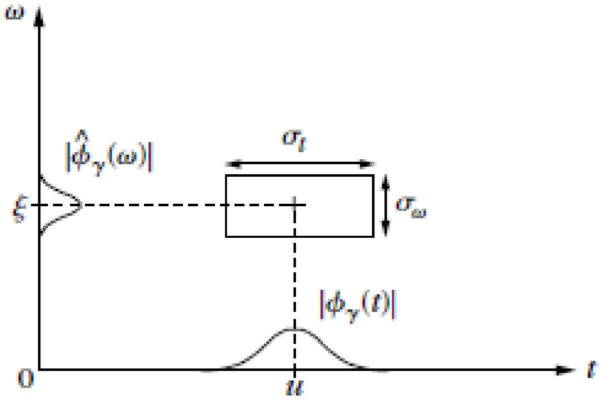
Image transforms -Wavelet transform

Joint localisation of a signal

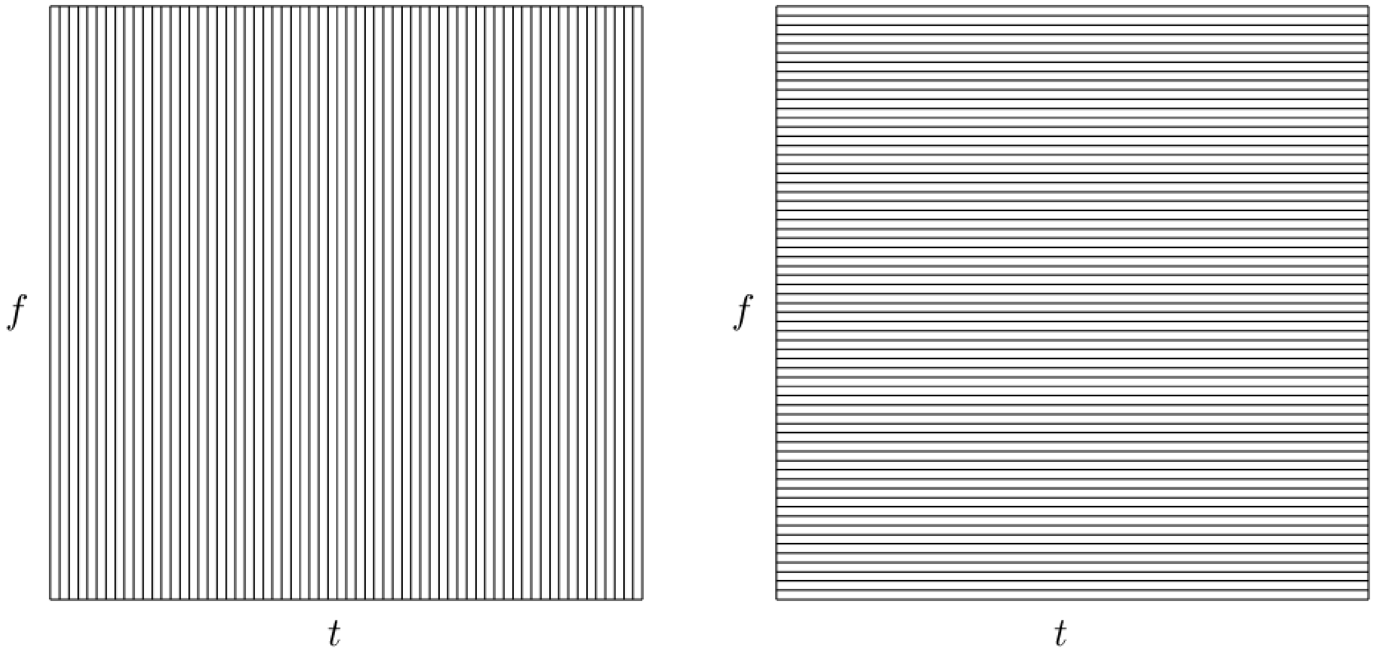
- T: $x(t) \rightarrow X(\omega)$
- Does T provide good time-frequency *localisation* ?
  - can one do instantaneous frequency analysis of a signal?
- $\{x(t), X(\omega)\} \rightarrow$ a point in the time-frequency plane

- Heisenberg's uncertainty principle puts a lower bound on the localisation *accuracy*
  - I.e. area of a time-frequency tile: $\sigma_t^2 \sigma_\omega^2 \geq 1/4$



How good is DFT for joint localisation?

- Basis functions: sinusoids (infinite support)

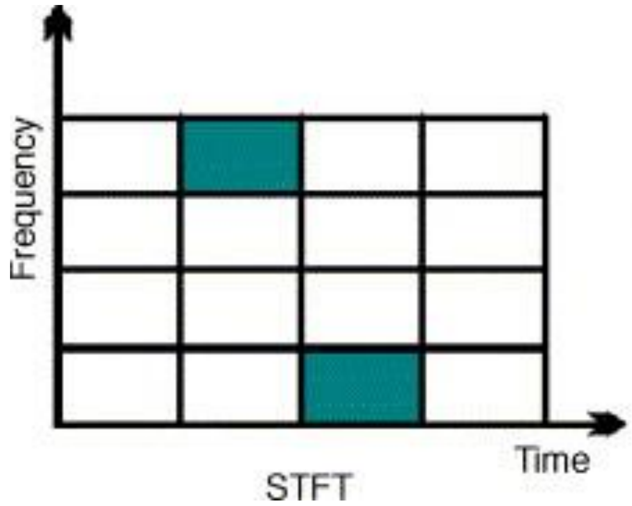


(a) (b)

- Very bad joint localisation!
- DFT can only localise in time **or** frequency dimension
- Assumes all frequency components exist all the time

Short Time Fourier Transform (STFT)

- Divide the time axis into smaller intervals and do DFT over them
  - Equivalent to using a moving window
- Results in uniform tiling of the t-f plane
- Definite improvement over DFT

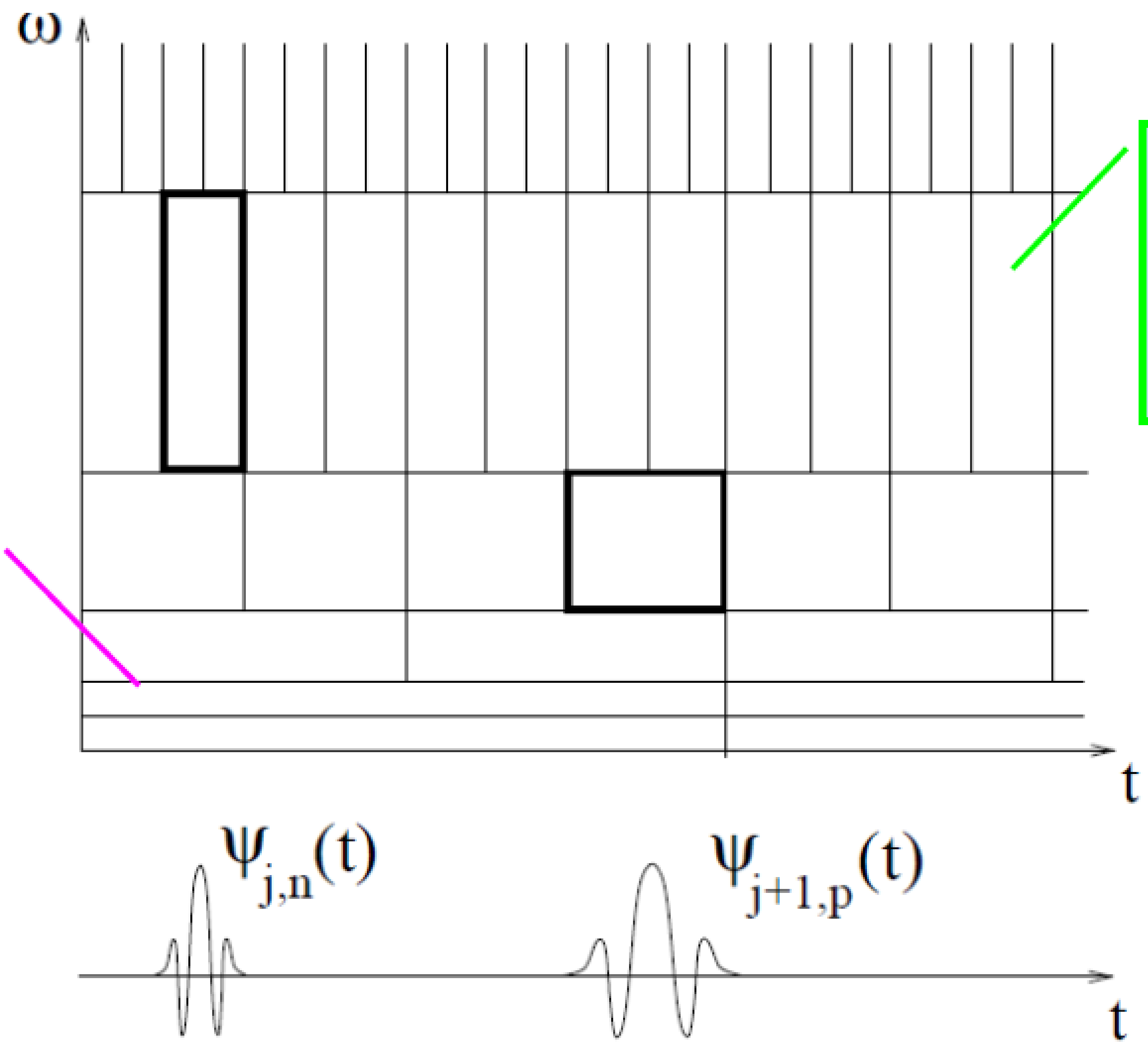


- Window width controls resolution
- Tradeoff between time and frequency resolution
  - Small window – poor frequency resolution and vice versa

Wavelet Transform – key points

- Basis functions are short duration waves or wavelets
  - Aids localisation
- Uses the concept of scale instead of frequency
- A non-uniform tiling approach to cover the t-f plane
- Aids in a multiresolution analysis of signals

Time-frequency tiles in Wavelet Transform



Better time resolution; Poor frequency resolution

Better frequency resolution; Poor time resolution

Wavelets - perspective from signal decomposition

$$f(x) = \sum_k a_k \phi_k \qquad a_k = \langle \phi_k, f \rangle$$

f is decomposed/expanded using basis functions ϕ_k

DFT: $\phi_k(x) = e^{jxk}$

- ϕ_k is an orthonormal set $\langle \phi_k, \phi_{k'} \rangle = \delta(k-k')$
- e^{jxk} have **infinite** support on x-axis (waves)

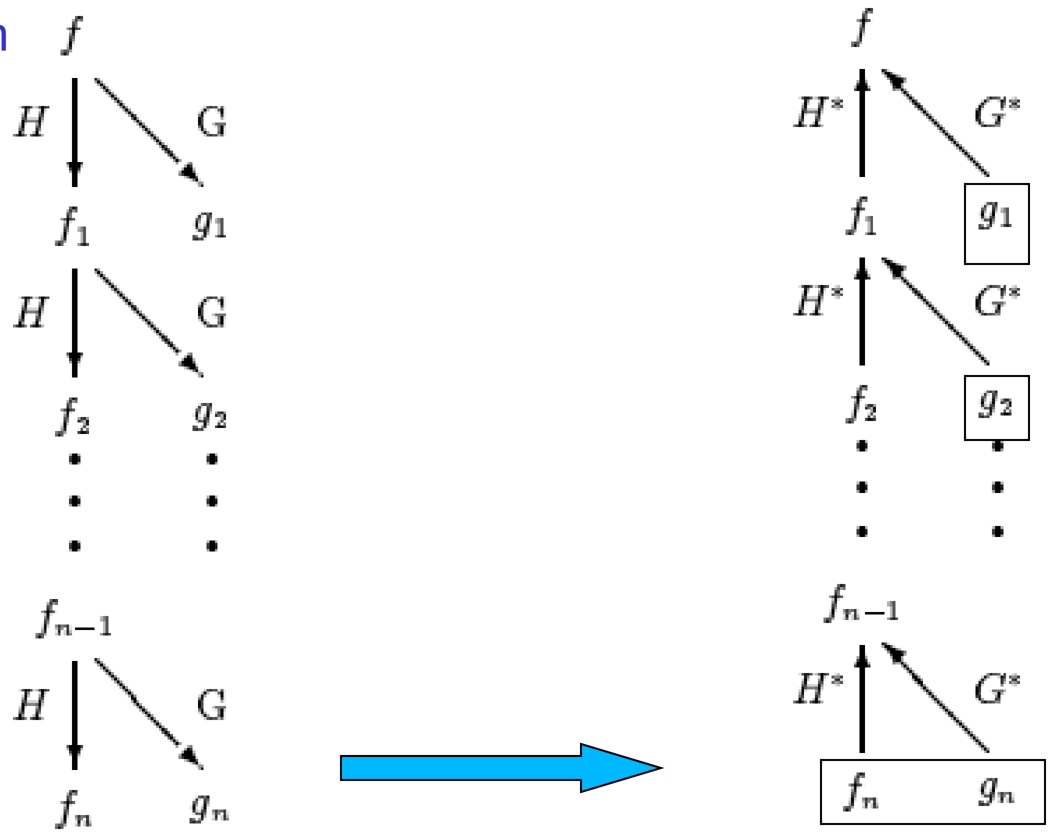
Wavelet transform: $\phi_k(x)$ are wavelets with finite support

How is this decomposition done?

Decomposing f: f_n is an approximation of f_{n-1}
g_n is the leftover detail in f_{n-1}

Highest resolution

Lowest resolution



Reconstruct f

By starting from low resolution and adding details

Towards wavelets - Haar transform

Haar transform decomposes a signal into 2 subsignals:

1. Pairwise average (trend)
2. Pairwise difference (fluctuation)

$$\boxed{f = (f_1, f_2, .. f_N) \mapsto (a^1 \mid d^1)}$$

For energy conservation

$$a^1 = (a_1, a_2 .. a_m .. a_{N/2}); \quad a_m = \sqrt{2}\,\frac{f_{2m-1} + f_{2m}}{2}$$

$$d^1 = (d_1, d_2 .. d_m .. d_{N/2}); \quad d_m = \sqrt{2}\,\frac{f_{2m-1} - f_{2m}}{2}$$

This decomposition can be done iteratively!

1-level Harr decomposition example

decompose

$$f = (4,6,10,12,8,6,5,5)$$

$$\Big\downarrow \mathcal{H}$$

$$(a^1 \mid d^1)$$

$$a^1 = \sqrt{2}(5,11,7,5) \qquad d^1 = \sqrt{2}(-1,-1,1,0)$$

<u>*Note*</u>*:* d_m is significantly smaller than f_m

Energy in a^1 is >> energy in d^1 (energy compaction)

---

reconstruct

$$(a^1 \mid d^1) \rightarrow f = \left(\frac{a_1 + d_1}{\sqrt{2}}, \ldots \frac{a_m + d_m}{\sqrt{2}}, \ldots \frac{a_{N/2} + d_{N/2}}{\sqrt{2}}\right)$$

3-level Example

$$f = (4,6,10,12,8,6,5,5)$$

Pairwise sum and divide by √2

Pairwise difference and divide by √2

$$a^1 = \sqrt{2}(5,11,7,5)$$

$$d^1 = \sqrt{2}(-1,-1,1,0)$$

$$a^2 = (16,12)$$

$$d^2 = (-6,2)$$

$$a^3 = (\frac{28}{\sqrt{2}}) = (14\sqrt{2})$$

$$d^3 = (\frac{4}{\sqrt{2}}) = (2\sqrt{2})$$

$$f = (4,6,10,12,8,6,5,5) \mapsto (\boxed{14\sqrt{2}} \mid \boxed{(2\sqrt{2}) \mid (-6,2) \mid \sqrt{2}(-1,-1,1,0)})$$

Final Haar transform

Harr decomposition

- A hierarchical decomposition

- At each level, the signal is decomposed into an approximate and detail component

➢ A signal is expressed as a sum of a low resolution (approximation) + many detail signals

- There is a strict relationship between approximation signals at a level *k* and that at a lower level (*k-1*):

$$A^k = A^{k-1} - D^k$$

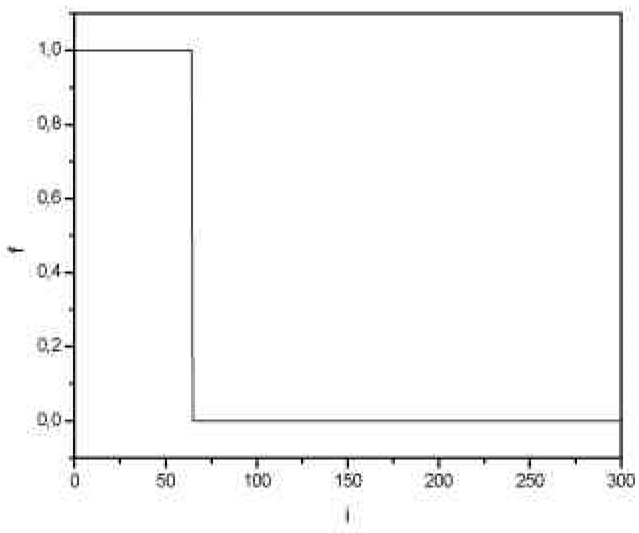
Decomposition as projections

- The k^{th} level approximate signal is generated by projecting the signal onto a set of shifted scaling functions $< f, v_j^k >$
  - ➢ The scaling function gets wider as we move up the levels
    - More smoothing across scales

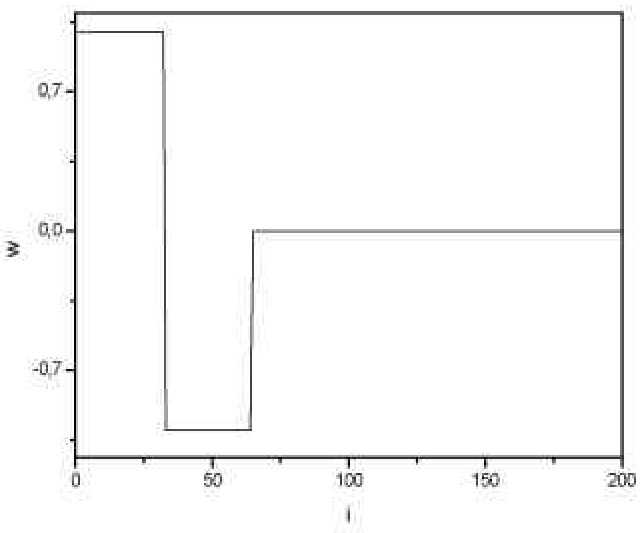
- The k^{th} level detail signal is generated by projecting the signal onto a set of shifted wavelet functions $< f, w_j^k >$
  - ➢ The wavelet function gets wider as we move up the levels

Haar functions

(Continuous case)



scaling function



wavelet function

Haar scaling functions at level 1

$$v_1^1 = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0,0,0,0...)$$

$$v_2^1 = (0,0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0,0,0,0...)$$

$$v_3^1 = (0,0,0,0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0,0,0,0...)$$

$$v_{N/2}^1 = (0,0,0,.....0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$$

Scaling coeff. at scale 1

$$a_1 = \; < f, v_1^1 >$$

$$a_2 = \; < f, v_2^1 >$$

$$a_3 = \; < f, v_3^1 >$$

$$a_{N/2} = \; < f, v_{N/2}^1 >$$

Support of *2*

$v^1_j(n) = v^1_{j-1}(n-2); j = 2,3..N/2$

Average value of $v^1_j(n)$ is $1/\sqrt{2}$

first level approximation subsignal

$$a^1 = (a_1, a_2..a_m..a_{N/2})$$

Haar wavelet

wavelets at scale 1

$$w_1^1 = (\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0,0,0,0\ldots)$$

$$w_2^1 = (0,0, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0,0,0,0\ldots)$$

$$w_3^1 = (0,0,0,0, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0,0,0,0\ldots)$$

$$w_{N/2}^1 = (0,0,0,\ldots..0, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}})$$

wavelet coeff. at scale 1

$$d_1 = \; < f, w_1^1 >$$

$$d_2 = \; < f, w_2^1 >$$

$$d_3 = \; < f, w_3^1 >$$

$$d_{N/2} = \; < f, w_{N/2}^1 >$$

Note: $w_j(n) = w_{j-1}(n-2); \; j = 2,3\ldots N/2)$

Average value of any $w_j = 0$

first level detail subsignal

$$d^1 = (d_1, d_2 .. d_m .. d_{N/2})$$

Scaling and wavelet functions at 2^{nd} level

$$a^2 = (< f, v_1^2 >, < f, v_2^2 >, \ldots., < f, v_{N/4}^2 >)$$

$$v_1^2 = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0,0,0 \ldots)$$

Support of *4*

$v^2_j(n) = v^2_{j-1}(n-4); j = 2,3..N/4$

Average value of each $v^2_j(n) = 1/2$

$$d^2 = (< f, w_1^2 >, < f, w_2^2 >, \ldots., < f, w_{N/4}^2 >)$$

$$w_1^2 = (\frac{1}{2}, \frac{1}{2}, \frac{-1}{2}, \frac{-1}{2}, 0,0,0,0 \ldots)$$

Signal synthesis

$$f = A^1 + D^1 = \sum_j a_j^1 v_j^1 + \sum_j d_j^1 w_j^1$$

$$A^1 = [< f, v_1^1 > v_1^1 + < f, v_2^1 > v_2^1 \ldots\ldots < f, v_{N/2}^1 > v_{N/2}^1] = \sum_{j=1}^{N/2} < f, v_j^1 > v_j^1$$

First approximation signal

$$D^1 = [< f, w_1^1 > w_1^1 + < f, w_2^1 > w_2^1 \ldots\ldots < f, w_{N/2}^1 > w_{N/2}^1] = \sum_{j=1}^{N/2} < f, w_j^1 > w_j^1$$

First detail signal

In the example: $f = (4,6,10,12,8,6,5,5)$

$$a^1 = \sqrt{2}(5,11,7,5) \quad \Rightarrow \quad A^1 = (5,5,11,11,7,7,5,5)$$

$$d^1 = \sqrt{2}(-1,-1,1,0) \quad \Rightarrow \quad D^1 = (-1,1,-1,1,1,-1,0,0)$$

$$A^1 + D^1 = (5,5,11,11,7,7,5,5) + (-1,1,-1,1,1,-1,0,0) = (4,6,10,12,8,6,5,5) = f$$

low resolution part

detail part

Scaling and wavelet coeff. at kth level

If f is N-length and $N = 2^k$, a k-level decomposition is possible

$$a^k = (< f, v_1^k >, < f, v_2^k >, \ldots., < f, v_{N/2^k}^k >)$$

Scaling coeff.

$$d^k = (< f, w_1^2 >, < f, w_2^2 >, \ldots., < f, w_{N/2^k}^2 >)$$

Wavelet coeff.

Signal synthesis with multilevel

2-levels $f = A^2 + D^2 + D^1 = A^1 + D^1$

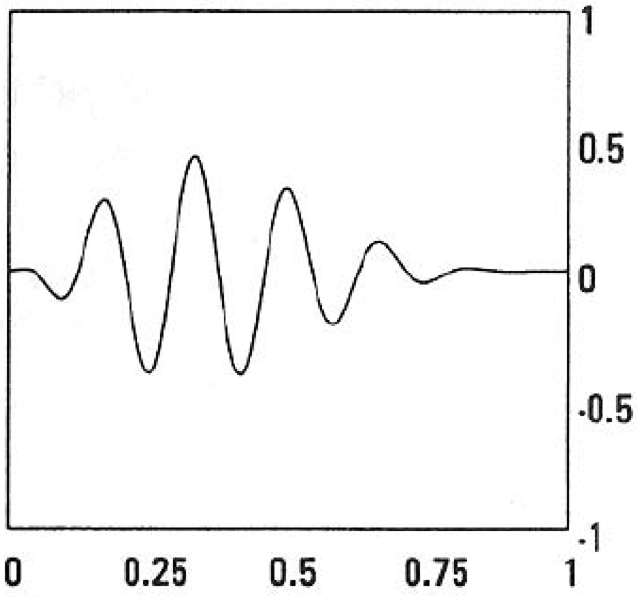
k-levels; *k* is divisible by 2

$$f = A^k + D^k + D^{k-1} + D^{k-2}.....D^1$$

$$A^{k-1} = A^k + D^k$$

Example 2

Signal f



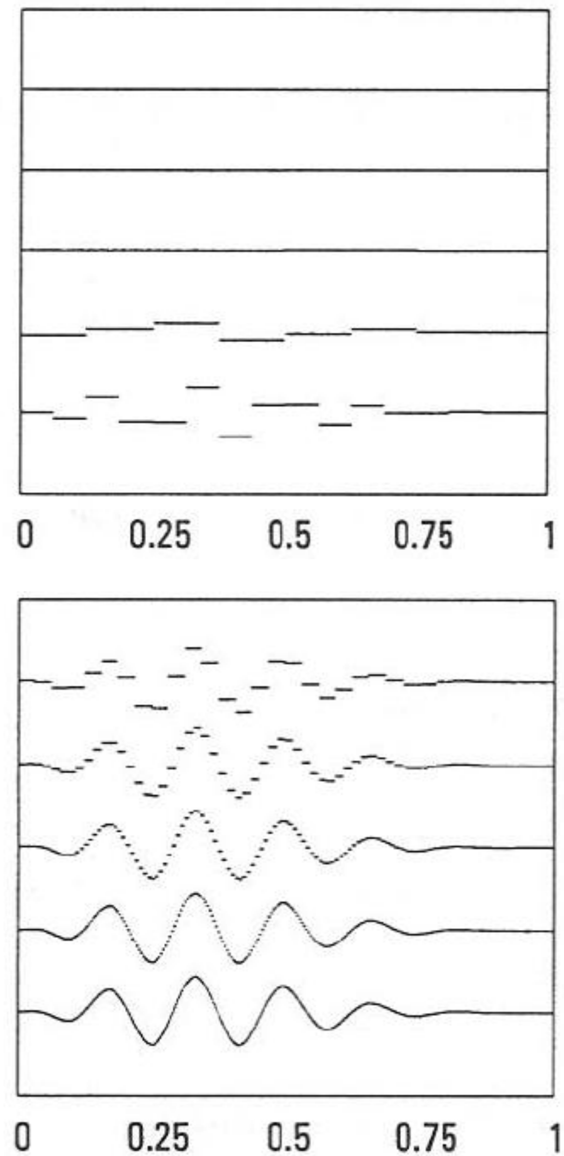
10 approximation signals

$k = 10$

$k = 6$

$k = 5$

$k = 1$



Generalisation

The Haar wavelets $\psi_{k,j}$ can be generalised as follows:

$$\psi_{k,j}(x) = 2^{\frac{-k}{2}}\psi(2^{-k}x - j); \quad j,k \in Z$$

$\psi \in L^2(R)$; $\{\psi_{k,j}\}$ are an orthonormal basis for $L^2(R)$

Multi-Resolution analysis (MRA) is an extension of this concept:

A family of subspaces of $L^2(R)$

Daubechies wavelets

Daub4 scaling function

- Daub4 differs from Haar in the choice of scaling and wavelet functions

Weighted averaging

$$v_1^1 = (\boxed{\alpha_1, \alpha_2, \alpha_3, \alpha_4}, 0,0,0,0\ldots 0)$$

$$v_2^1 = (0,0, \boxed{\alpha_1, \alpha_2, \alpha_3, \alpha_4}, 0,0,\ldots 0)$$

$$\vdots$$

$$v_{\frac{N}{2}-1}^1 = (0,0,0,0.\boxed{..\alpha_1, \alpha_2, \alpha_3, \alpha_4})$$

$$v_{N/2}^1 = (\boxed{\alpha_3, \alpha_4}, 0,0,0,0\ldots \boxed{\alpha_1, \alpha_2})$$

Weights

$$\alpha_1 = \frac{1+\sqrt{3}}{4\sqrt{2}} \quad \alpha_2 = \frac{3+\sqrt{3}}{4\sqrt{2}}$$

$$\alpha_3 = \frac{3-\sqrt{3}}{4\sqrt{2}} \quad \alpha_4 = \frac{1-\sqrt{3}}{4\sqrt{2}}$$

$$\sum_i \alpha_i = \sqrt{2} \quad \sum_i \alpha^2{}_i = 1$$

- Support of 4
- Each v_j is obtained by shifting v_{j-1} by *2* with wrap around
- Energy of each v_j = *1*

Daub4 wavelet function

$$w_1^1 = (\beta_1, \beta_2, \beta_3, \beta_4, 0,0,0,0 \ldots 0)$$
$$w_2^1 = (0,0, \beta_1, \beta_2, \beta_3, \beta_4, 0,0, \ldots 0)$$
$$\vdots$$
$$w_{\frac{N}{2}-1}^1 = (0,0,0,0 \ldots \beta_1, \beta_2, \beta_3, \beta_4)$$
$$w_{N/2}^1 = (\beta_3, \beta_4, 0,0,0,0 \ldots \beta_1, \beta_2)$$

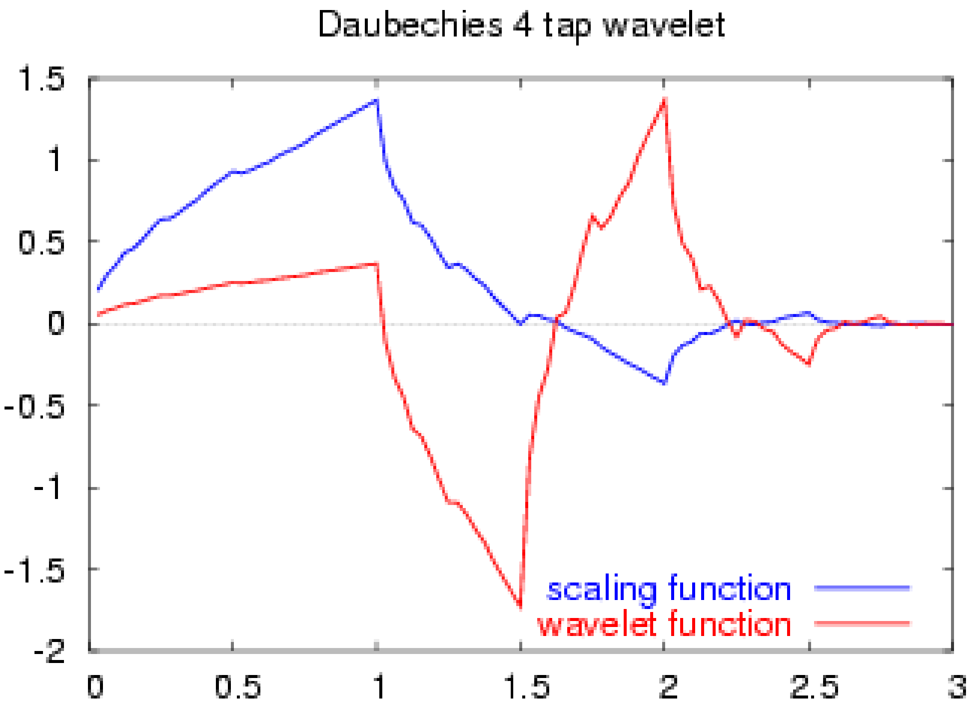
$$\alpha_1 = \frac{1+\sqrt{3}}{4\sqrt{2}} \quad \alpha_2 = \frac{3+\sqrt{3}}{4\sqrt{2}}$$
$$\alpha_3 = \frac{3-\sqrt{3}}{4\sqrt{2}} \quad \alpha_4 = \frac{1-\sqrt{3}}{4\sqrt{2}}$$

$$\beta_1 = \alpha_4 = \frac{1-\sqrt{3}}{4\sqrt{2}} \quad \beta_2 = -\alpha_3 = \frac{\sqrt{3}-3}{4\sqrt{2}}$$
$$\beta_3 = \alpha_2 = \frac{3+\sqrt{3}}{4\sqrt{2}} \quad \beta_4 = -\alpha_1 \frac{-1-\sqrt{3}}{4\sqrt{2}}$$

- Support of 4
- Each w_j is obtained by shifting v_{j-1} by *2* with wrap around
- Energy of each $w_j = 0$

$$\sum_i \beta_i = 0$$
$$0\beta_1 + 1\beta_2 + 2\beta_3 + 3\beta_4 = 0$$

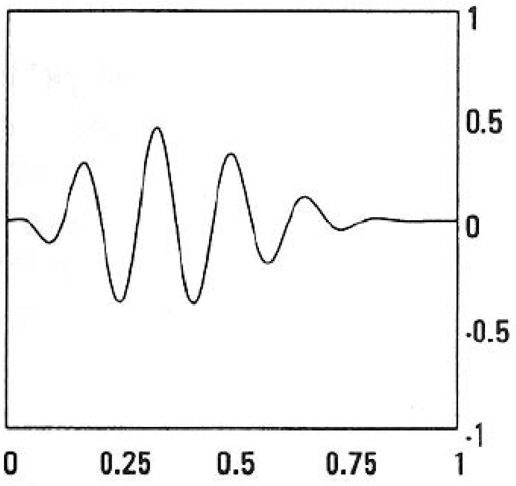
Daub4 functions



Revisit example 2

10–level approximations

Signal *f*



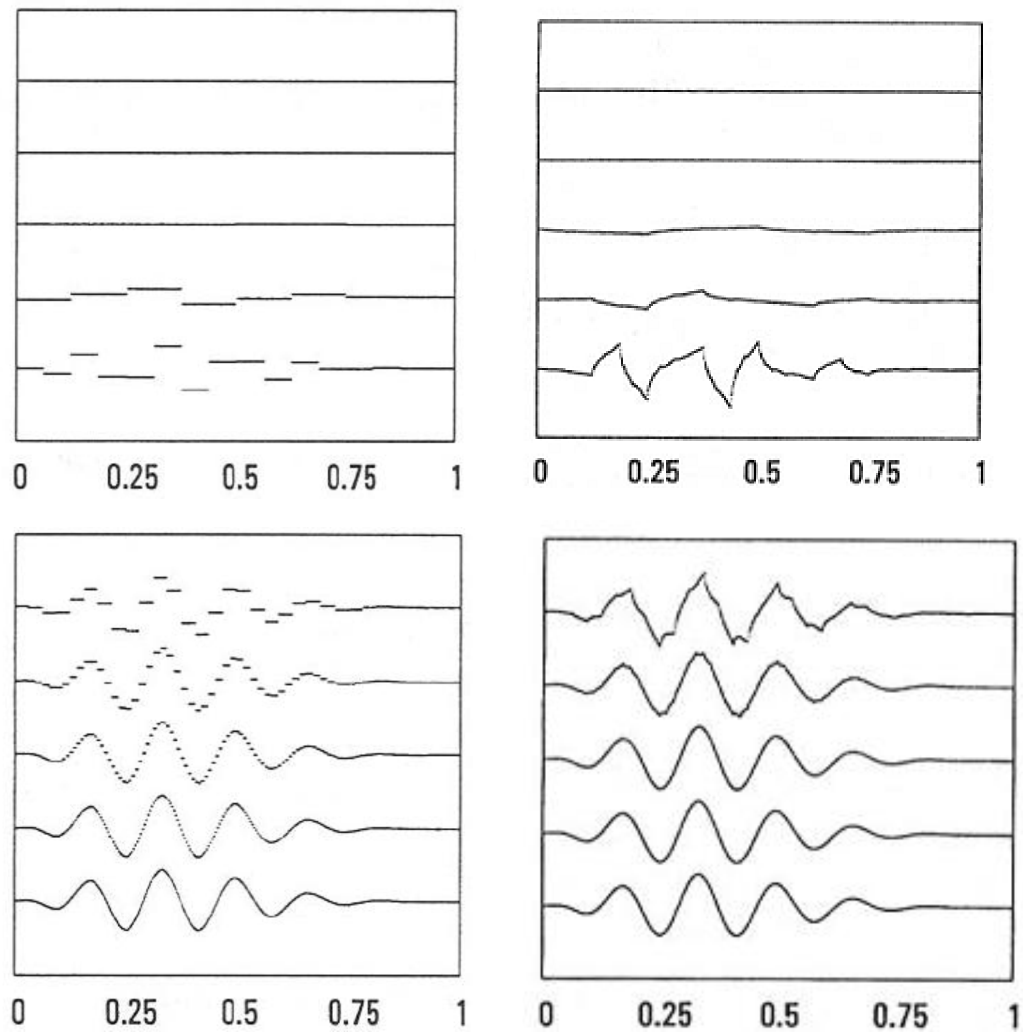
Haar **Daub4**

k =*10*

k =*6*

k =*5*

k =*1*



Properties of Daub4

1. The scaling and wavelet functions satisfy

$$< v_n^k, v_m^k > = \delta(n-m)$$

$$< w_n^k, w_m^k > = \delta(n-m)$$

$$< v_n^k, w_m^k > = 0 \quad \forall n, m$$

- The scaling functions form an *orthonormal* basis and their span is a vector space **V**
- The wavelet functions form an *orthonormal* basis and their span is a vector space **W**
- **V** $\perp$ **W**

Properties of Daub4 contd.

$$w_m^1 = \beta_1 v_{2m-1}^0 + \beta_2 v_{2m}^0 + \beta_3 v_{2m+1}^0 + \beta_4 v_{2m+2}^0$$

and

$$w_m^k = \beta_1 v_{2m-1}^{k-1} + \beta_2 v_{2m}^{k-1} + \beta_3 v_{2m+1}^{k-1} + \beta_4 v_{2m+2}^{k-1}$$

1. Daub4 wavelet at level k is related to (and so is derivable from) the scaling function at the previous level

$$\sum_i \beta_i = 0 \quad \Longrightarrow \quad d_m = < f . w_m^k > = 0 \quad \forall m$$

$$0\beta_1 + 1\beta_2 + 2\beta_3 + 3\beta_4 = 0$$

2. any signal which is <u>linear</u> <u>over the support</u> of k-level Daub4 wavelet will have (~) zero detail coeff.

So far

Wavelet transform decomposes signal into approximation and detail signals using finite support functions

$$f(x) \rightarrow \{A^k, D^k\}$$

- Need a Scaling function to generate the approximations
- Need a Wavelet function to generate the details

- D^k represents the 'degree of similarity' (cross correlation) between the wavelet at scale k with the function f

Signal synthesis with wavelet transform

Given a signal *f(x)*

$$f = A^k + D^k + D^{k-1} + D^{k-2} \ldots\ldots D^1$$

$$A^{k-1} = A^k + D^k$$

A^k: approximation at k^{th} level or scale

D^k: detail at k^{th} level scale

$$A^k = \sum_j a_j^k v_j^k; \quad D^k = \sum_j d_j^k w_j^k$$

$$a_j^k = \langle f, v_j^k \rangle; \quad d_j^k = \langle f, w_j^k \rangle$$

v: scaling function

w: wavelet function

Multi-Resolution Analysis Mallat 1989

- A mathematical framework for representing or analysing a given function at several resolution (levels of detail)

- Uses a set of nested subspaces to represent a signal in a vector space

$$\{0\}\ldots\ldots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \ldots\ldots \subset L^2(R)$$

MRA in a nutshell

Decompose a function space → subspaces

→ if $f(x) \in$ function space then a piece (projection) of $f(x)$ is in each subspace

→ each piece is a finer and finer detail of f

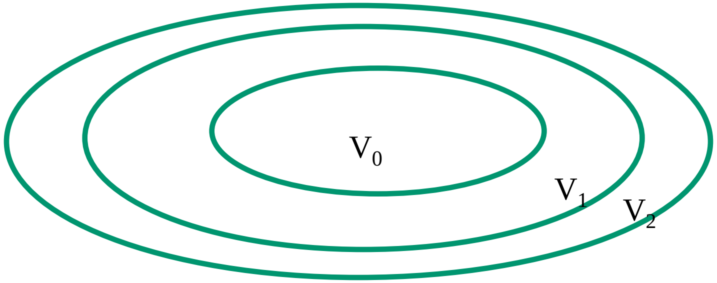
Two subspaces (further decomposed) of interest:

V scaling (approximation) and **W** detail

Key condition for MRA: If $f(x) \in V_j$ then its translates and dilations are in V_{j-1}

MRA - definition

- $L^2(R)$: Linear space of finite energy signals; $f(x)$: a signal
- MRA is a decomposition of $L^2(R)$ into a series of nested subspaces V_m, (m is an integer) such that



$$\{0\}\ldots\ldots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \ldots\ldots \subset L^2(R)$$

$$V_m \subset V_{m+1}$$

$$f(x) \in V_m \Leftrightarrow f(2x) \in V_{m+1}$$ Scale invariance

$$\cap_m V_m = \{0\}$$ All scale subspaces intersect only at the zero vector

$$\vee_m V_m = L^2(R)$$ Any function in L^2 can be closely approximated by elements from V

$\vee$: Closure of $\cup V_m$

$$f(x) \in V_0 \Leftrightarrow f(x-j) \in V_0; \quad j \in Z$$ Shift invariance

$$\exists \varphi \in V_0 \quad such \quad that \quad \varphi(x-j); \quad j \in Z$$ *is an orthonormal basis for* V_0

Shift invariant basis

φ is the scaling (or father) function of the MRA

Warning

- Notations differ between engineers and mathematicians
- Positive scale indicates decreasing resoln for engineers and increasing resoln for mathematicians
  - $V_{-\infty} = L^2(R)$ for engineers
  - $V_{-\infty} = \{0\}$ for others

back

Implications of MRA

$$\varphi_{k,j}(x) = 2^{\frac{k}{2}}\varphi(2^k x - j) \quad spans \quad V_k$$

$$\varphi_{0,0}(x) \in V_0 \quad \Rightarrow \quad \varphi_{0,j} = \{\varphi(x-j)\} \quad spans \quad V_0$$

Translates of φ_0

$$\varphi_{1,j}(x) = \{\sqrt{2}\varphi(2x-j)\} \quad spans \quad V_1$$

$V_0 \subset V_1$

$\therefore \varphi_0(x) \in V_0$ can be written in terms of the basis of V_1 : $\varphi_{1,j}(x)$

$$\boxed{\varphi_{0,0}(x) = \sum_j c_j \varphi_{1,j}(x) = \sum_j c_j \sqrt{2}\varphi\ (2x-j); \quad j \in Z}$$

Dilation equation

The average value of $\varphi(x)$ is non zero: $\int \varphi(x)dx = a \neq 0$

$$\int_{-\infty}^{\infty} \varphi(x)dx = a = \int_{-\infty}^{\infty} \sum_j c_j \sqrt{2}\varphi(2x-j)dx = \sum_j c_j \frac{\sqrt{2}}{2}\cancelto{a}{\left[\int_{-\infty}^{\infty} \varphi(\tilde{x})d\tilde{x}\right]}$$

$$a = \sum_j c_j \frac{1}{\sqrt{2}} a \Rightarrow \boxed{\sum_j c_j = \sqrt{2}}$$

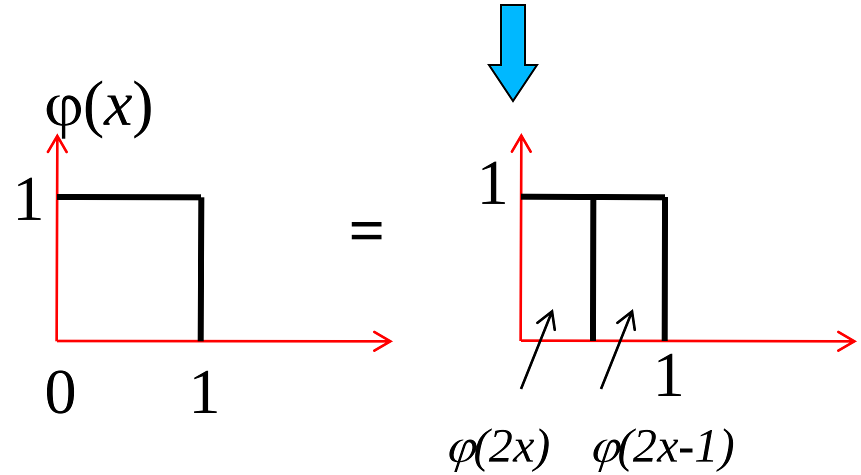
- There are many ways of choosing c_j to meet the above condition and the dilation equation
  - Each choice gives rise to a different scaling (and wavelet) function

Some solutions to dilation equation

Box function (~Haar)

$$\varphi(x) = \varphi(2x) + \varphi(2x-1)$$

$$\varphi(x) = 1 \quad 0 \le x < 1$$



$$\varphi(x) = 0 \quad otherwise$$

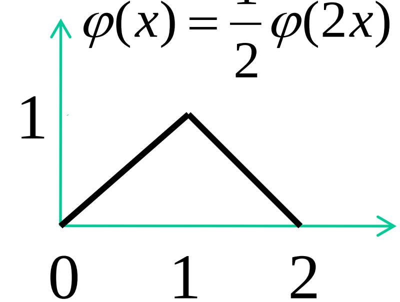
$$c_0 = 1 = c_1$$

Hat function

$$\varphi(x) = x \quad 0 \le x \le 1$$

$$\varphi(x) = \frac{1}{2}\varphi(2x) + \varphi(2x-1) + \frac{1}{2}\varphi(2x-2)$$

$$\varphi(x) = 2 - x; 1 \le x \le 2$$



$$\varphi(x) = 0 \quad otherwise$$

$$c_0 = \frac{1}{2} \quad c_1 = 1 \quad c_2 = \frac{1}{2}$$

Note: the normalising factor is ignored

Dilation eqn. - Solution by iteration

An iterative solution is more practical:

$$\varphi^{[l]}(x) = \sum_j c_j \varphi^{[l-1]}(2x - j)$$

$$l : iteration$$

Start with the box function and different coefficients c_j :

$c_0=c_3=1/4; c_1=c_2=3/4$ ➔ Quadratic spline function

$c_0 =c_4=1/8; c_1=c_3=1/2; c_2 =3/4$ ➔ B-spline

:

:

Daub*M* wavelets are also found this way

Deriving the Wavelet function

Let $W_m \subset V_{m+1}$, be an orthogonal complement of V_m

$$V_{m+1} = V_m \oplus W_m$$

and

$$L^2(R) = \bigoplus_{m=-\infty}^{\infty} W_m$$

$$\begin{tikzcd} & V_{m+1} \arrow[dl] \arrow[dr] & \\ V_m & & W_m \end{tikzcd}$$

Then basis for W_k are wavelets $\{\psi_{k,j}\}$

ψ can <u>also</u> be derived from the scaling function φ

$$\psi(x) = \sum_j b_j \varphi(2x - j)$$

How to find b_j?

Finding d_j

We know $W_0 \perp V_0$

$$\therefore < \psi(x), \varphi(x-j) > \; = 0$$ → Using

$$\psi(x) = \sum_j b_j \varphi(2x-j)$$

$$\varphi(x) = \sum_j c_j \varphi(2x-j)$$

After manipulations

$$\Rightarrow \sum_k \overline{b_k} c_{k-2j} = 0 \quad and \quad \sum_k \overline{b_k} b_{k-2j} = 2\delta(j)$$

A solution is

$$\boxed{b_j = (-1)^j c_{1-j}}$$

assuming real weights *c*, *d*

And the wavelet eqn is

$$\psi_{k,j}(x) = 2^{\frac{k}{2}} \psi(2^k x - j)$$

$\{\psi_{k,j}(x)\}$ spans W_k

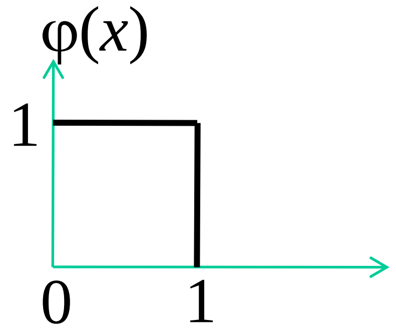
Harr

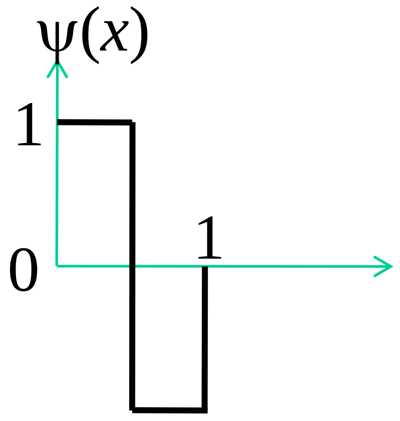
$$\psi(x) = \sum_j b_j \varphi(2x - j)$$

$$b_j = (-1)^j c_{1-j}$$

$$c_0 = c_1 = 1 \Rightarrow b_0 = 1; b_1 = -1$$

$$\psi(x) = \varphi(2x) + (-1)\varphi(2x - 1)$$





Discrete Wavelet Transform

- Given some $f(x) \in V_0$ we can expand it as

$$f(x) = \sum_j a_j \varphi_{0,j}(x) + \sum_{k=0} \sum_j d_{j,k} \psi_{j,k}(x)$$

$$\left. \begin{aligned} a_j &= \langle f(x), \varphi_j(x) \rangle \\ d_{j,k} &= \langle f(x), \psi_{j,k}(x) \rangle \end{aligned} \right\} \text{DWT coefficients}$$

Extension of DWT to images

To find 1-level DWT of an image of size *MxN*

- assume *M* and *N* are divisible by *2*

1. Perform 1D DWT on each row of $\boldsymbol{f}$ to get $\boldsymbol{f_r}$
2. Perform 1D DWT on each column of $\boldsymbol{f_r}$ to get the DWT{f}

Size of each subimage is *M/2 x N/2*

a: approximation; *h*: horizontal details

d: diagonal details; *v*: vertical details

$$f \mapsto \begin{bmatrix} a & v \\ h & d \end{bmatrix}$$

k=2

For *k*-level decomposition the subimage *a* will be subdivided *k*-1 times

$$f \mapsto \left[\begin{array}{cc|c} a^2 & v^2 & \multirow{2}{*}{v^1} \\ h^2 & d^2 & \\ \hline \multicolumn{2}{c|}{h^1} & d^1 \end{array}\right]$$

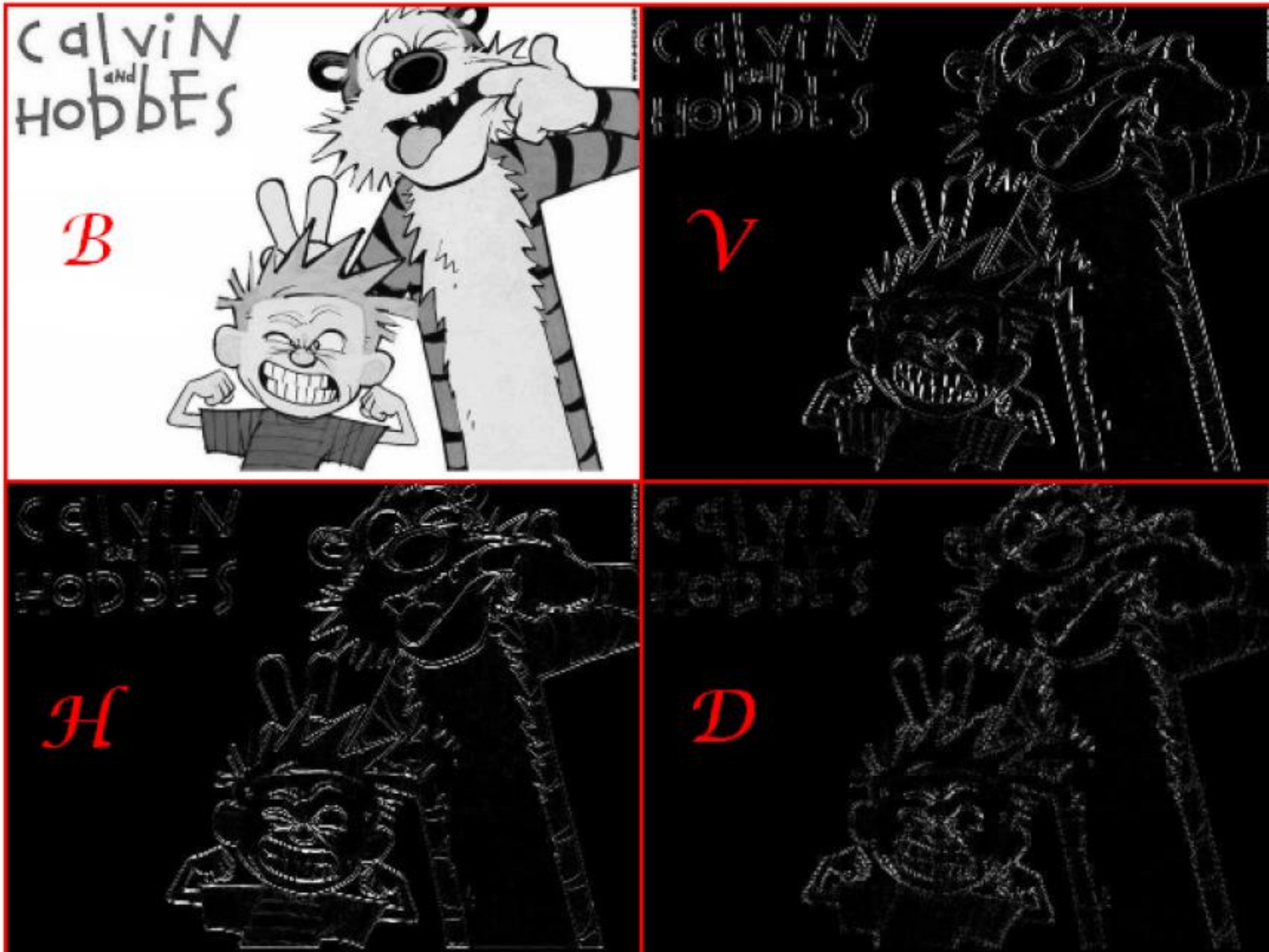


Image from C Benetau, UFL

Applications of DWT

- Denoising
  - Can also design wavelets adapted to noise
- Compression
  - Thresholding the detailed coefficients is a simple option
  - Ex. JPEG2000
- Analysis
  - Represent image/object with DWT coefficients
    - Ex. DWT features for texture classification
  - Segmentation