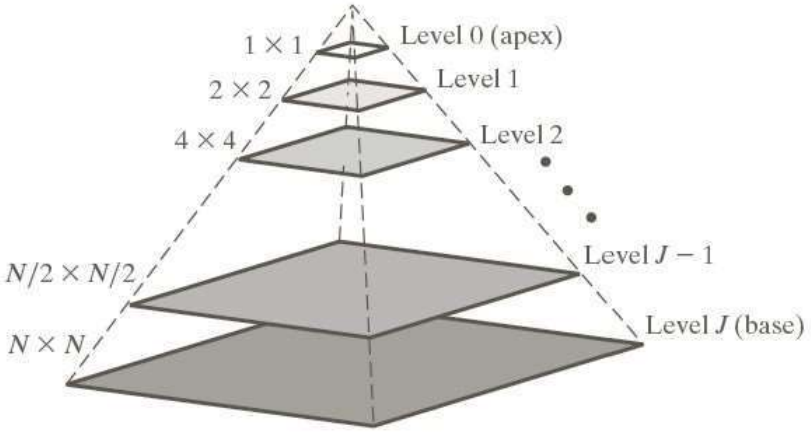
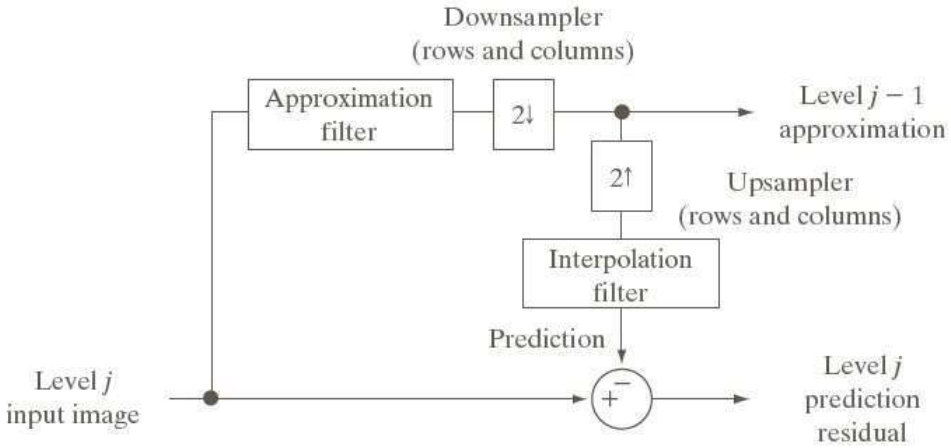


Image Pyramids..

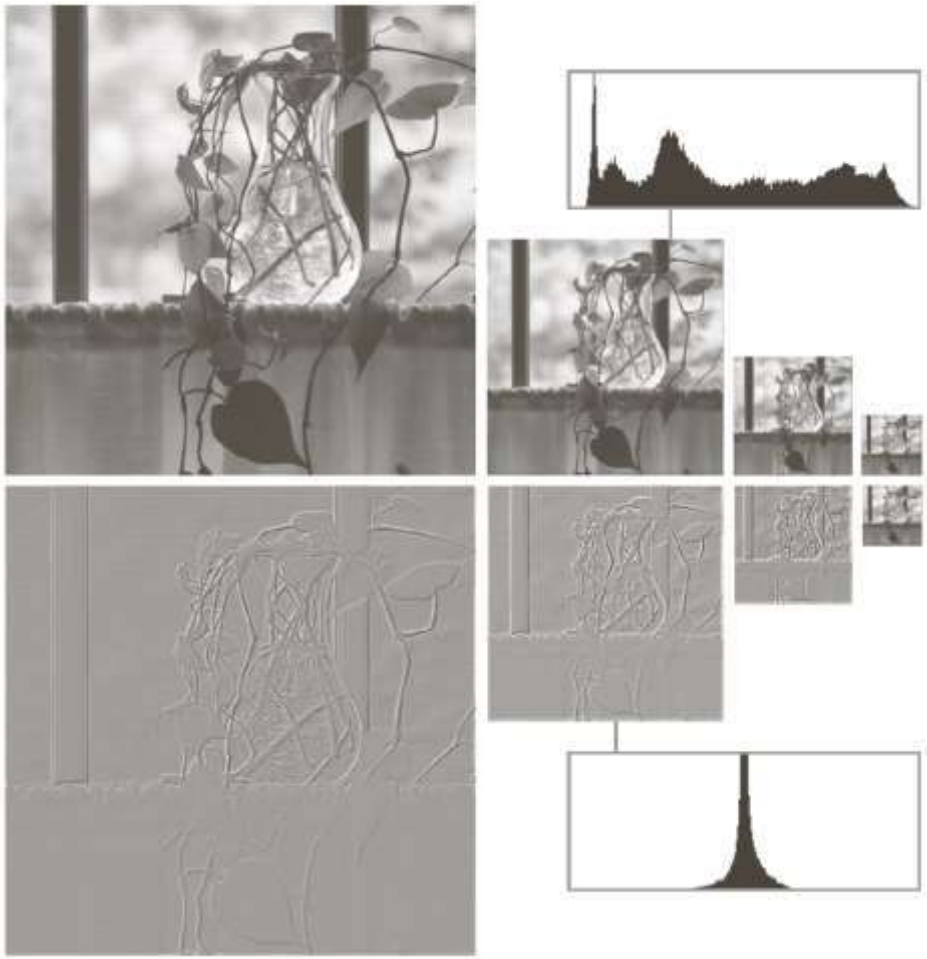


a
b

FIGURE 7.2
(a) An image pyramid. (b) A simple system for creating approximation and prediction residual pyramids.



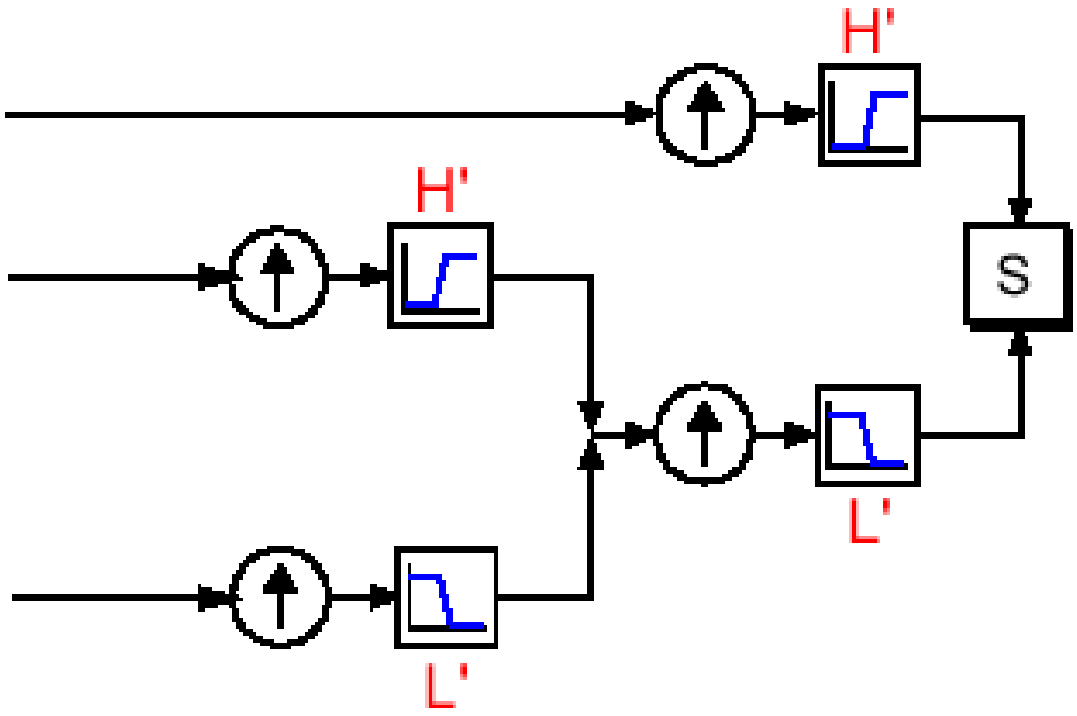
Laplacian Pyramid



a
b

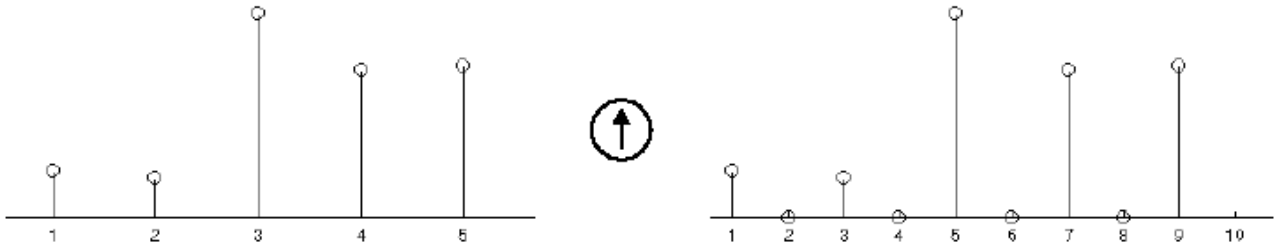
FIGURE 7.3
Two image pyramids and their histograms: (a) an approximation pyramid; (b) a prediction residual pyramid.

IDWT: reconstruction



Analysis vs Synthesis

Where wavelet analysis involves filtering and downsampling, the wavelet reconstruction process consists of upsampling and filtering. Upsampling is the process of lengthening a signal component by inserting zeros between samples:



Signal component

Upsampled signal component

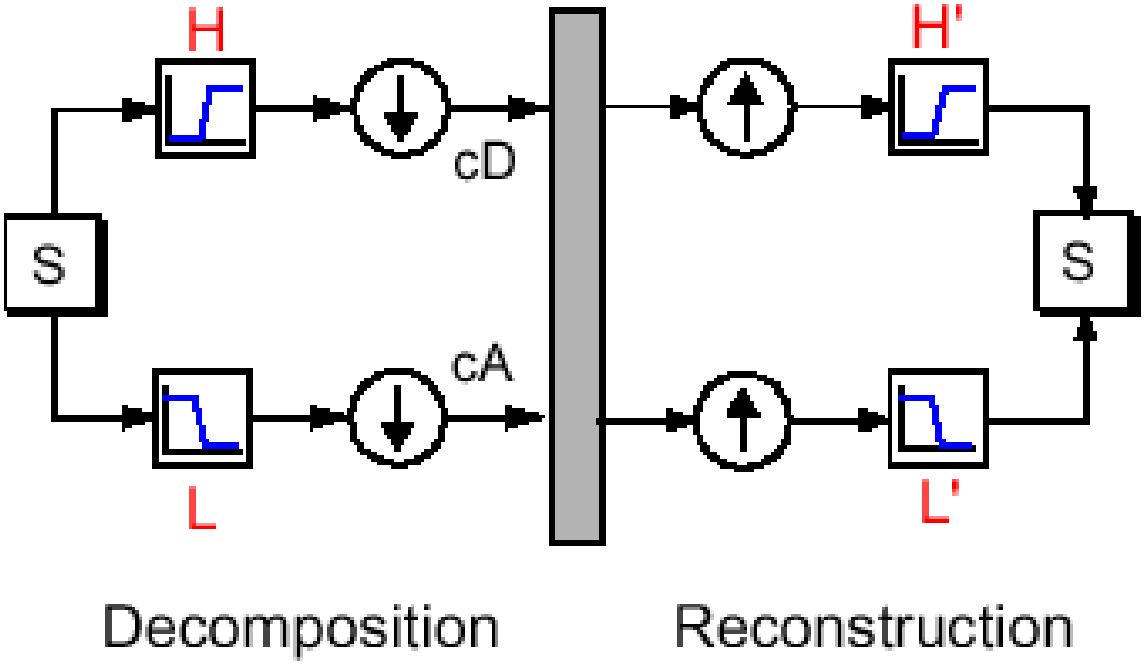
Perfect reconstruction

The filtering part of the reconstruction process also bears some discussion, because it is the choice of filters that is crucial in achieving perfect reconstruction of the original signal.

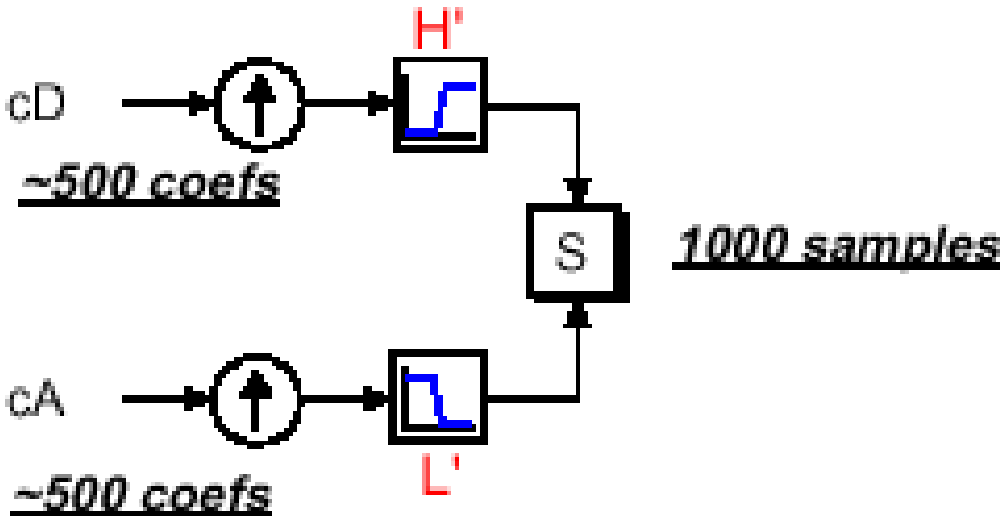
That perfect reconstruction is even possible is noteworthy. Recall that the downsampling of the signal components performed during the decomposition phase introduces a distortion called aliasing. It turns out that by carefully choosing filters for the decomposition and reconstruction phases that are closely related (but not identical), we can "cancel out" the effects of aliasing. This was the breakthrough made possible by the work of Ingrid Daubechies.

A technical discussion of how to design these filters can be found in p. 347 of the book *Wavelets and Filter Banks*, by Strang and Nguyen. The low- and highpass decomposition filters (`L` and `H`), together with their associated reconstruction filters (`L'` and `H'`), form a system of what are called *quadrature mirror filters*:

Quadrature Mirror Filters

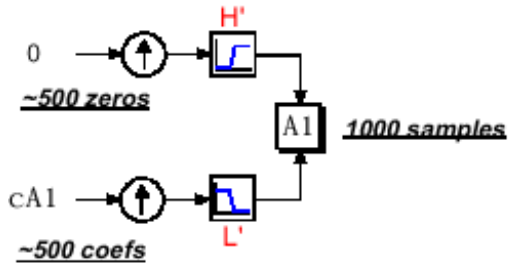


Reconstructing Approximation & Details



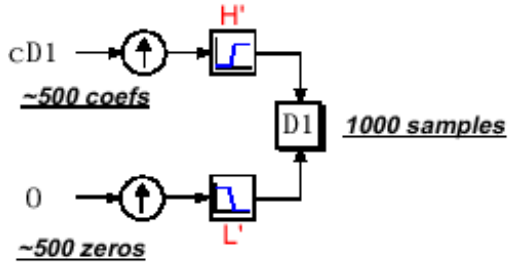
It is also possible to reconstruct the approximations and details themselves from their coefficient vectors. As an example, let's consider how we would reconstruct the first-level approximation A1 from the coefficient vector cA1.

We pass the coefficient vector cA1 through the same process we used to reconstruct the original signal. However, instead of combining it with the level-one detail cD1, we feed in a vector of zeros in place of the details:



The process yields a reconstructed *approximation* A1, which has the same length as the original signal S and which is a real approximation of it.

Similarly, we can reconstruct the first-level detail D1, using the analogous process:



The reconstructed details and approximations are true constituents of the original signal. In fact, we find when we combine them that:

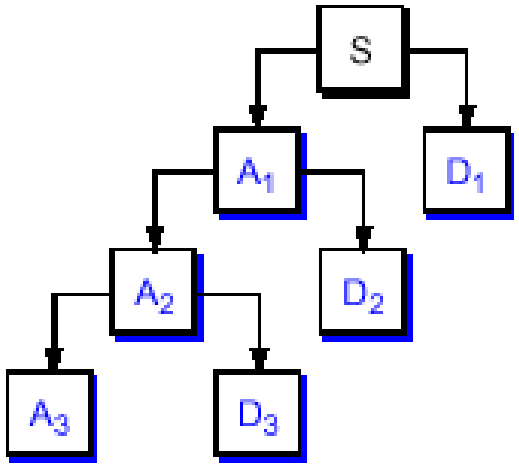
$$A_1 + D_1 = S$$

Reconstructing As and Ds..contd..

Note that the coefficient vectors cA1 and cD1 — because they were produced by downsampling, contain aliasing distortion, and are only half the length of the original signal — cannot directly be combined to reproduce the signal. *It is necessary to reconstruct the approximations and details before combining them.*

Reconstructing the signal

Reconstructed Signal Components



$$S = A_1 + D_1$$

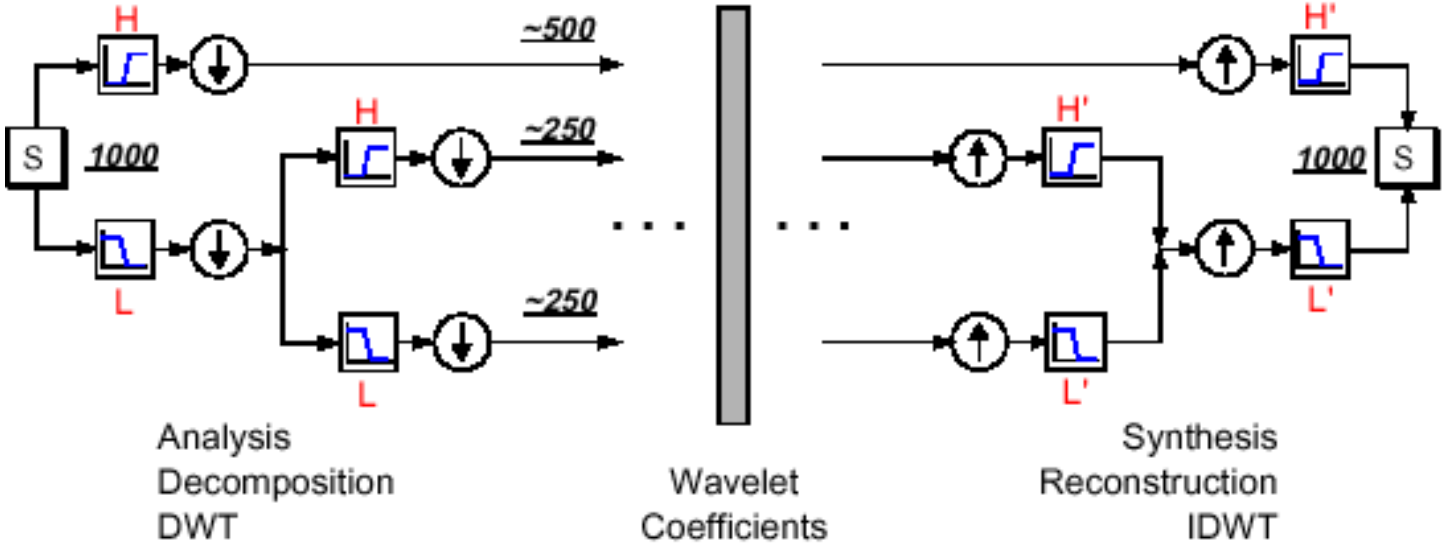
$$= A_2 + D_2 + D_1$$

$$= A_3 + D_3 + D_2 + D_1$$

Multiscale Analysis

Multistep Decomposition and Reconstruction

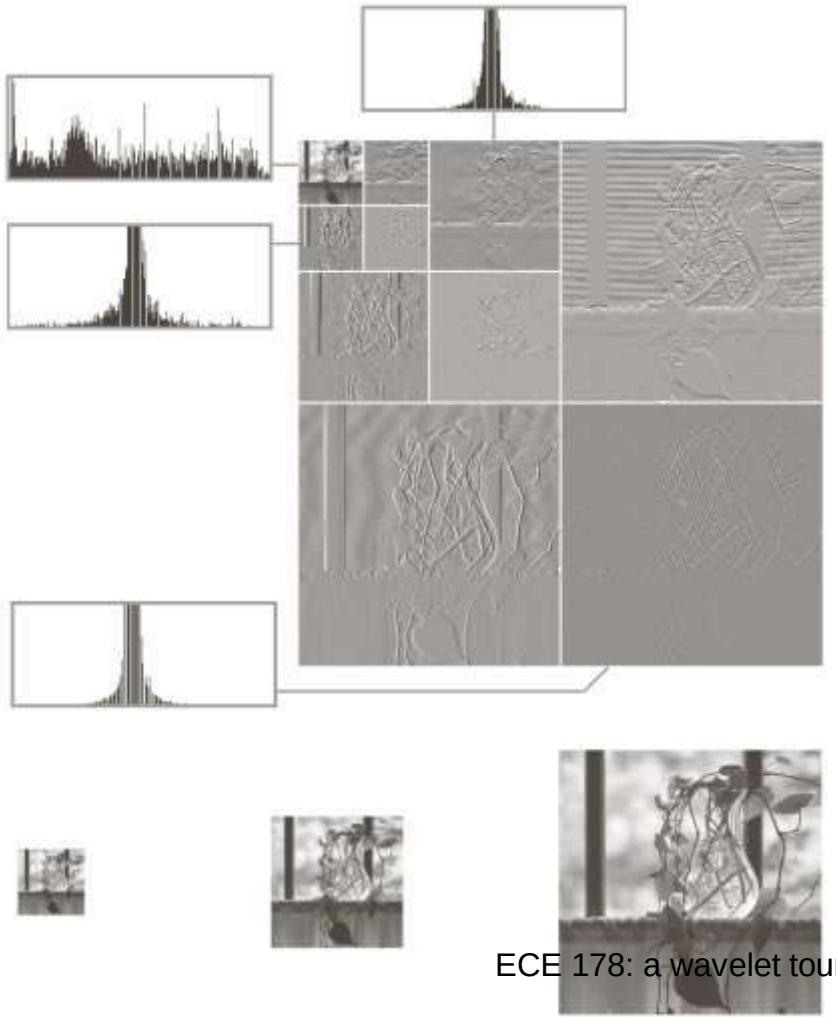
A multistep analysis-synthesis process can be represented as:



Various Types of Wavelets

Wavelets	Abbreviations
Haar Wavelet	*Haar*
Daubechies Wavelet	*Db*
Symlets	*Sym*
Coiflets	*Coif*
Bi-Orthogonal Wavelet	*Bior*
Meyer Wavelet	*Meyr*
Discrete Meyer Wavelet	*Dmey*
Battle and Lemarié Wavelets	*Btlm*
Gaussian Wavelet	*Gaus*
Mexican Hat Wavelets	*Mexh*
Morlet Wavelet	*Morl*
Complex Gaussian Wavelets	*Cgau*
Complex Shannon Wavelets	*Shan*
Complex B-spline frequency Wavelets	*Fbsp*
Complex Morlet Wavelets	*Cmor*

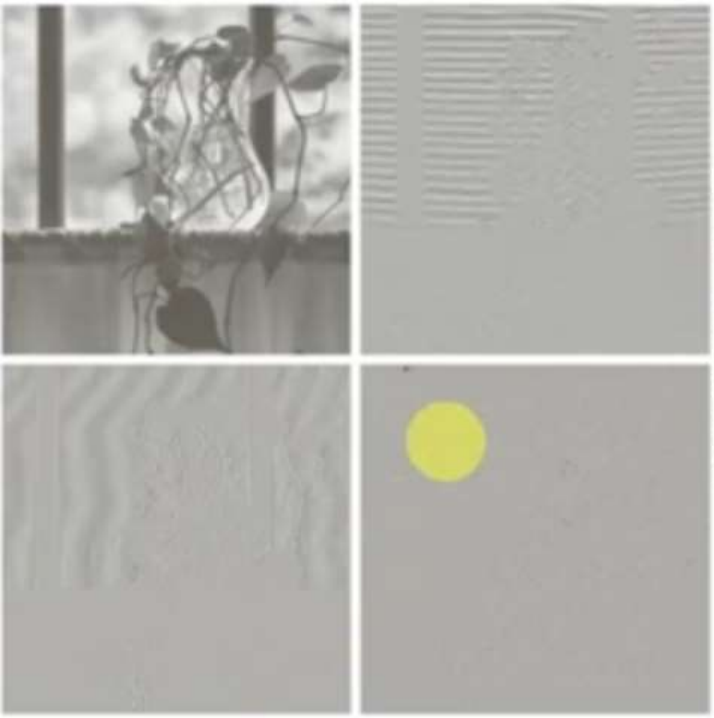
Haar Wavelet Transform



a
b c d

FIGURE 7.10
(a) A discrete wavelet transform using Haar $\mathbf{H}_2$ basis functions. Its local histogram variations are also shown. (b)–(d) Several different approximations (64×64, 128×128, and 256×256) that can be obtained from (a).

Discrete Wavelet transform (eg. Haar wavelet)



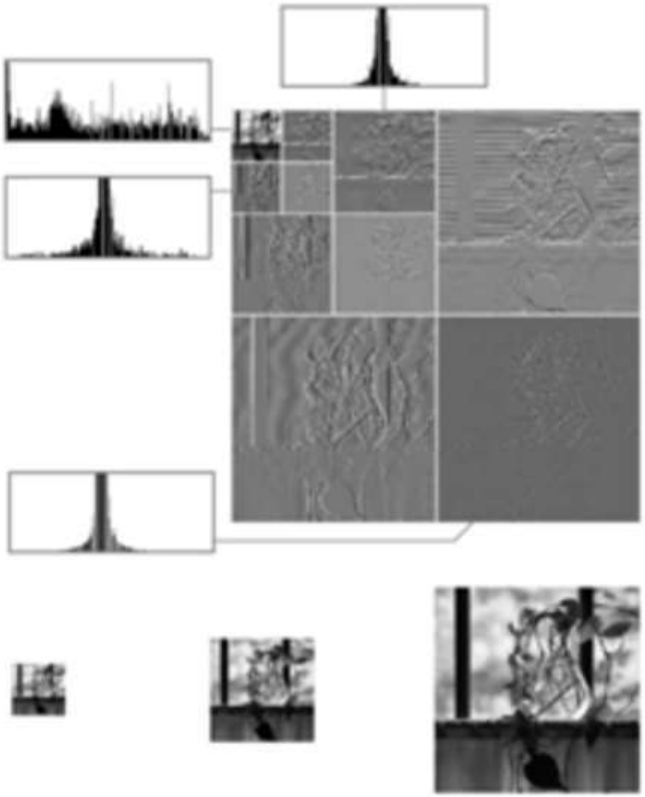
a b
c d

FIGURE 7.9
A four-band split of the vase in Fig. 7.1 using the subband coding system of Fig. 7.7. The four subbands that result are the (a) approximation, (b) horizontal detail, (c) vertical detail, and (d) diagonal detail subbands.

An Example of One-level Decomposition

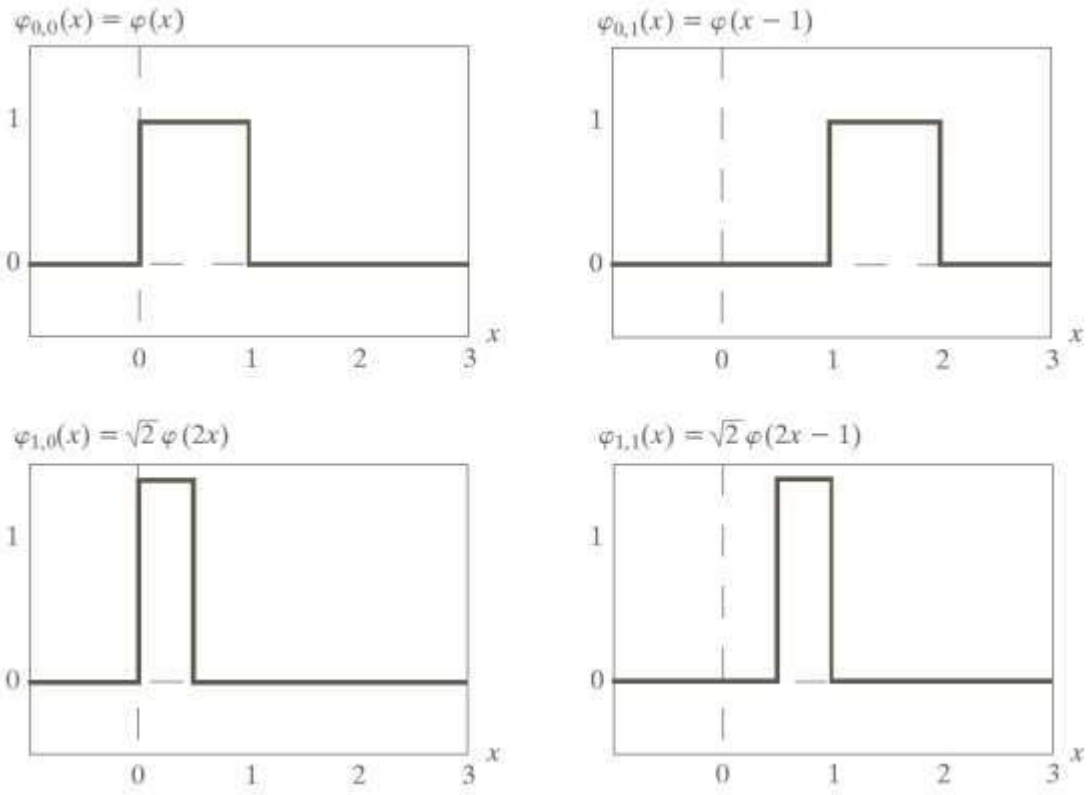
a
b c d

FIGURE 7.10
(a) A discrete wavelet transform using Haar H_2 basis functions. Its local histogram variations are also shown. (b)–(d) Several different approximations (64×64, 128×128, and 256×256) that can be obtained from (a).

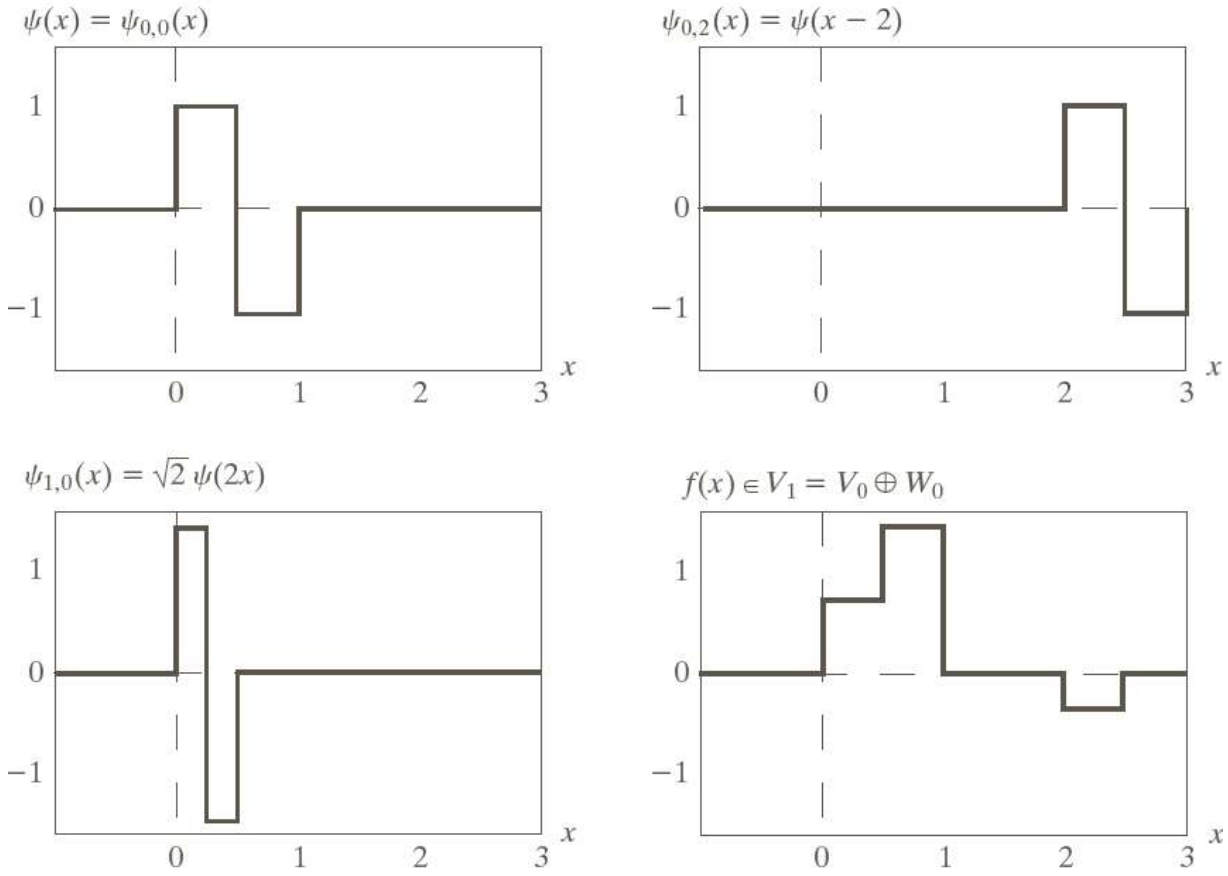


An Example of Multi-level Decomposition

Haar-- scaling function (approximations)



Haar -- wavelet functions



Haar decomposition

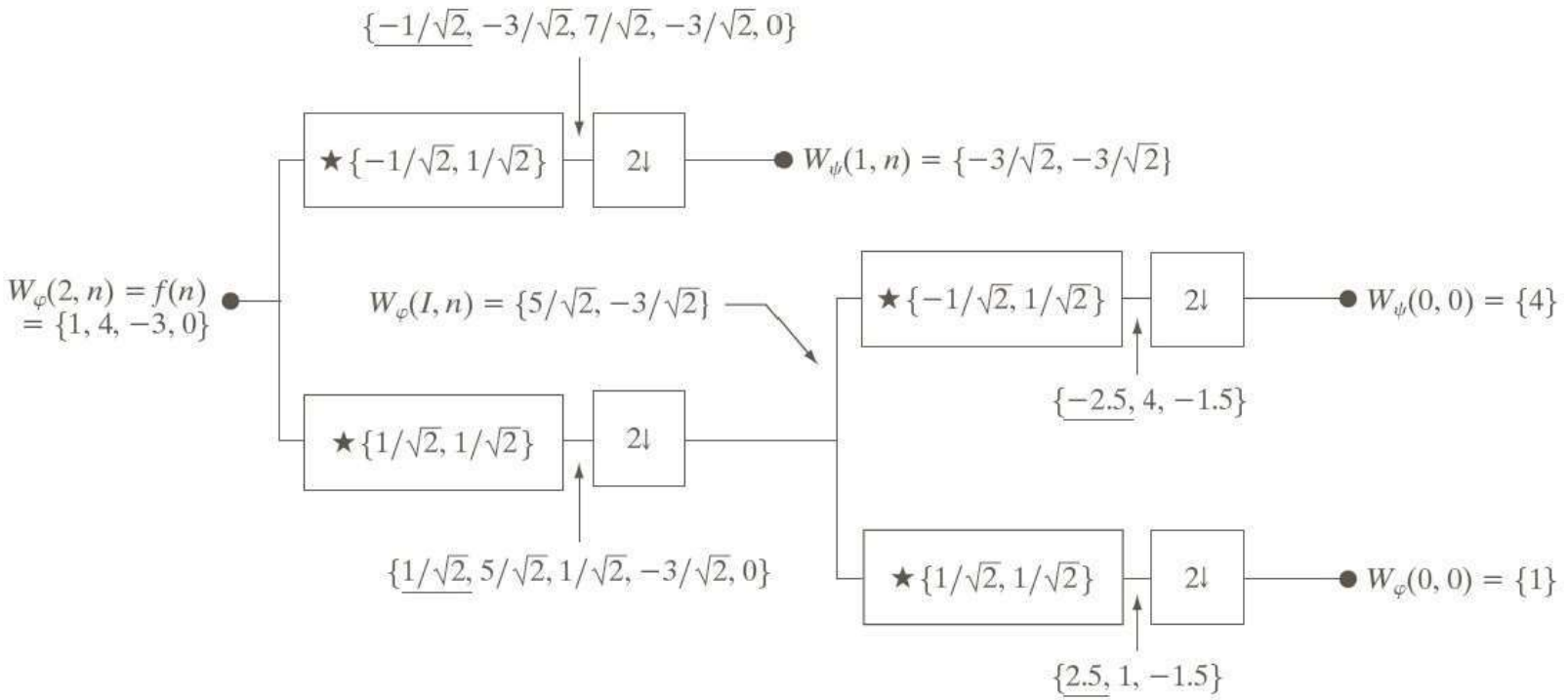


FIGURE 7.19 Computing a two-scale fast wavelet transform of sequence $\{1, 4, -3, 0\}$ using Haar scaling and wavelet vectors.

Haar reconstruction

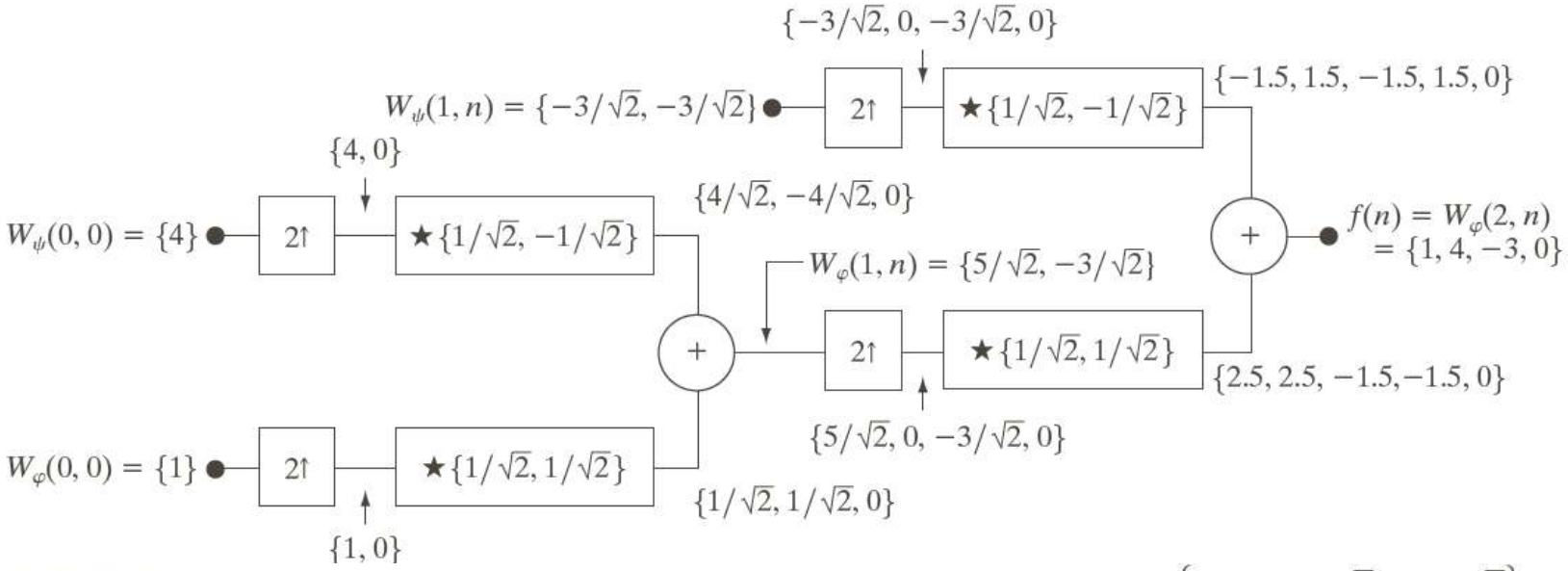
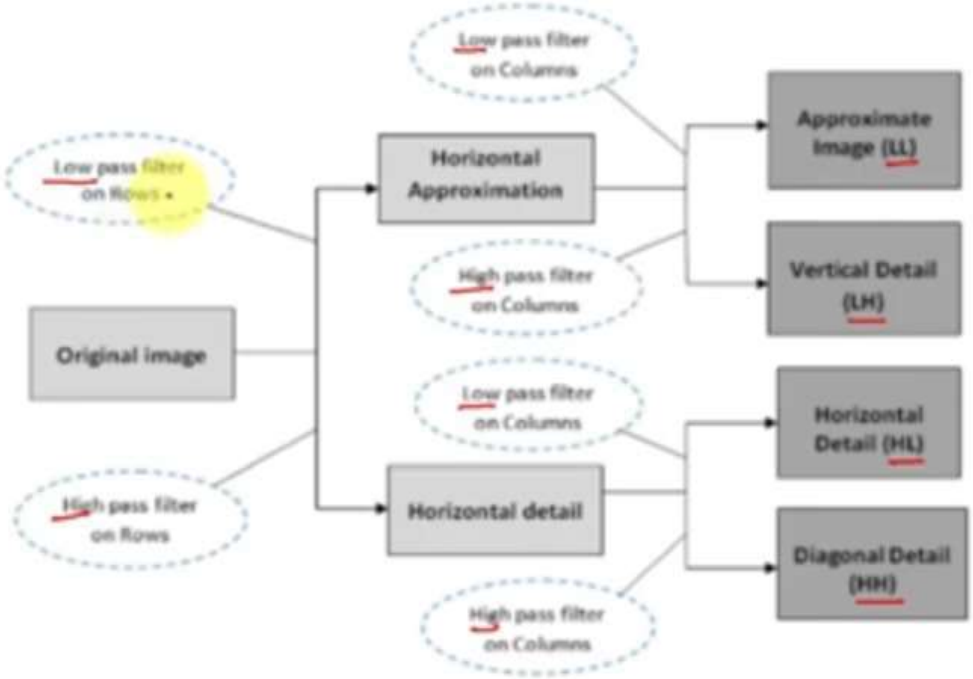
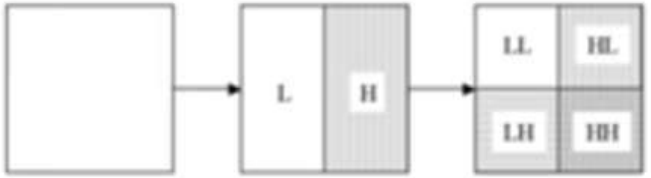
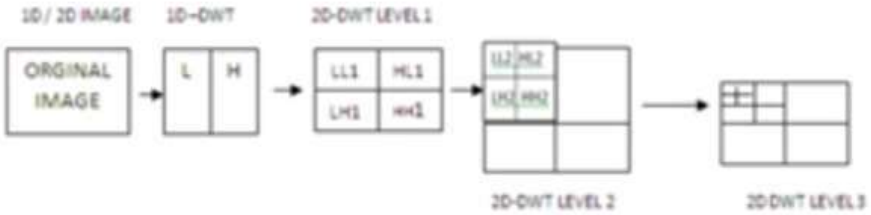
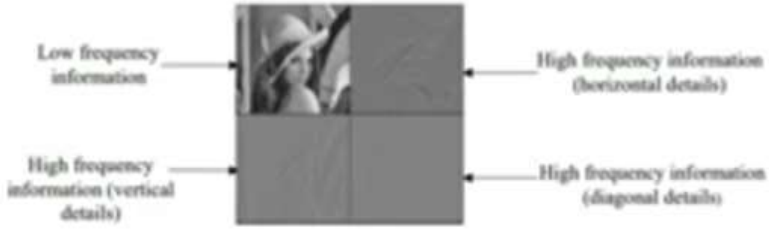


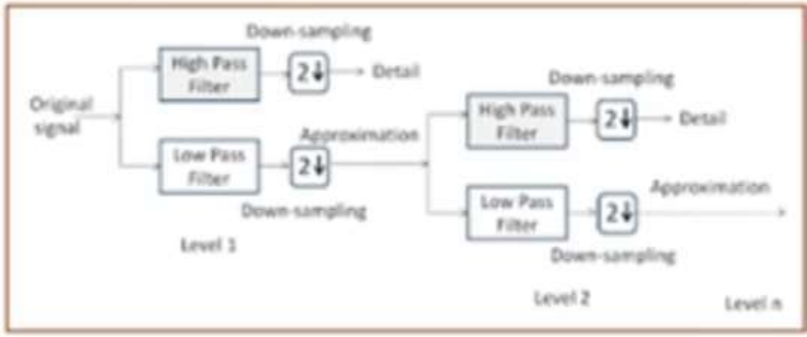
FIGURE 7.22 Computing a two-scale inverse fast wavelet transform of sequence $\{1, 4, -1.5\sqrt{2}, -1.5\sqrt{2}\}$ with Haar scaling and wavelet functions.


Low pass filter on Columns
Low pass filter on Rows
Horizontal Approximation
High pass filter on Columns
Approximate Image (LL)
Vertical Detail (LH)
Original image
Low pass filter on Columns
High pass filter on Rows
Horizontal detail
Horizontal Detail (HL)
Diagonal Detail (HH)
High pass filter on Columns


L
H
LL
HL
LH
HH


1D / 2D IMAGE
1D-DWT
2D-DWT LEVEL 1
ORGINAL IMAGE
L
H
LL1
HL1
LH1
HH1
LL2
HL2
LH2
HH2
2D-DWT LEVEL 2
2D-DWT LEVEL 3


Low frequency information
High frequency information (vertical details)
High frequency information (horizontal details)
High frequency information (diagonal details)


Down-sampling
Original signal
High Pass Filter
Detail
Low Pass Filter
Approximation
Down-sampling
Level 1
Down-sampling
High Pass Filter
Detail
Low Pass Filter
Approximation
Down-sampling
Level 2
Level n

A Wavelet

A Wavelet is a waveform of effectively limited duration that has an average value of zero. It is defined as,

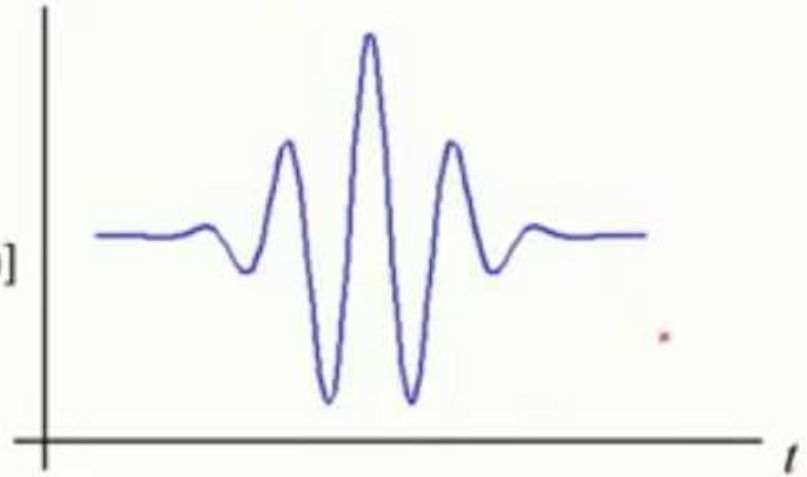
$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\,\psi\left(\frac{t-b}{a}\right) \quad a, b \in \mathbb{R}$$

Here a and b are called Dilation (Scale) and Translation (Position) parameters respectively.

An example wavelet is shown below.

$$\psi(t) = e^{j\omega_0 t} e^{-\frac{t^2}{2}}$$

Morlet Wavelet



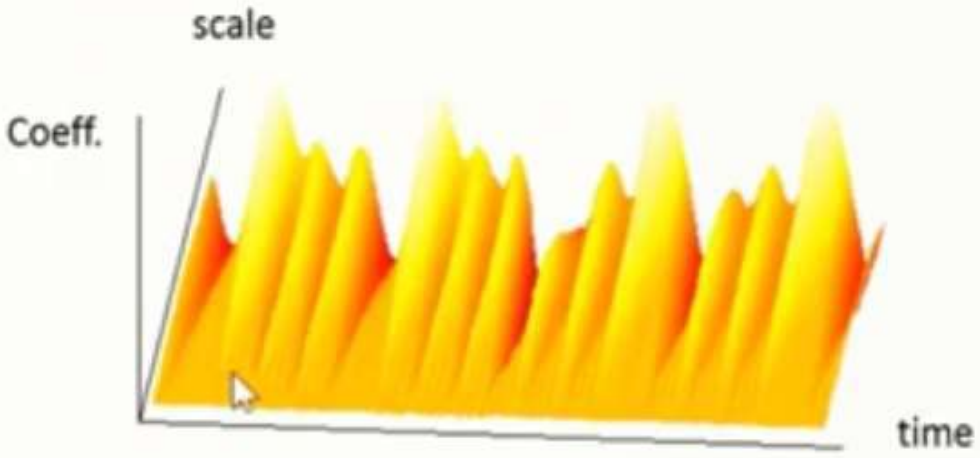
Continuous Wavelet Transform (CWT)

The Continuous Wavelet Transform (CWT) of a signal $f(t)$ is then given by the equation,

$$CWT(a,b) = \langle f,\ \psi_{a,b} \rangle = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(t).\ \psi^* \left(\frac{t-b}{a} \right) dt$$

Here, $\langle f, \psi_{a,b} \rangle$ is the $\mathbb{L}^2$ inner product.

The results of the CWT are many wavelet coefficients, which are a function of a (scale) and b (position).



Discrete Wavelet Transform (DWT)

- In CWT, calculating wavelet coefficients at every possible scale is a fair amount of work, and it generates an awful lot of data.

- If scales (a) and positions (b) are chosen to be discrete then analysis will be much easier and will not generate the huge data.

- This idea of choosing discrete values of dilation (a) and translation (b) parameters is implemented in,
  - ✓ **Redundant Wavelet Transform (Frames)** and
  - ✓ **Orthonormal bases for wavelets or Multi Resolution Analysis (MRA).**

Discrete Wavelet Transform (Redundant Wavelet Transform (Frames)

- The a is chosen to be an integer powers of one fixed dilation parameter $a_0 > 1$, i.e. $a = a_0^m$.
- The different values of m correspond to wavelets of different widths.
- The narrow wavelets are translated by small steps, while wider wavelets are translated by larger steps. Therefore, b is discretized by $b = nb_0a_0^m$, where $b_0 > 0$ is fixed and $n \in \mathbb{Z}$.
- The corresponding discretely labeled wavelets are therefore,

$$\psi_{m,n}(k) = a_0^{-\frac{m}{2}}\psi\left(a_0^{-m}(k - nb_0a_0^m)\right) \qquad m, n \in \mathbb{Z}$$

- For a given function $f(k)$, the inner product $\langle f, \psi_{m,n}\rangle$ then gives the discrete wavelet transform as given as,

$$DWT(m,n) = \langle f, \psi_{m,n}\rangle = a_0^{-\frac{m}{2}} \sum_{k=-\infty}^{\infty} f(k).\psi^*(a_0^{-m}\, k - nb_0)$$

Discrete Wavelet Transform (Multi Resolution Analysis (MRA))

- If scales and positions are chosen based on powers of two, so-called Dyadic scales and positions, then analysis becomes much more efficient and just as accurate.
- It was developed in 1988 by S. Mallat. For some very special choice of $\psi(k)$ and a_0, b_0, the $\psi_{m,n}(k)$ constitute an orthonormal basis for $\mathbb{L}^2(\mathbb{R})$.
- In particular, if $a_0 = 2, b_0 = 1$, then there exist ψ(k) with good time-frequency localization properties, such that the,

$$\psi_{m,n}(k) = 2^{-\frac{m}{2}}\psi(2^{-m}k - n) \quad m, n \in \mathbb{Z}$$, constitutes an orthonormal basis for $\mathbb{L}^2$ ($\mathbb{R}$).

- For a given function $f(k)$, the inner product $\langle f, \psi m,n\rangle$ then gives the discrete wavelet transform as given as,

$$DWT(m,n) = \langle f, \psi_{m,n}\rangle = 2^{-\frac{m}{2}} \sum_{k=-\infty}^{\infty} f(k).\psi^*(2^{-m}k - n)$$

❑ Steps of Wavelet Transform

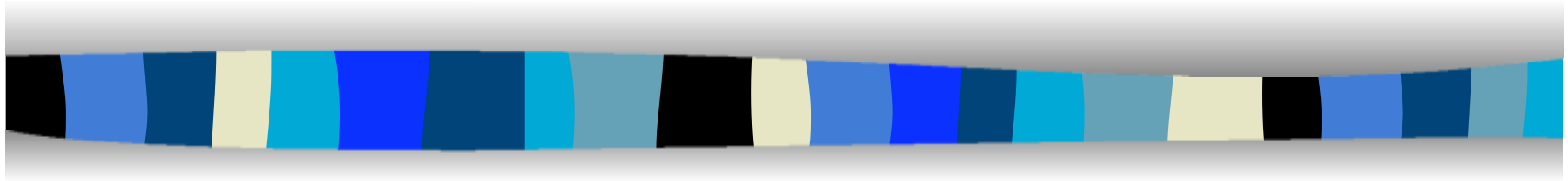
Step1: Start with a mother wavelet such as Haar, Morlet, Daubechies etc. The image is then translated into shifted and scaled versions of this mother wavelet. First original image have to been passed through high pass filter and low pass filter by applying filter on each row. We know when we apply LPF we get approximation and when we apply HPF we get the details.

Step2: Now to the horizontal approximation, we again apply LPF and HPF to the columns. Hence we get the approximate image (LL) and vertical detail of the horizontal approximation(LH).

Step3: Next we apply LPF and HPF to the horizontal detail. Hence we get horizontal detail of the image (HL) and the diagonal detail of the image (HH).

If the 3 detail sub-signals i.e. LH, HL and HH are small, they can be assumed to be zero, without any significant change in the image. Hence large compression can be achieved using wavelet transform.

Wavelets: a preview



Acknowledgements:

Material compiled from the MATLAB Wavelet Toolbox User Guide and Chapter 7, DIP 3e.

READING: Chapter 7: 7.1.1.

Template Matching

Longin Jan Latecki

Temple University

CIS 601

Based on a project by Roland Miezianko

Agenda

- Template Matching
  - Definition and Method
  - Bi-Level Image
  - Gray-Level Image
- Matlab Example
  - Gray-Level Template Matching
  - Machine Vision Example

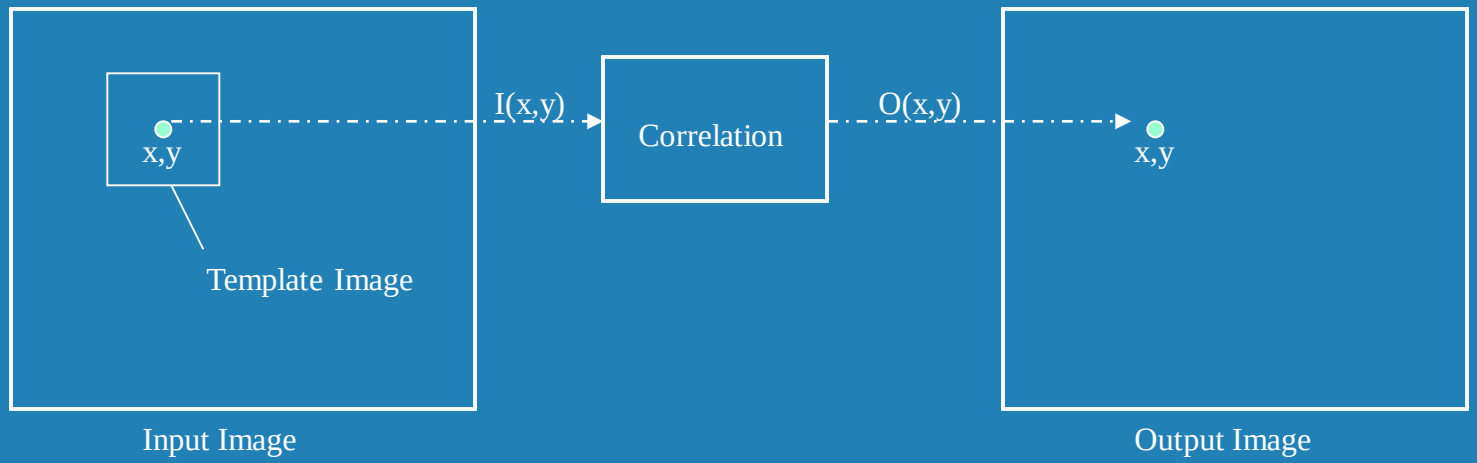
Definition

- Technique used in classifying objects.
- Template Matching techniques compare portions of images against one another.
- Sample image may be used to recognize similar objects in source image.

Definition, cont.

- If standard deviation of the template image compared to the source image is small enough, template matching may be used.
- Templates are most often used to identify printed characters, numbers, and other small, simple objects.

Method



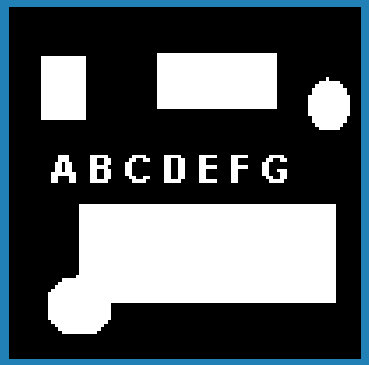
The matching process moves the template image to all possible positions in a larger source image and computes a numerical index that indicates how well the template matches the image in that position.

Match is done on a pixel-by-pixel basis.

Bi-Level Image TM

- **Template is a small image, usually a bi-level image.**
- **Find template in source image, with a Yes/No approach.**

Template

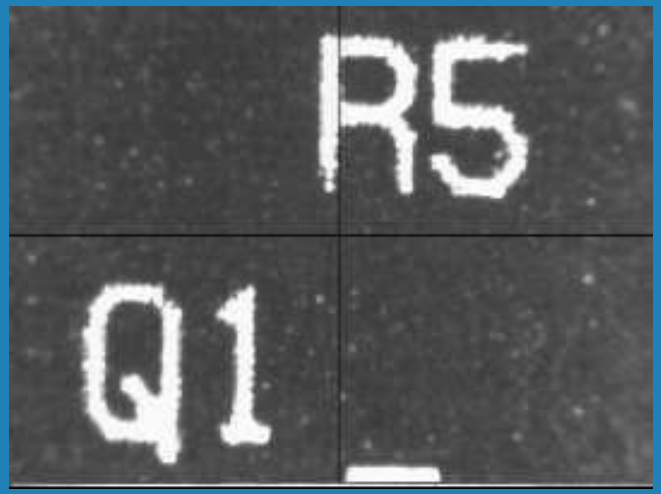


Source

Grey-Level Image TM

- When using template-matching scheme on grey-level image it is unreasonable to expect a perfect match of the grey levels.
- Instead of yes/no match at each pixel, the difference in level should be used.

Template



Source Image

Euclidean Distance

Let I be a gray level image
and g be a gray-value template of size $n \times m$.

$$d(I,g,r,c) = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(I(r+i,c+j) - g(i,j)\right)^2}$$

In this formula (r,c) denotes the top left corner of template g.

Correlation

- ***Correlation*** **is a measure of the degree to which two variables agree, not necessary in actual value but in general behavior.**
- **The two variables are the corresponding pixel values in two images, template and source.**

Grey-Level Correlation Formula

$$cor = \frac{\sum_{i=0}^{N-1} (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sqrt{\sum_{i=0}^{N-1} (x_i - \bar{x})^2 \cdot \sum_{i=0}^{N-1} (y_i - \bar{y})^2}}$$

x is the template gray level image

$\bar{x}$ is the average grey level in the template image

y is the source image section

$\bar{y}$ is the average grey level in the source image

N is the number of pixels in the section image

(N= template image size = columns * rows)

The value cor is between –1 and +1,
with larger values representing a stronger relationship between the two images.

Correlation is Computation Intensive

- Template image size: 53 x 48
- Source image size: 177 x 236
- Assumption: template image is inside the source image.
- Correlation (search) matrix size: 124 x 188 (177-53 x 236-48)

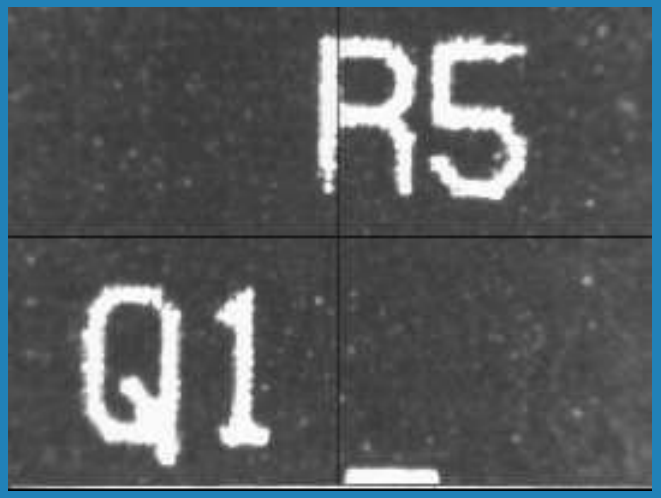
- Computation count

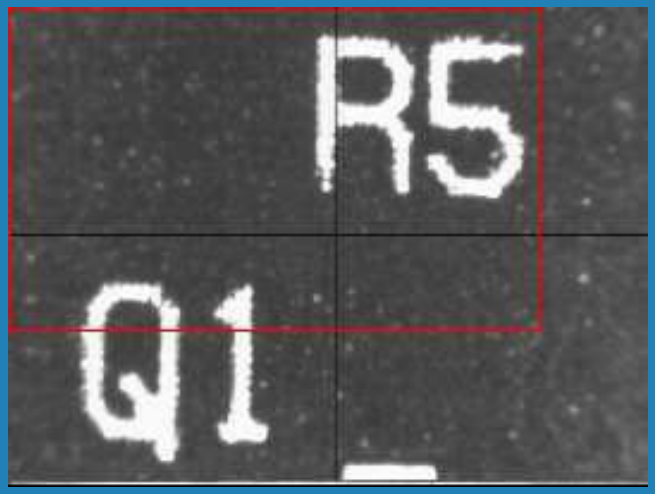
  124 x 188 x 53 x 48 = 59,305,728

Machine Vision Example

- Load printed circuit board into a machine
- Teach template image (select and store)
- Load printed circuit board
- Capture a source image and find template

Machine Vision Example





Assumptions and Limitations

1. Template is entirely located in source image

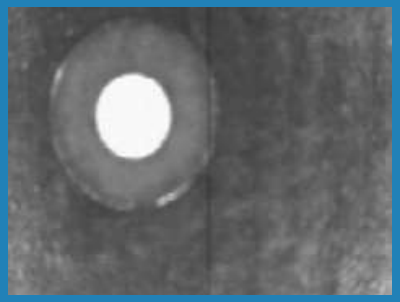
2. Partial template matching was not performed (at boundaries, within image)

3. Rotation and scaling will cause poor matches

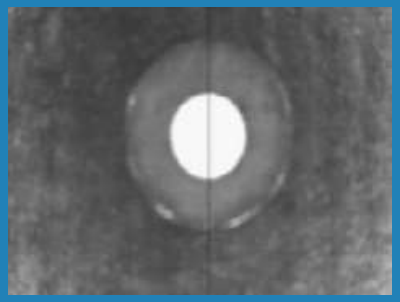
Matlab Example

Matlab Data Set

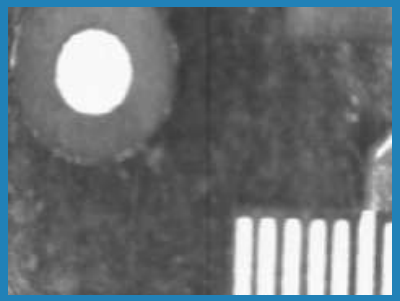
Template



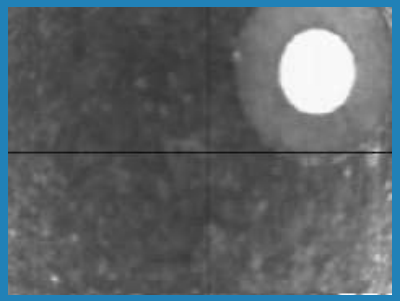
Data Set 1



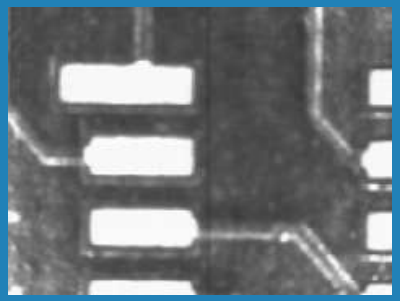
Data Set 2



Data Set 3

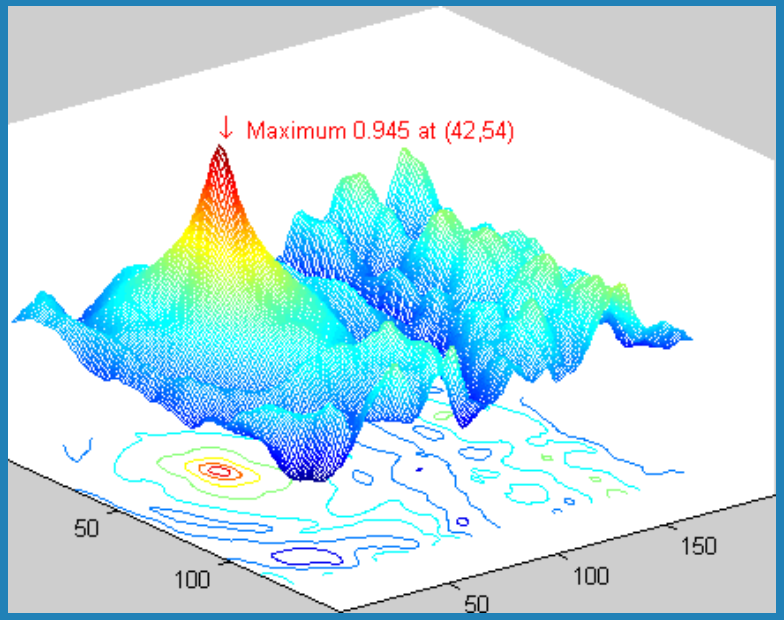


Data Set 4



Data Set 5

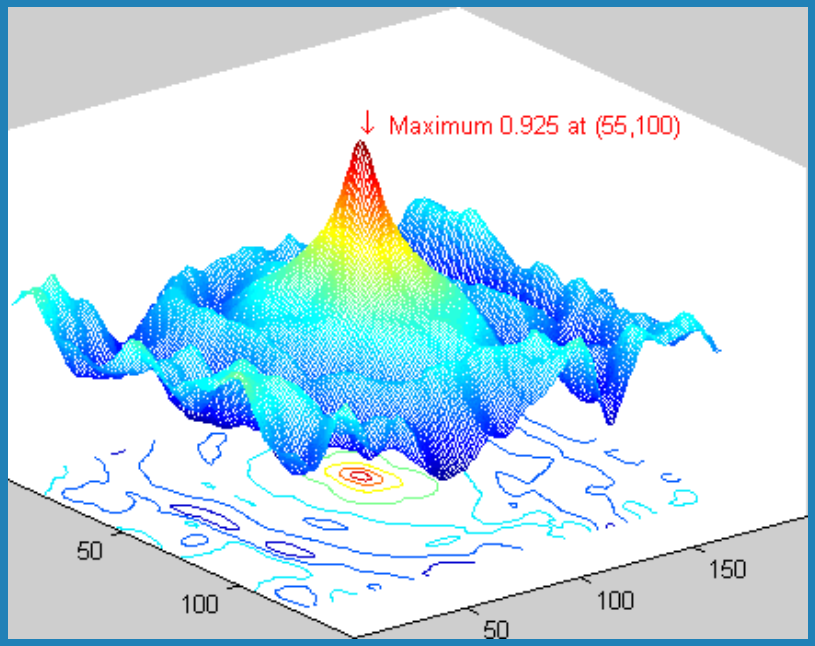
Data Set 1



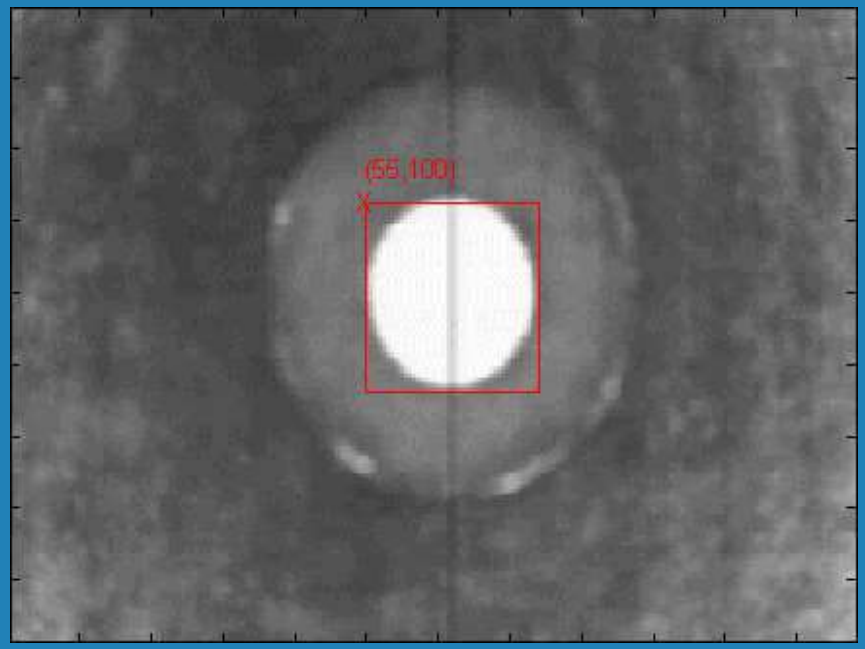
Correlation Map with Peak

Source Image, Found Rectangle, and Correlation Map

Data Set 2

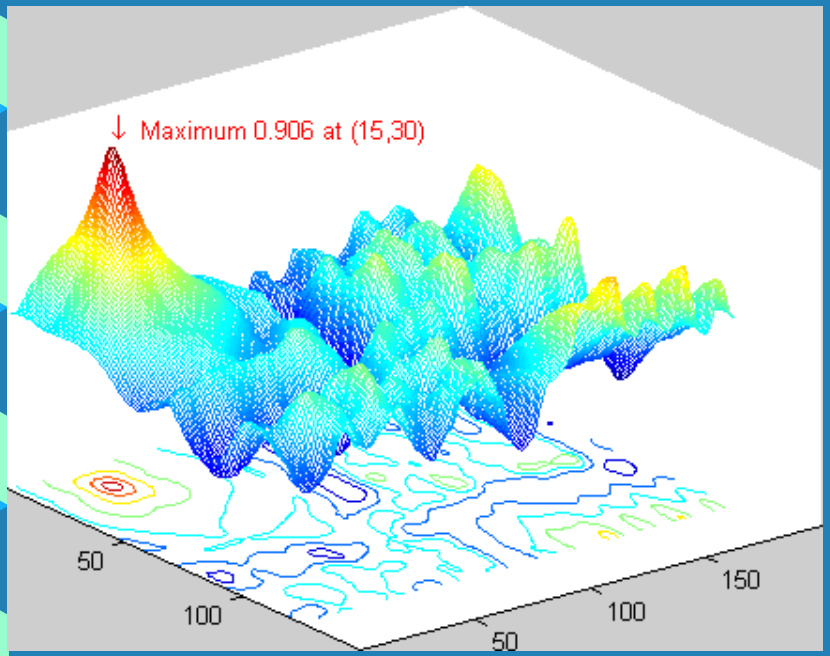


Correlation Map with Peak

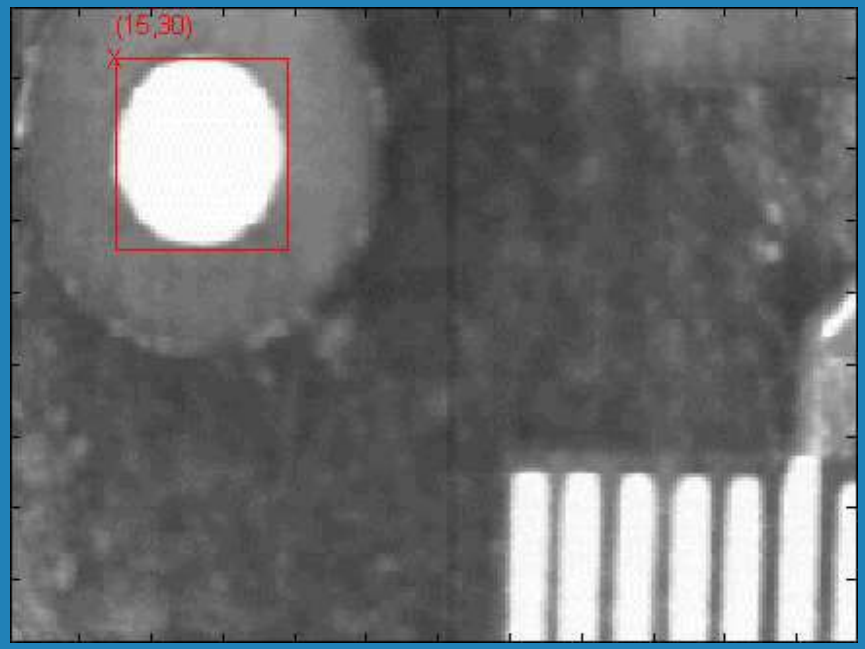


Source Image and Found Rectangle

Data Set 3

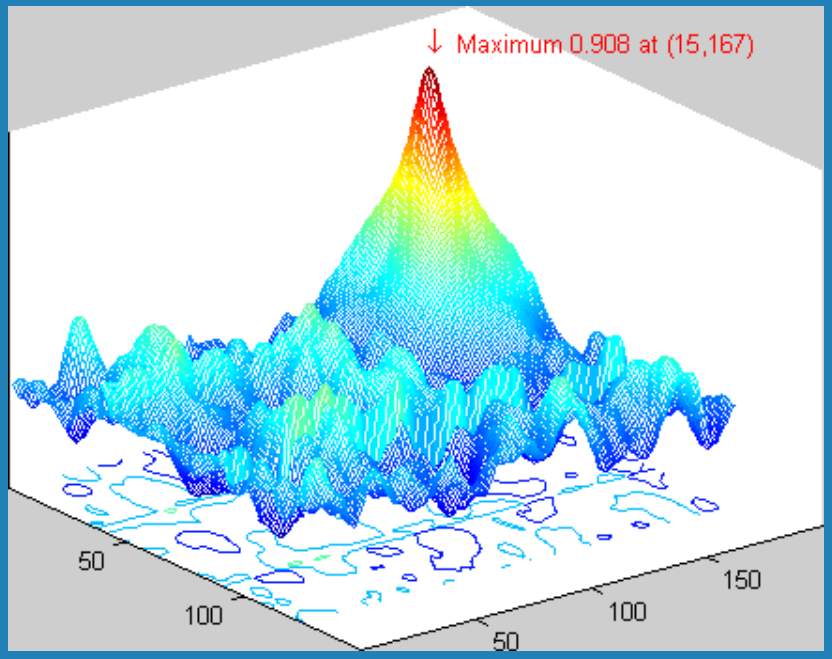


Correlation Map with Peak

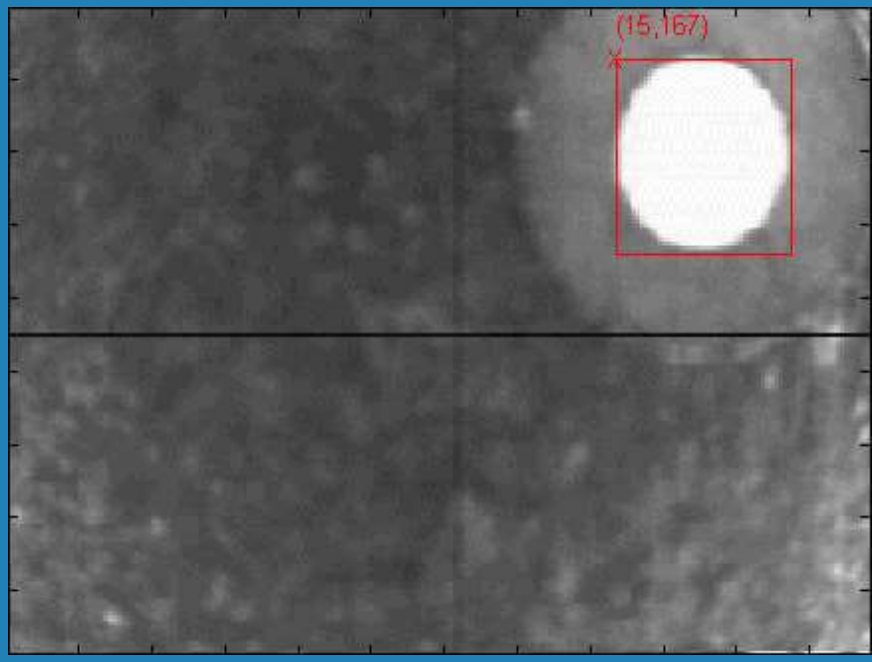


Source Image and Found Rectangle

Data Set 4

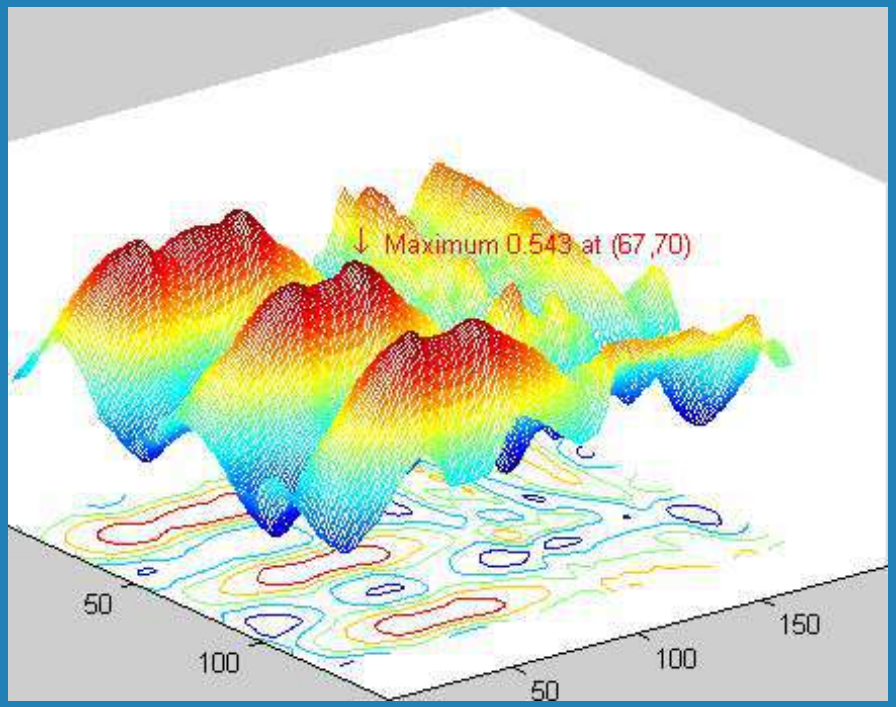


Correlation Map with Peak

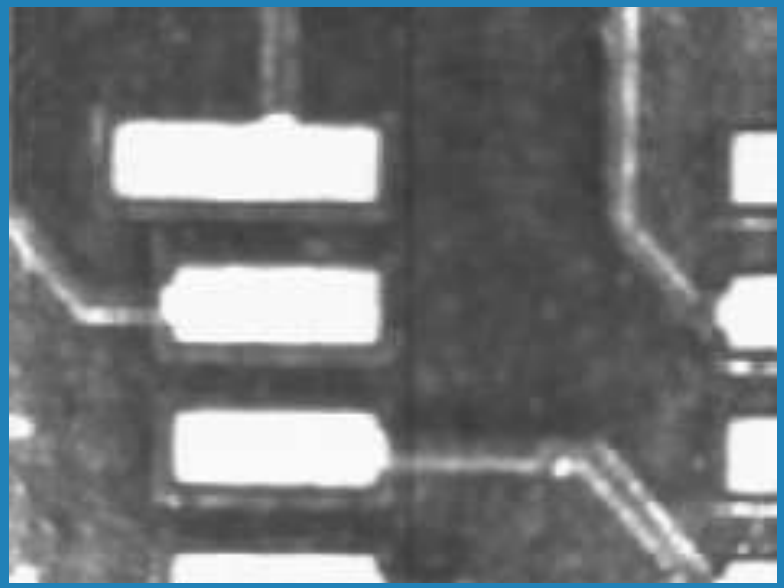


Source Image and Found Rectangle

Data Set 5, Corr. Map

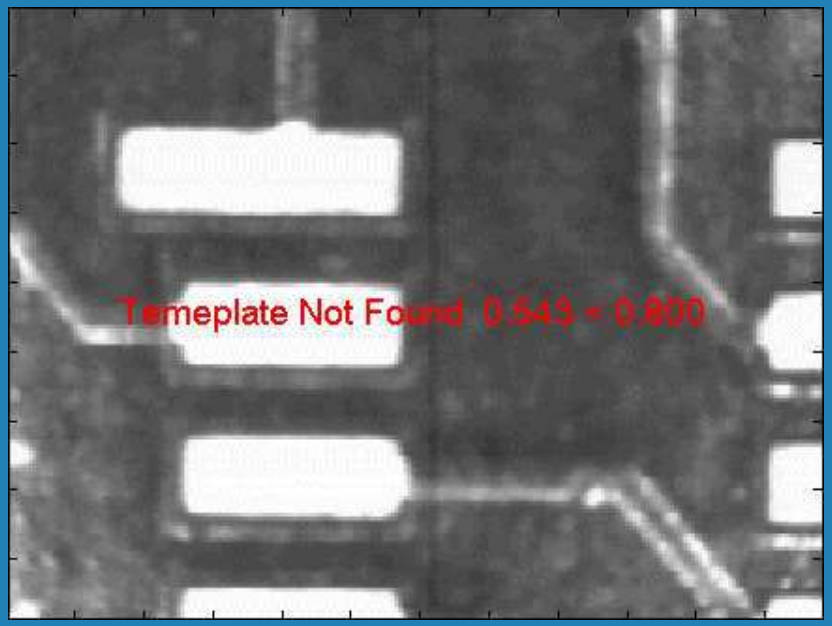


Correlation Map with Peak

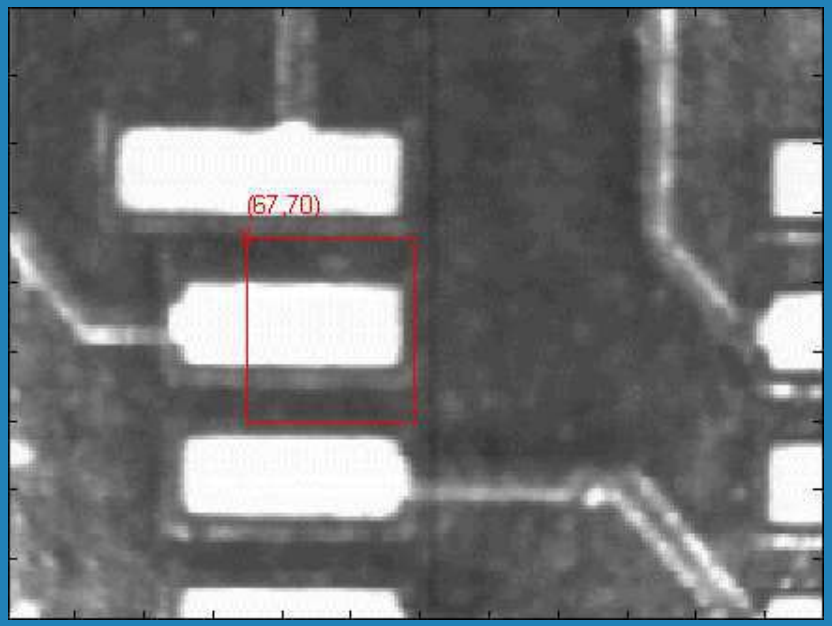


Source Image

Data Set 5, Results



Threshold set to 0.800



Threshold set to 0.200

Thank you for your attention