Filters in Image Processing

Image Transforms (II) – Wavelets

David Svoboda

email: svoboda@fi.muni.cz
Centre for Biomedical Image Analysis
Faculty of Informatics, Masaryk University, Brno, CZ

CBIA

October 11, 2019

Outline

Motivation

Fourier transform and its derivatives have nice properties

- reversible
- linear operator
- easy to construct (data independent)
- decorrelation property

Fourier transform has some drawbacks

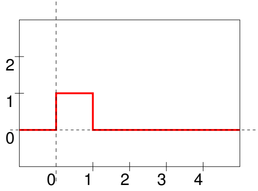
- localization is very expensive
- complexity cannot be lower than $O(n \log n)$

New basis

- Fourier basis originates from sin and cos functions.
- Let us introduce so called *scaling function* φ and *wavelet function* ϕ.
- φ and ψ should be strictly localized.
- A new basis should originate from φ and ψ.

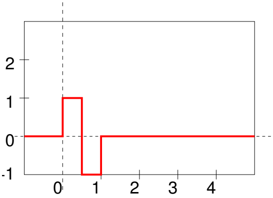
Scaling function φ

$$\varphi(x) = \begin{cases} 1 & \text{iff} \quad 0 \leq x < 1 \\ 0 & \text{otherwise} \end{cases}$$



Wavelet function ψ

$$\psi(x) = \begin{cases} 1 & \text{iff} \quad 0 \leq x < 0.5 \\ -1 & \text{iff} \quad 0.5 \leq x < 1 \\ 0 & \text{elsewhere} \end{cases}$$



New basis
Scaling function

Let $\varphi(x)$ be a scaling function. We will modify it as follows:

$$\varphi_{j,k}(x) = 2^{j/2}\varphi\left(2^j \frac{x}{M} - k\right)$$

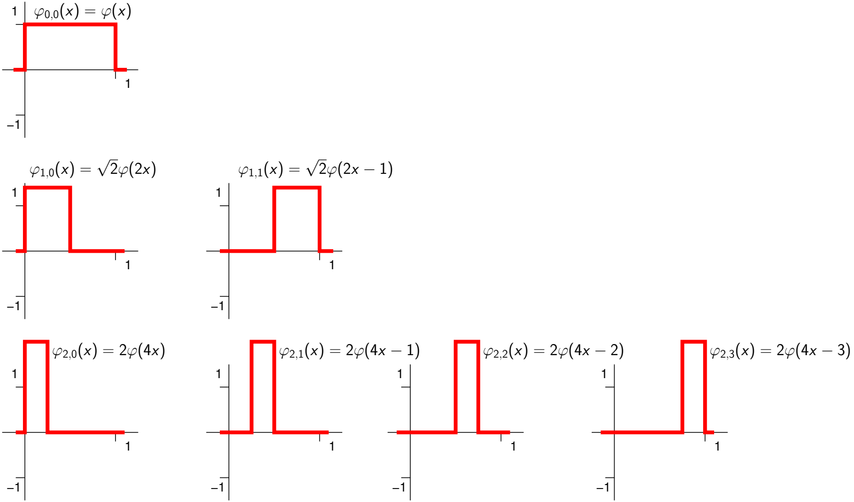
where $j, k \in \mathbb{Z}$ then

- j . . . $\varphi_{j,k}(x)$'s width
  controls broadness and height of the function along x and y axis, respectively.
- k . . . shift of $\varphi_{j,k}(x)$ along x-axis
- M . . . length of the processed signal

Notice: The function is orthogonal to its integer shifts

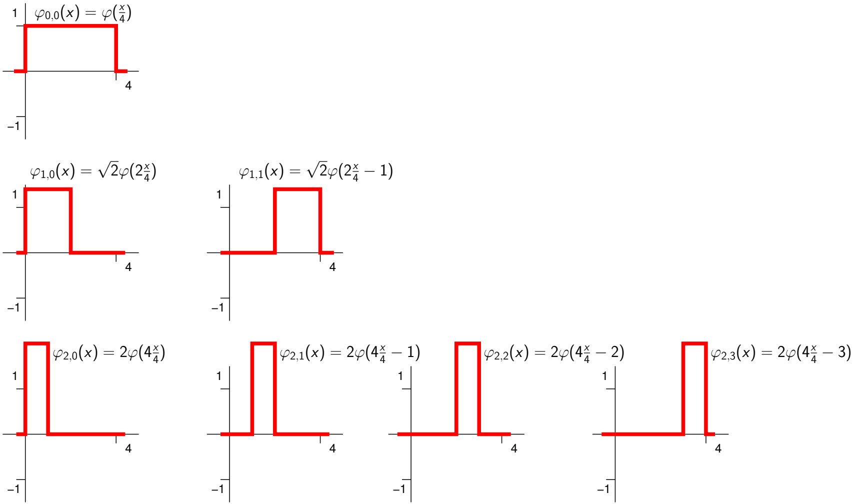
New basis

Scaling function ($M = 1$) – examples



New basis

Scaling function ($M = 4$) – examples



New basis

Wavelet function

Let $\psi(x)$ be a wavelet function. We will modify it as follows:

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

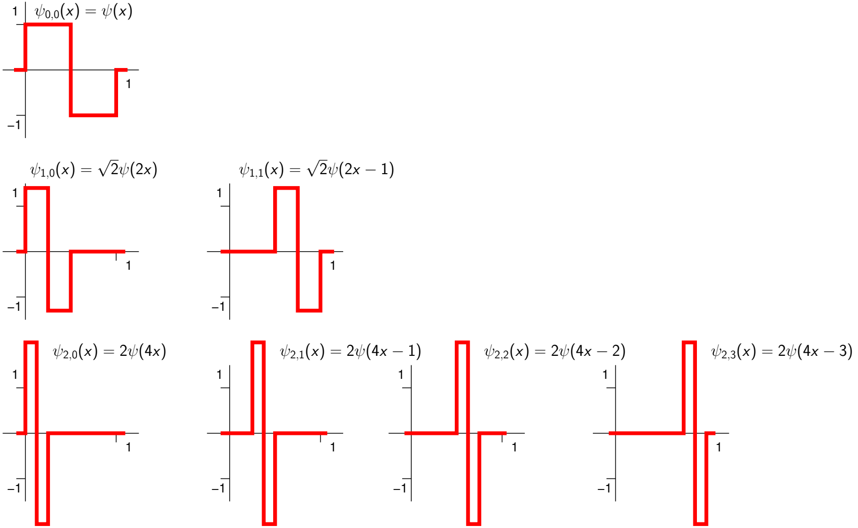
where $j, k \in \mathbb{Z}$ then

- $j \ldots \varphi_{j,k}(x)$'s width
  controls broadness and height of the function along x and y axis, respectively.
- $k \ldots$ shift of $\varphi_{j,k}(x)$ along x-axis
- $M \ldots$ length of the processed signal

Notice: The function is orthogonal to its integer shifts

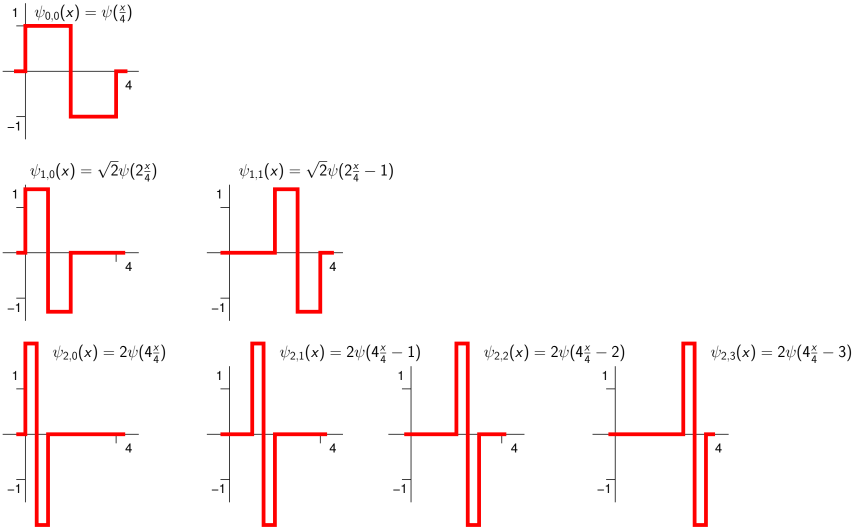
New basis

Wavelet function ($M = 1$) – examples



New basis

Wavelet function ($M = 4$) – examples



New basis

Support of basis functions

Fourier basis

- The domain of sin and cos is $\langle 0; 1 \rangle$.
- The domain is periodic.
- When applying the Fourier basis function $\varphi_m(k)$ to a transformed function f of length N, this basis function $\varphi_m(k)$ (its period) is stretched to the length N.

New basis

- The domain of $\varphi(x)$ and $\psi(x)$ is $\langle 0; 1 \rangle$.
- Each function has limited compact support.
- When applying the scaling or wavelet function to a transformed function f of length M, both scaling and wavelet function are appropriately stretched to the required length.

Let us recall that the basis forms the rows of transform matrix!

Sample matrix for signal of length $M = 4$

$$HT_4 = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{pmatrix} \begin{matrix} = & \varphi_{0,0} \\ = & \psi_{0,0} \\ = & \psi_{1,0} \\ = & \psi_{1,1} \end{matrix}$$

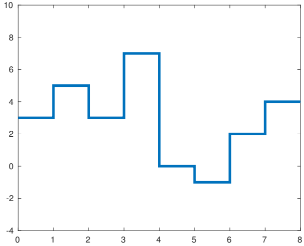
- Notice the structure of individual rows.
- Explain the value $\frac{1}{2}$.

New basis

Design of transform matrix

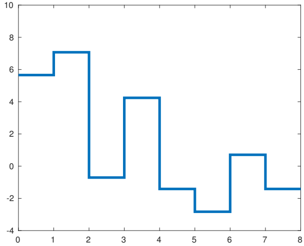
Matrices for signal of length $M = 8$

$$HT_8 = \frac{1}{\sqrt{8}} \begin{pmatrix} 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{pmatrix} \begin{matrix} = \varphi_{2,0} \\ = \varphi_{2,1} \\ = \varphi_{2,2} \\ = \varphi_{2,3} \\ = \psi_{2,0} \\ = \psi_{2,1} \\ = \psi_{2,2} \\ = \psi_{2,3} \end{matrix}$$



f = [3 5 3 7 0 -1 2 4]

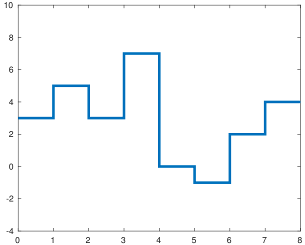
$\Rightarrow$



HT8 * f'

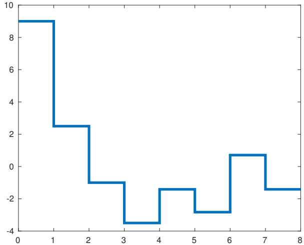
Matrices for signal of length $M = 8$

$$HT_8 = \frac{1}{\sqrt{8}} \begin{pmatrix} \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{pmatrix} \begin{array}{l} = \varphi_{1,0} \\ = \varphi_{1,1} \\ = \psi_{1,0} \\ = \psi_{1,1} \\ = \psi_{2,0} \\ = \psi_{2,1} \\ = \psi_{2,2} \\ = \psi_{2,3} \end{array}$$



f = [3 5 3 7 0 -1 2 4]

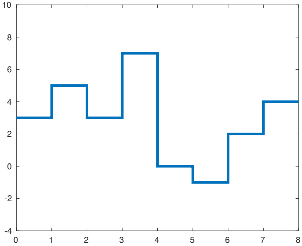
$\Rightarrow$



HT8 * f'

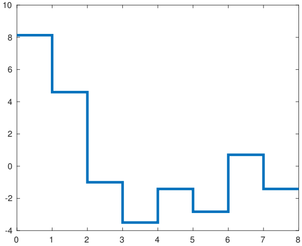
Matrices for signal of length $M = 8$

$$HT_8 = \frac{1}{\sqrt{8}} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{pmatrix} \begin{matrix} = \varphi_{0,0} \\ = \psi_{0,0} \\ = \psi_{1,0} \\ = \psi_{1,1} \\ = \psi_{2,0} \\ = \psi_{2,1} \\ = \psi_{2,2} \\ = \psi_{2,3} \end{matrix}$$



`f = [3 5 3 7 0 -1 2 4]`

$\Rightarrow$



`HT8 * f'`

1D Discrete Wavelet Transform (DWT)

Forward

$$A_{j_0}(k) = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} f(m)\varphi_{j_0,k}(m)$$

$$D_j(k) = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} f(m)\psi_{j,k}(m)$$

Inverse

$$f(m) = \frac{1}{\sqrt{M}} \sum_{k=0}^{2^{j_0}-1} A_{j_0}(k)\varphi_{j_0,k}(m) + \frac{1}{\sqrt{M}} \sum_{j=j_0}^{J-1} \sum_{k=0}^{2^j-1} D_j(k)\psi_{j,k}(m)$$

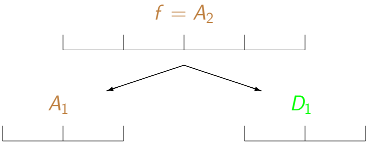
- φ, ψ ... orthogonal scaling and wavelet function, respectively
- $A_{j_0}(k)$ ... scaling coefficients (approximations)
- $D_j(k)$ ... wavelet coefficients (details)
- $M = 2^J$ ... number of samples in function f
- $j \in \{j_0, ..., J-1\}$ ... level of detail, where $j_0 \geq 0$
- $k \in \{0, 1, \dots, 2^j - 1\}$

1D Discrete Wavelet Transform (DWT)

An example

Input:

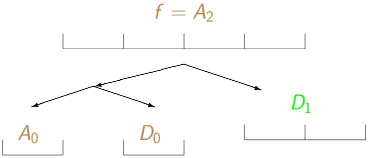
- $f = [1, 4, -3, 0]$
- $|f| = M = 4 \quad \Rightarrow \quad J = 2$
- decomposition level: $j_0 = 1$



$$
\begin{aligned}
A_1(0) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\varphi_{1,0}(m) = \frac{1}{2}[1\cdot\sqrt{2} + 4\cdot\sqrt{2} + (-3)\cdot 0 + 0\cdot 0] = \frac{5}{\sqrt{2}} \\
A_1(1) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\varphi_{1,1}(m) = \frac{1}{2}[1\cdot 0 + 4\cdot 0 + (-3)\cdot\sqrt{2} + 0\cdot\sqrt{2}] = \frac{-3}{\sqrt{2}} \\
D_1(0) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\psi_{1,0}(m) = \frac{1}{2}[1\cdot\sqrt{2} + 4\cdot(-\sqrt{2}) - 3\cdot 0 + 0\cdot 0] = -\frac{3}{\sqrt{2}} \\
D_1(1) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\psi_{1,1}(m) = \frac{1}{2}[1\cdot 0 + 4\cdot 0 - 3\cdot\sqrt{2} + 0\cdot(-\sqrt{2})] = -\frac{3}{\sqrt{2}}
\end{aligned}
$$

Input:

- $f = [1, 4, -3, 0]$
- $|f| = M = 4 \quad \Rightarrow \quad J = 2$
- decomposition level: $j_0 = 0$



$$
\begin{aligned}
A_0(0) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\varphi_{0,0}(m) = \frac{1}{2}[1\cdot 1 + 4\cdot 1 - 3\cdot 1 + 0\cdot 1] = 1\\
D_0(0) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\psi_{0,0}(m) = \frac{1}{2}[1\cdot 1 + 4\cdot 1 - 3\cdot(-1) + 0\cdot(-1)] = 4\\
D_1(0) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\psi_{1,0}(m) = \frac{1}{2}[1\cdot\sqrt{2} + 4\cdot(-\sqrt{2}) - 3\cdot 0 + 0\cdot 0] = -\frac{3}{2}\sqrt{2}\\
D_1(1) &= \frac{1}{2}\sum_{m=0}^{3} f(m)\psi_{1,1}(m) = \frac{1}{2}[1\cdot 0 + 4\cdot 0 - 3\cdot\sqrt{2} + 0\cdot(-\sqrt{2})] = -\frac{3}{2}\sqrt{2}
\end{aligned}
$$

1D Discrete Wavelet Transform (DWT)

An example

$\text{DWT}(f) = [1, 4, -\frac{3}{2}\sqrt{2}, -\frac{3}{2}\sqrt{2}]$, i.e.

$$
\begin{aligned}
f(m) &= IDWT([1, 4, -\frac{3}{2}\sqrt{2}, -\frac{3}{2}\sqrt{2}]) \\
&= \frac{1}{2} A_0(0) \cdot \varphi_{0,0}(m) + \\
&\quad \frac{1}{2}\left(D_0(0) \cdot \psi_{0,0}(m) + D_1(0) \cdot \psi_{1,0}(m) + D_1(1) \cdot \psi_{1,1}(m)\right) \\
&= \frac{1}{2} \cdot 1 \cdot \varphi_{0,0}(m) + \\
&\quad \frac{1}{2}\left(4 \cdot \psi_{0,0}(m) - \frac{3}{2}\sqrt{2} \cdot \psi_{1,0}(m) - \frac{3}{2}\sqrt{2} \cdot \psi_{1,1}(m)\right)
\end{aligned}
$$

Utilization of the same basis functions in forward and inverse transforms is conditioned to orthonormality of selected functions.

1D Discrete Wavelet Transform (DWT)

An example

After submitting signal

$$f = \boxed{f(0)}\ \boxed{f(1)}\ \boxed{f(2)}\ \boxed{f(3)}$$

to 1D-DWT, we obtain separately approximations and details of the signal:

- for $j_0 = 2$:
  no decomposition
- for $j_0 = 1$:
  $\mathrm{DWT(f)} = \boxed{A_1(0)}\ \boxed{A_1(1)}\ \boxed{D_1(0)}\ \boxed{D_1(1)}$
- for $j_0 = 0$:
  $\mathrm{DWT(f)} = \boxed{A_0(0)}\ \boxed{D_0(0)}\ \boxed{D_1(0)}\ \boxed{D_1(1)}$

Notice: The output signal is always of the same length as the input signal.

The common families of scaling (father) and wavelet (mother) functions

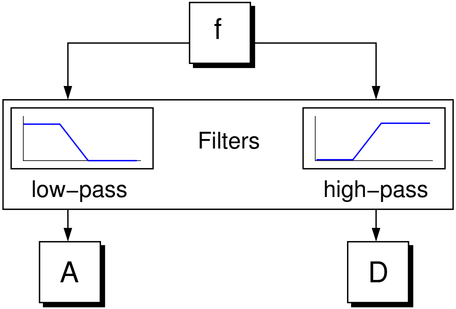
- Haar (already introduced)
- Daubechies: db1, db2, db3, db4, ...
- Meyer
- Coiflets: coif1, coif2, coif3, ...
- Symlets: sym2, sym3, sym4, ...
- Biorthogonal: bior1, bior2, bior3, ...

Complexity of 1D-DWT

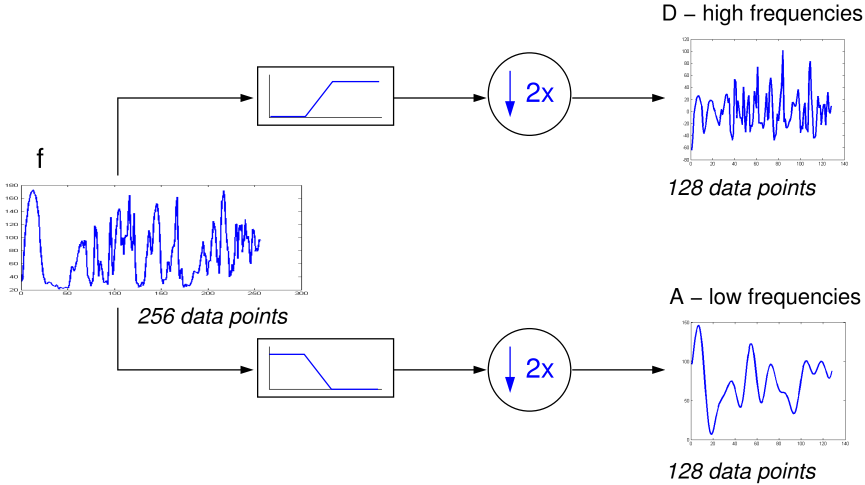
- matrix multiplication – $O(n^2)$
- the whole transform matrix typically built only for Haar wavelets
- other wavelets computed iteratively
  (one matrix per one level of decomposition) $\Rightarrow$ *iterations* $\times O(n^2)$
- can we speed it up?

Any signal f can be decomposed into two parts:

- approximation (A) . . . obtained by low-pass filtering of the original signal
- detail (D) . . . obtained by high-pass filtering of the original signal

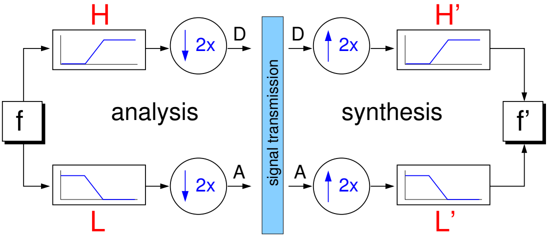


The filtered signal must be downsampled ($\downarrow 2\times$) to avoid data redundancy.



The decomposed signal may be reconstructed:

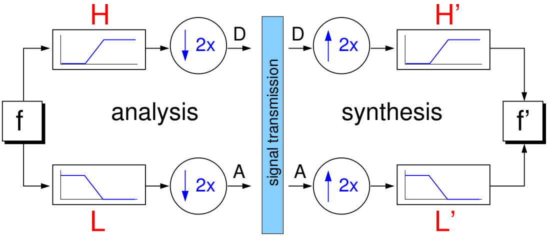
- detail (D) is upsampled ($\uparrow 2\times$) and then high-pass filtered
- approximation (A) is upsampled ($\uparrow 2\times$) and then low-pass filtered
- results are added $\rightarrow f'$



Notice: We would like to have $f = f'$

Filter banks

- H ... high-pass analysis filter (FIR)
- L ... low-pass analysis filter (FIR)
- H' ... high-pass synthesis filter (FIR)
- L' ... low-pass synthesis filter (FIR)

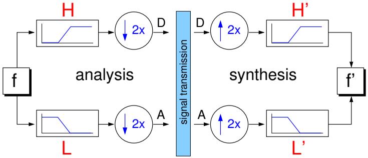


Filter banks

If $f = f'$ then the filters L, L', H, H' are called perfect reconstruction filters and they must fulfill one of the following conditions:

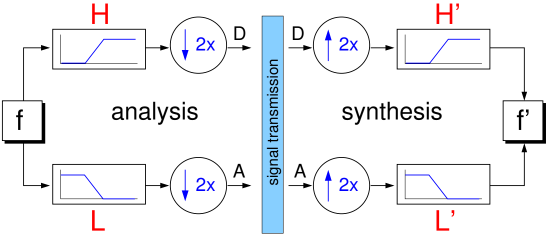
$$\begin{aligned} H'(n) &= (-1)^n L(n) \\ L'(n) &= (-1)^{n+1} H(n) \end{aligned}$$

$$\begin{aligned} H'(n) &= (-1)^{n+1} L(n) \\ L'(n) &= (-1)^n H(n) \end{aligned}$$



Filter banks

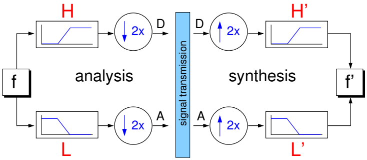
- H and L' are mutually cross-modulated
- H' and L are mutually cross-modulated
- H, H', L, L' are called quadrature mirror filters (QMF)



Filter banks

Biorthogonal filters

We need to define two filters H and L. The remaining H' and L' are derived by cross-modulation.



Orthogonal filters

We define only one filter H'. The remaining filters fulfill:

$$
\begin{aligned}
L'(n) &= (-1)^n H'(length - 1 - n) \\
H(n) &= H'(length - 1 - n) \\
L(n) &= L'(length - 1 - n)
\end{aligned}
$$

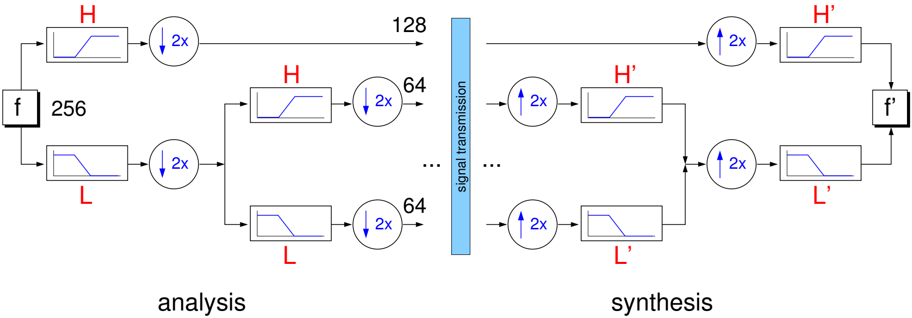
where

$length = \mathtt{size}(H')$ &

$\mathtt{is\_even}(length) = true$

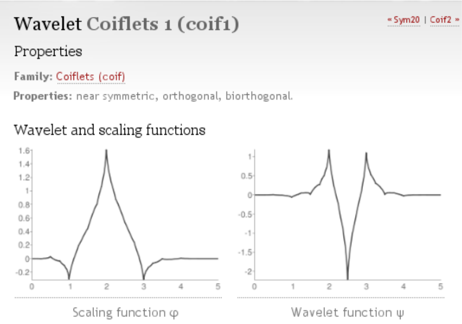
Notice: We will focus namely on the orthogonal filters.

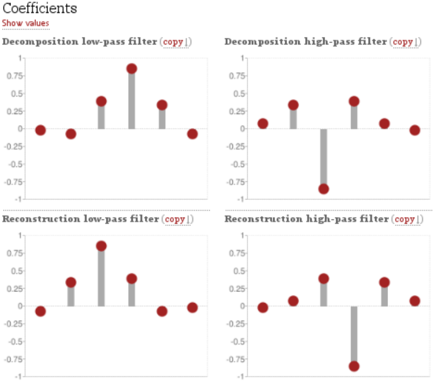
Once the input signal is decomposed into two parts (A and D), its approximation (A) can be further decomposed. In the reverse order, the same is valid for reconstruction.



Notice: Let us assume we have already employed (bi)orthogonal filters.

...and their scaling and wavelet functions



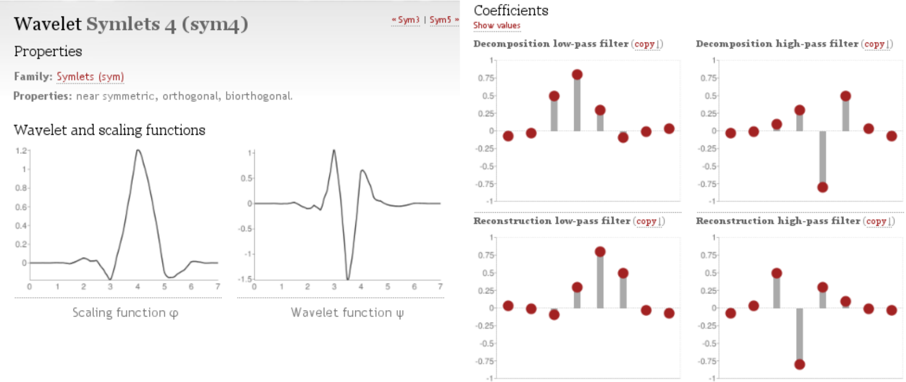


Notice: Useful web-pages: `http://wavelets.pybytes.com/`

...and their scaling and wavelet functions

Notice: Useful web-pages: `http://wavelets.pybytes.com/`

. . . and their scaling and wavelet functions



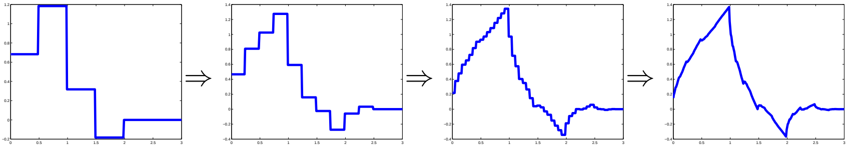
Notice: Useful web-pages: `http://wavelets.pybytes.com/`

. . . and their scaling and wavelet functions

Notice: Useful web-pages: `http://wavelets.pybytes.com/`

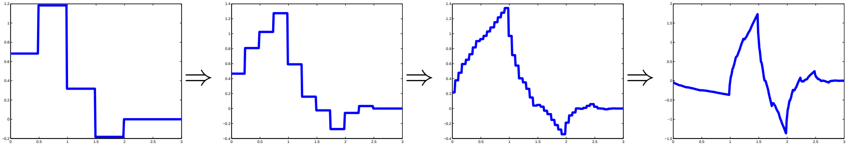
Algorithm

$L' \leftarrow$ fetch low-pass synthesis filter from the selected filter bank
$h_\varphi = \texttt{fliplr}(L')$
$\varphi \leftarrow$ Dirac delta impulse
while (φ is converging) **do**
    $\varphi \leftarrow \texttt{conv}(\varphi, h_\varphi)$
    $\varphi \leftarrow \texttt{upsample}(\varphi, 2\times)$
end while
OUTPUT $\leftarrow \varphi$



Algorithm

1: $\varphi \leftarrow$ call Cascade algorithm to get φ function
2: $H' \leftarrow$ fetch high-pass synthesis filter from the selected filter bank
3: $h_\psi = \texttt{fliplr}(H')$
4: $\psi \leftarrow \texttt{conv}(\varphi, h_\psi)$
5: $\psi \leftarrow \texttt{upsample}(\psi, 2\times)$
6: OUTPUT $\leftarrow \psi$



1D Fast Discrete Wavelet Transform

Definition

$$
\begin{aligned}
D_j(k) &= \sum_{r=0}^{|A_{j+1}|-1} H'(2k+1-r)A_{j+1}(r) \\
A_j(k) &= \sum_{r=0}^{|A_{j+1}|-1} L'(2k+1-r)A_{j+1}(r) \\
A_J(k) &= f(k)
\end{aligned}
$$

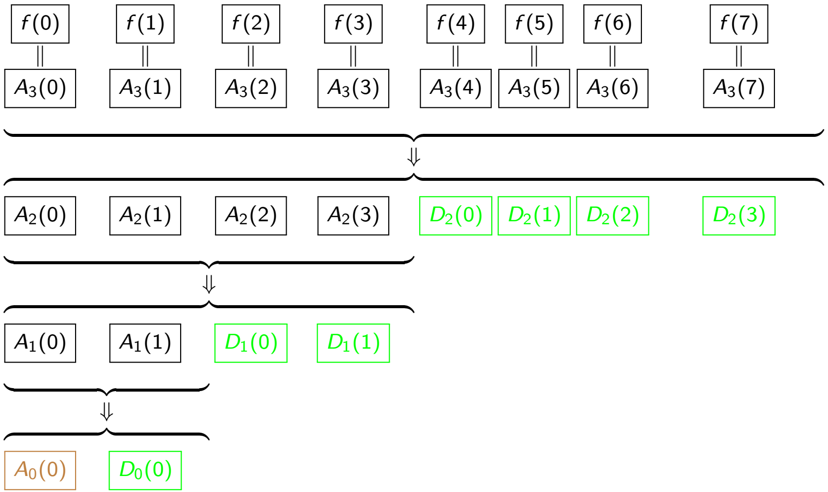
Each step in FWT corresponds to convolution with high-pass and low-pass analysis filter followed by down-sampling ($\downarrow 2\times$).

1D-DWT $\equiv$ Subband coding

1D Fast Discrete Wavelet Transform

Basic scheme

Let $|f| = M = 8 = 2^3 = 2^J$ and $j_0 = 0$



Fast Wavelet Transform

An example

Given $f(k) = [1, 4, -3, 0]$ and Haar scaling and wavelet coefficients

$$L'(k) = \begin{cases} 1/\sqrt{2} & k = 0, 1 \\ 0 & \text{otherwise} \end{cases} \quad \text{/and/} \quad H'(k) = \begin{cases} -1/\sqrt{2} & k = 0 \\ 1/\sqrt{2} & k = 1 \\ 0 & \text{otherwise} \end{cases}$$

we can evaluate the following:

$$\begin{aligned}
\text{level 2:} \quad A_2(k) &= f(k) = [1, 4, -3, 0] \\
\text{level 1:} \quad A_1(k) &= \sum_{r=0}^{3} L'(2k+1-r) A_2(r) = [5/\sqrt{2}, -3/\sqrt{2}] \\
D_1(k) &= \sum_{r=0}^{3} H'(2k+1-r) A_2(r) = [-3/\sqrt{2}, -3/\sqrt{2}] \\
\text{level 0:} \quad A_0(k) &= \sum_{r=0}^{1} L'(2k+1-r) A_1(r) = [1] \\
D_0(k) &= \sum_{r=0}^{1} H'(2k+1-r) A_1(r) = [4]
\end{aligned}$$

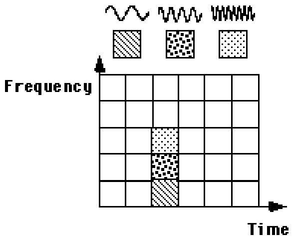
Comparison of FWT and FFT

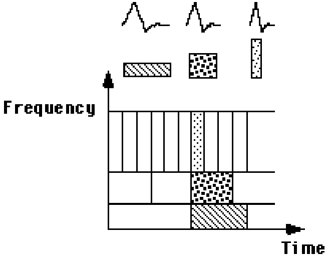
Fast Fourier Transform

- complexity: $O(n \log n)$
- existence: at any time
- time *versus* frequency domain

Fast Wavelet Transform

- complexity $O(cn)$
  c ...support of L' filter (typically small)
- existence: depends upon the availability of scaling function and the orthogonality of the scaling function
- time & frequency changes simultaneously





2D Discrete Wavelet Transform

An extension of scaling function and wavelets to 2D in straightforward:

$$\begin{aligned}\varphi(x) &\rightarrow \varphi(x,y)\\ \psi(x) &\rightarrow \psi^H(x,y),\psi^V(x,y),\psi^D(x,y)\end{aligned}$$

where all the 2D functions are separable in the following manner:

$$\begin{aligned}\varphi(x,y) &= \varphi(x)\varphi(y)\\ \psi^H(x,y) &= \psi(x)\varphi(y)\\ \psi^V(x,y) &= \varphi(x)\psi(y)\\ \psi^D(x,y) &= \psi(x)\psi(y)\end{aligned}$$

2D Discrete Wavelet Transform

What is the meaning of new wavelets?

- $\psi^H(x, y)$ ... intensity variations for image columns
- $\psi^V(x, y)$ ... intensity variations along rows
- $\psi^D(x, y)$ ... intensity variations along diagonals

Corollary:

$$\varphi_{j,m,n}(x, y) = 2^{j/2}\varphi\left(2^j\frac{x}{M} - m, 2^j\frac{y}{N} - n\right)$$
$$\psi^i_{j,m,n}(x, y) = 2^{j/2}\psi^i\left(2^j\frac{x}{M} - m, 2^j\frac{y}{N} - n\right), \quad i = \{H, V, D\}$$

2D Discrete Wavelet Transform
Definition

Forward

$$A_{j_0}(m,n) = \frac{1}{\sqrt{MN}} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} f(k,l)\varphi_{j_0,m,n}(k,l)$$

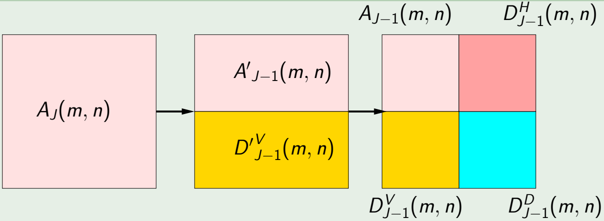
$$D_j^i(m,n) = \frac{1}{\sqrt{MN}} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} f(k,l)\psi_{j,m,n}^i(k,l)$$

Inverse

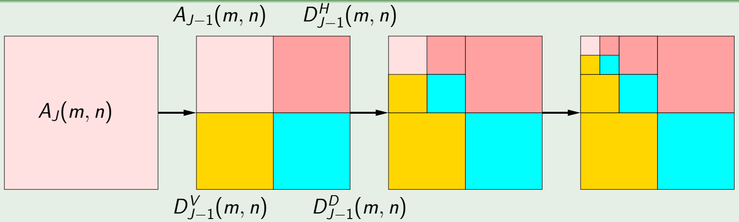
$$f(k,l) = \frac{1}{\sqrt{MN}} \sum_m \sum_n A_{j_0}(m,n)\varphi_{j_0,m,n}(k,l) + \frac{1}{\sqrt{MN}} \sum_{i=H,V,D} \sum_{j=j_0}^{J-1} \sum_m \sum_n D_j^i(m,n)\psi_{j,m,n}^i(k,l)$$

where $i = \{H, V, D\}$

1-level decomposition as a 2-step process



n-th level decomposition as a iterative process

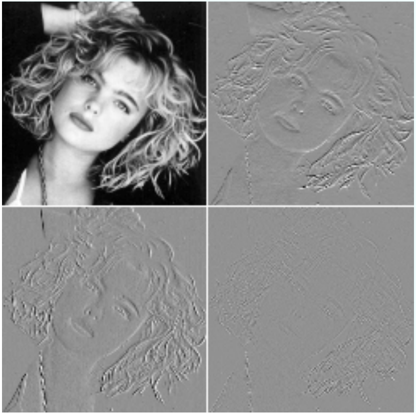


Level of detail: $j_0 = J$

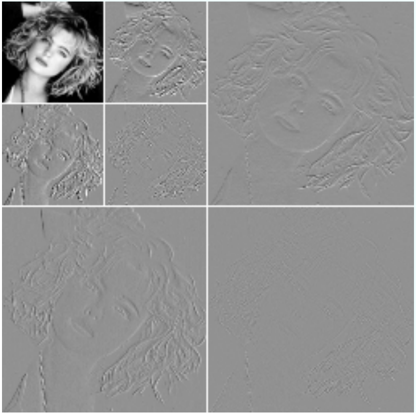
2D Discrete Wavelet Transform

An example – DWT using Haar wavelets

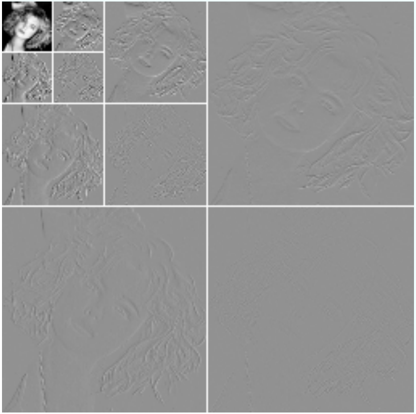
Level of detail: $j_0 = J - 1$



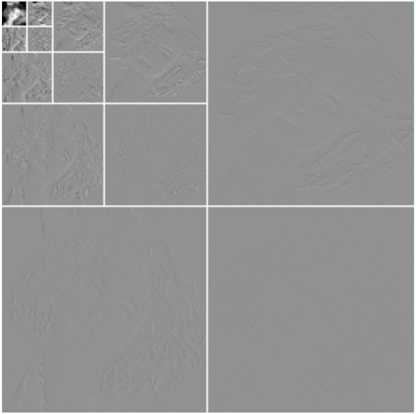
Level of detail: $j_0 = J - 2$



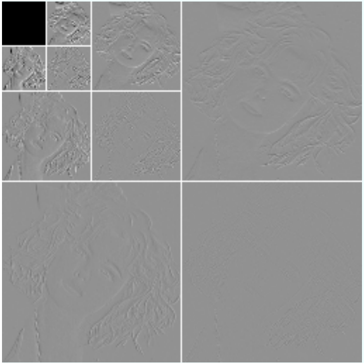
Level of detail: $j_0 = J - 3$

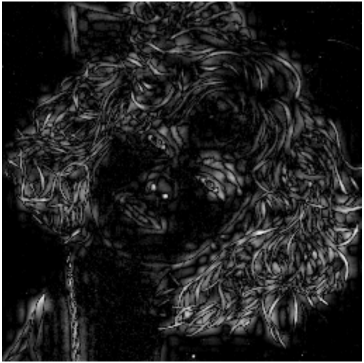


Level of detail: $j_0 = J - 4$



DWT $\rightarrow$ modification $\rightarrow$ IDWT





Wavelet Packets

Problem to solve:

Traditional wavelet transform decomposes the (image) data always in the same manner.

Solution:

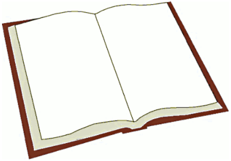
Decompose those parts of the data which need it.

An example:

The lowest entropy lead to better compression. Let us split those parts of the image (not only $A_j(m, n)$) which need it $\rightarrow$ which division causes entropy reduction.

Bibliography

- Burt P. J., Adelson E. H. The Laplacian Pyramid as a Compact Image Code, IEEE Trans. on Communications, pp. 532–540, April 1983
- Gonzalez, R. C., Woods, R. E. Digital image processing / 2nd ed., Upper Saddle River: Prentice Hall, 2002, pages 793, ISBN 0201180758
- Klette R., Zamperoni P. Handbook of Image Processing Operators, Wiley, 1996, ISBN-0471956422
- Strang G., Nguyen T. Wavelets and Filter Banks, Wellesley-Cambridge Press, 1997, ISBN 0-9614088-7-1



You should know the answers . . .

- Explain the difference between Fourier basis functions and scaling and wavelet functions.
- Given a signal of fixed length and given a particular scaling a wavelet function we can perform discrete wavelet transform. The result is however not unique. Which parameter controls the behaviour of DWT? Demonstrate on some sample data.
- Explain the meaning of A and D coefficients.
- Derive the complexity for DWT and separately for FWT.
- What would happen if the quadrature mirror filters are not *perfect reconstruction filters*.
- Describe the *Cascade algorithm*.
- Design an algorithm for computing 2D-FWT.