Image Processing and Analysis – Math Part: The Discrete Wavelet Transform

Ivar Farup

Based upon Chapter 6 of Broughton and Bryan's *Discrete Fourier Analysis and Wavelets* and Maciej Piętka's Lecture Notes from 2010

Overview

- Review of the Haar filter bank
- The general one-stage two-channel filter bank
- Multistage filter banks
- The discrete wavelet transform, DWT
- The 2D DWT

Review of the Haar Filter Bank

Analysis Filter Bank

$$\begin{array}{ccccccccc} & & [\boldsymbol{\ell}_a] & \rightarrow & [\downarrow 2] & \rightarrow & \mathbf{X}_\ell & & \\ & \nearrow & & & & & & \searrow & \\ \mathbf{x} & & & & & & & & \mathbf{X} = (\mathbf{X}_\ell, \mathbf{X}_h) \\ & \searrow & & & & & & \nearrow & \\ & & [\mathbf{h}_a] & \rightarrow & [\downarrow 2] & \rightarrow & \mathbf{X}_h & & \end{array}$$

Review of the Haar Filter Bank

Synthesis Filter Bank

$$
\begin{array}{ccccccccc}
 & & \mathbf{X}_l & \rightarrow & [\uparrow 2] & \rightarrow & [\boldsymbol{\ell}_s] & & \\
 & \nearrow & & & & & & \searrow & \\
\mathbf{X} & & & & & & & & \mathbf{x} \\
 & \searrow & & & & & & \nearrow & \\
 & & \mathbf{X}_h & \rightarrow & [\uparrow 2] & \rightarrow & [\mathbf{h}_s] & &
\end{array}
$$

Review of the Haar Filter Bank

Analysis/Synthesis Filter Bank

$$
\begin{array}{ccccccccccc}
 & & [\boldsymbol{\ell}_a] & \to & [\downarrow 2] & \to & & \to & [\uparrow 2] & \to & [\boldsymbol{\ell}_s] & & \\
 & \nearrow & & & & & \text{compress,} & & & & & \searrow & \\
\mathbf{x} & & & & & & \text{denoise,} & & & & & & \tilde{\mathbf{x}} \\
 & \searrow & & & & & \text{etc.} & & & & & \nearrow & \\
 & & [\mathbf{h}_a] & \to & [\downarrow 2] & \to & & \to & [\uparrow 2] & \to & [\mathbf{h}_s] & &
\end{array}
$$

The General One-Stage Two-Channel Filter Bank

- Analysis filters $\boldsymbol{\ell}_a, \mathbf{h}_a$ need not be 2-point averaging/difference filters
- Any pair of low-/high-pass finite impulse response filters would do

The General One-Stage Two-Channel Filter Bank

- The reconstructed signal $\mathbf{x}'$ should be a copy of the input signal $\mathbf{x}$ (possibly delayed)
- Such a filter bank is called a *perfect reconstruction filter bank*
- If the above condition is satisfied, synthesis filters are determined from analysis filters

- Analysis filters:

$$\boldsymbol{\ell}_a = (\dots, 0, -\frac{1}{8}, \frac{1}{4}, \frac{3}{4}, \frac{1}{4}, -\frac{1}{8}, 0, \dots)$$
$$\mathbf{h}_a = (\dots, 0, -\frac{1}{2}, 1, -\frac{1}{2}, 0, \dots)$$

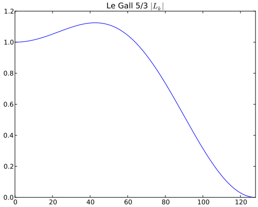
- Synthesis filters:

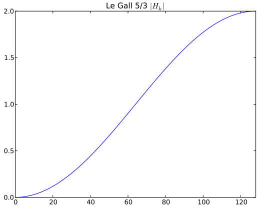
$$\boldsymbol{\ell}_s = (\dots, 0, \frac{1}{2}, 1, \frac{1}{2}, 0, \dots)$$
$$\mathbf{h}_s = (\dots, 0, -\frac{1}{8}, -\frac{1}{4}, \frac{3}{4}, -\frac{1}{4}, -\frac{1}{8}, 0, \dots)$$

The General One-Stage Two-Channel Filter Bank

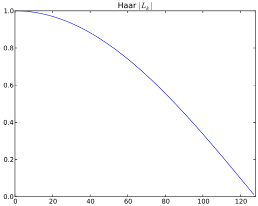
Example: Le Gall 5/3 Filters

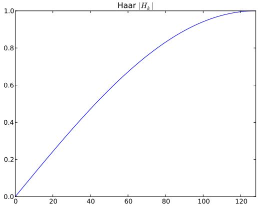
Frequency response of the Le Gall 5/3 analysis filters





Frequency response of the Haar analysis filters





Multistage Filter Banks

General Idea

- Consider filter bank transform $\mathbf{x} \to \mathbf{X} = (\mathbf{X}_\ell \mathbf{X}_h)$
- Detail coefficients $\mathbf{X}_h$ are not processed further
- Approximation coefficients are input to second-stage filter bank
- Can continue iteratively ("multistage analysis")

$$
\begin{array}{ccccccccccccccc}
 & & & & & & & & & & & & & & \cdots \\
 & & & & & & & & & & & & & \nearrow & \\
 & & & & & & & & [\boldsymbol{\ell}_a] & \to & [\downarrow 2] & \to & \mathbf{X}_{\ell\ell} & & \\
 & & & & & & & \nearrow & & & & & & \searrow & \\
 & & [\boldsymbol{\ell}_a] & \to & [\downarrow 2] & \to & \mathbf{X}_\ell & & & & & & & & \cdots \\
 & \nearrow & & & & & & \searrow & & & & & & & \\
\mathbf{x} & & & & & & & & [\mathbf{h}_a] & \to & [\downarrow 2] & \to & \mathbf{X}_{\ell h} & & \\
 & \searrow & & & & & & & & & & & & & \\
 & & [\mathbf{h}_a] & \to & [\downarrow 2] & \to & \mathbf{X}_h & & & & & & & &
\end{array}
$$

Filter bank tree for multistage analysis

- Multistage analysis filter bank transform

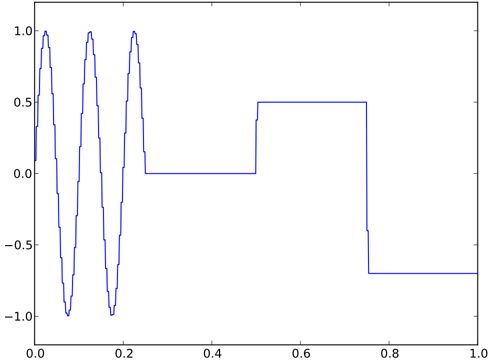
$$\mathbf{x} \rightarrow \begin{pmatrix} \mathbf{x}_\ell \\ \mathbf{x}_h \end{pmatrix} \rightarrow \begin{pmatrix} \mathbf{x}_{\ell\ell} \\ \mathbf{x}_{\ell h} \\ \mathbf{x}_h \end{pmatrix} \rightarrow \begin{pmatrix} \mathbf{x}_{\ell\ell\ell} \\ \mathbf{x}_{\ell\ell h} \\ \mathbf{x}_{\ell h} \\ \mathbf{x}_h \end{pmatrix} \rightarrow \begin{pmatrix} \mathbf{x}_{\ell\ell\ell\ell} \\ \mathbf{x}_{\ell\ell\ell h} \\ \mathbf{x}_{\ell\ell h} \\ \mathbf{x}_{\ell h} \\ \mathbf{x}_h \end{pmatrix} \rightarrow \cdots$$

- Multistage synthesis steps are applied in reverse

Multistage Filter Banks

4-Fold Signal Compression with Haar Filter

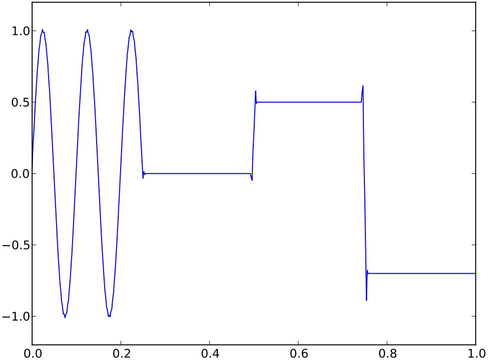
$$\mathbf{x} \to \begin{pmatrix} \mathbf{X}_{\ell\ell} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \to \begin{pmatrix} \mathbf{X}_{\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \to \tilde{\mathbf{x}}$$



Multistage Filter Banks

4-Fold Signal Compression with Daubechy 4-Tap Filter

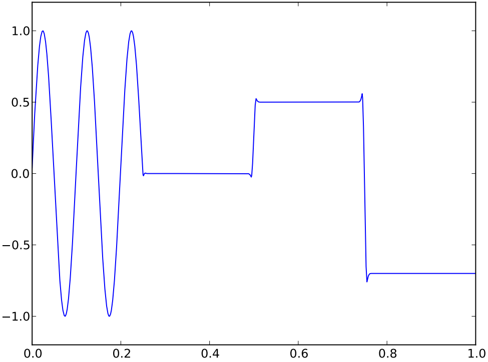
$$\mathbf{x} \to \begin{pmatrix} \mathbf{X}_{\ell\ell} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \to \begin{pmatrix} \mathbf{X}_{\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \to \tilde{\mathbf{x}}$$



Multistage Filter Banks

4-Fold Signal Compression with Biorthogonal3.3 (8-Tap Filter)

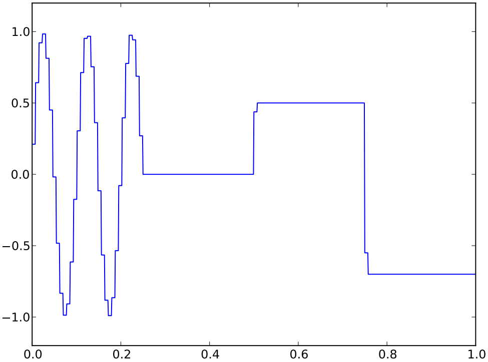
$$\mathbf{x} \to \begin{pmatrix} \mathbf{X}_{\ell\ell} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \to \begin{pmatrix} \mathbf{X}_{\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \to \tilde{\mathbf{x}}$$



Multistage Filter Banks

8-Fold Signal Compression with Haar Filter

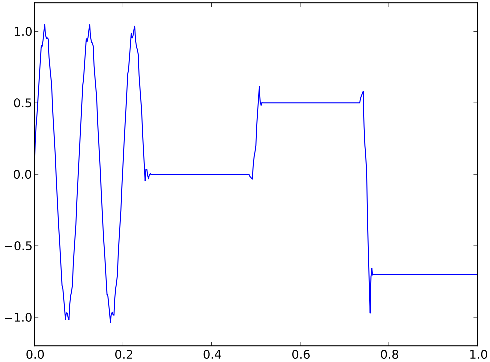
$$\mathbf{x} \rightarrow \begin{pmatrix} \mathbf{X}_{\ell\ell\ell} \\ \mathbf{X}_{\ell\ell h} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \rightarrow \begin{pmatrix} \mathbf{X}_{\ell\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \rightarrow \tilde{\mathbf{x}}$$



Multistage Filter Banks

8-Fold Signal Compression with Daubechy 4-Tap Filter

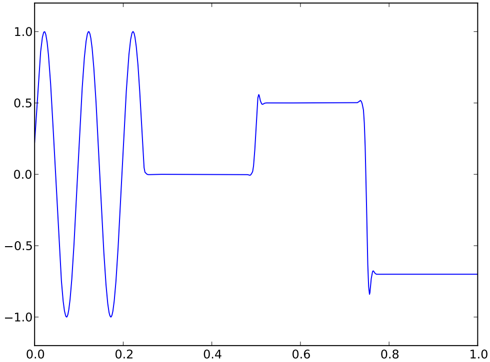
$$\mathbf{x} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell} \\ \mathbf{X}_{\ell\ell h} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \to \tilde{\mathbf{x}}$$



Multistage Filter Banks

8-Fold Signal Compression with Biorthogonal3.3 (8-Tap Filter)

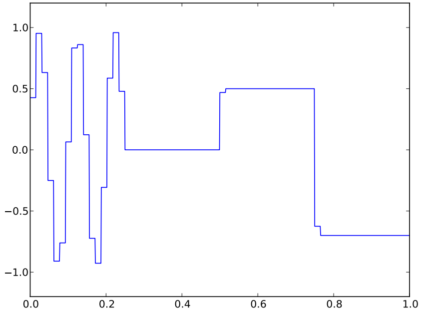
$$\mathbf{x} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell} \\ \mathbf{X}_{\ell\ell h} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \to \tilde{\mathbf{x}}$$



Multistage Filter Banks

16-Fold Signal Compression with Haar Filter

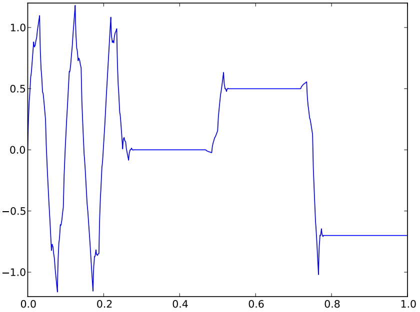
$$\mathbf{x} \rightarrow \begin{pmatrix} \mathbf{X}_{\ell\ell\ell\ell} \\ \mathbf{X}_{\ell\ell\ell h} \\ \mathbf{X}_{\ell\ell h} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \rightarrow \begin{pmatrix} \mathbf{X}_{\ell\ell\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \rightarrow \tilde{\mathbf{x}}$$



Multistage Filter Banks

16-Fold Signal Compression with Daubechy 4-Tap Filter

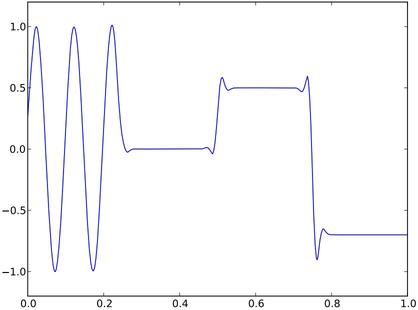
$$\mathbf{x} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell\ell} \\ \mathbf{X}_{\ell\ell\ell h} \\ \mathbf{X}_{\ell\ell h} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \to \tilde{\mathbf{x}}$$



Multistage Filter Banks

16-Fold Signal Compression with Biorthogonal3.3 (8-Tap Filter)

$$\mathbf{x} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell\ell} \\ \mathbf{X}_{\ell\ell\ell h} \\ \mathbf{X}_{\ell\ell h} \\ \mathbf{X}_{\ell h} \\ \mathbf{X}_{h} \end{pmatrix} \to \begin{pmatrix} \mathbf{X}_{\ell\ell\ell\ell} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \to \tilde{\mathbf{x}}$$



- The above formalism was for signals $\mathbf{x} \in L^2(\mathbb{Z})$
- Adopt it for finite lenght signals $\mathbf{x} \in \mathbb{C}^N$ by
  - Extending $\mathbf{x}$ to $\tilde{\mathbf{x}} \in L^2(\mathbb{Z})$
  - Perform the filtering on $L^2(\mathbb{Z})$ vectors
  - Truncate the filtered signal to a finite length

- Zero-padding: $\tilde{x}_k = 0$ for $k < 0$ and $k \geq N$

$$\mathbf{x} = (a, b, c) \quad \Rightarrow$$

$$\tilde{\mathbf{x}} = (\ldots, 0, 0, 0, 0, 0, 0, a, b, c, 0, 0, 0, 0, 0, 0, \ldots)$$

- Perodic extension $\tilde{x}_k = x_{x \mod N}$ (like DFT)

$$\mathbf{x} = (a, b, c) \quad \Rightarrow$$

$$\tilde{\mathbf{x}} = (\dots, a, b, c, a, b, c, , a, b, c, a, b, c, a, b, c, \dots)$$

- Half-point symmetric extension: reflect the signal about one of its enpoints, then extend periodically (like DCT), e.g.,

$$\mathbf{x} = (a, b, c) \quad \Rightarrow$$

$$\tilde{\mathbf{x}} = (\ldots, a, b, c, c, b, a, a, b, c, c, b, a, a, b, c, \ldots)$$

- Suppose $\mathbf{x} \in \mathbb{C}^N$ is extended periodically to $\tilde{\mathbf{x}}$
- We apply analysis filters as $\tilde{\mathbf{x}} * \mathbf{f}$, where $\mathbf{f}$ is either $\boldsymbol{\ell}_a$ or $\mathbf{h}_a$
- Key observation: if $\mathbf{f}$ is a FIR filter with at most N taps, then $\tilde{\mathbf{x}} * \mathbf{f}$ is exactly the circular convolution $\mathbf{x} * \mathbf{f}$ extended periodically
- Downsampled vectors $\tilde{\mathbf{X}}_\ell = D(\tilde{\mathbf{x}} * \boldsymbol{\ell}_a)$ and $\tilde{\mathbf{X}}_h = D(\tilde{\mathbf{x}} * \mathbf{h}_a)$ are periodic with period $N/2$
- Truncate $\tilde{\mathbf{X}}_\ell$ and $\tilde{\mathbf{X}}_h$ to $\mathbf{X}_\ell$ and $\mathbf{X}_h$ by taking components in the range $0 \leq k \leq N/2 - 1$
- Concatenate them to produce $\mathbf{X} = (\mathbf{X}_\ell, \mathbf{X}_h) \in \mathbb{C}^N$
- The mapping $\mathbf{x} \to \mathbf{X}$ is called a *discrete wavelet transform* (DWT)

- The DWT is linear
- The DWT can be represented as an $N \times N$ matrix
- If the filters are FIR filters for a perfect reconstruction filter bank, then the DWT is invertible

- $\mathbf{x} = (a, b, c, d) \in \mathbb{C}^4$
- $\boldsymbol{\ell}_a = (\frac{1}{2}, \frac{1}{2}, 0, 0)$, $\mathbf{h}_a = (\frac{1}{2}, -\frac{1}{2}, 0, 0)$
- $\mathbf{x} * \boldsymbol{\ell}_a = \frac{1}{2}(a + d, b + a, c + b, d + c)$ ext. periodically
- $\mathbf{x} * \mathbf{h}_a = \frac{1}{2}(a - d, b - a, c - b, d - c)$ ext. periodically
- Downsampling and truncating:

$$\mathbf{X}_\ell = \frac{1}{2}(a + d, c + b)$$
$$\mathbf{X}_h = \frac{1}{2}(a - d, c - b)$$

- The DWT of $\mathbf{x}$ is

$$\mathbf{X} = \frac{1}{2}(a + d, c + b, a - d, c - b)$$

Matrix view of the DWT:

$$\mathbf{X} = \mathbf{W}_4^a \mathbf{x}$$

where

$$\mathbf{W}_4^a = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 0 \end{pmatrix}$$

- The IDWT is governed by the synthesis filters
- $\boldsymbol{\ell}_s = (1, 0, 0, 1)$, $\mathbf{h}_s = (1, 0, 0, -1)$
- Let $\mathbf{X} = (A, B, C, D)$
- Upsampling gives $U(\mathbf{X}_\ell) = (A, 0, B, 0)$, $U(\mathbf{X}_h) = (C, 0, D, 0)$
- Convolving with synthesis filters:

$$U(\mathbf{X}_\ell) * \boldsymbol{\ell}_s = (A, B, B, A)$$
$$U(\mathbf{X}_h) * \mathbf{h}_s = (C, -D, D, -C)$$

- The IDWT of $\mathbf{X}$ is

$$\mathbf{x} = (A + C, B - D, B + D, A - C)$$

Matrix view of the IDWT:

$$\mathbf{x} = \mathbf{W}_4^s \mathbf{X}$$

where

$$\mathbf{W}_4^s = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \end{pmatrix}$$

Check that $\mathbf{W}_4^a \mathbf{W}_4^s = \mathbf{I}$

Changing the normalisation gives

$$\mathbf{W}_4^a = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 0 \end{pmatrix}$$

$$\mathbf{W}_4^s = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \end{pmatrix}$$

i.e., an orthogonal matrix $\mathbf{W}_4^s = (\mathbf{W}_4^a)^T$ (this is *not* the case for all wavelets)

The Discrete Wavelet Transform

Matrix Representation of Multistage DWT

- Let $\mathbf{W}_N^a, \mathbf{W}_N^s \in M_{N,N}(\mathbb{C})$ be the analysis and synthesis matrices for the N-point one-stage DWT
- Assume $N = 2^p$
- Two-stage transform has the matrix

$$\mathcal{W}_2^a = \begin{pmatrix} \mathbf{W}_{N/2}^a & 0 \\ 0 & \mathbf{I}_{N/2} \end{pmatrix} \mathbf{W}_N^a$$

  where $\mathbf{I}_{N/2}$ is the $\frac{N}{2} \times \frac{N}{2}$ identity matrix
- The inverse transform

$$\mathcal{W}_2^s = \mathbf{W}_N^s \begin{pmatrix} \mathbf{W}_{N/2}^s & 0 \\ 0 & \mathbf{I}_{N/2} \end{pmatrix}$$

- An r-stage DWT matrix obtained by iteration

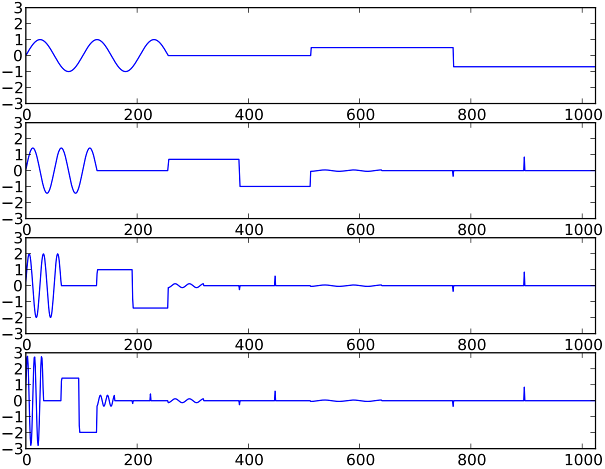
$$\mathcal{W}_r^a = \begin{pmatrix} \mathbf{W}_{N/2^{r-1}}^a & 0 \\ 0 & \mathbf{I}_{N(1-1/2^{r-1})} \end{pmatrix} \cdots \begin{pmatrix} \mathbf{W}_{N/2}^a & 0 \\ 0 & \mathbf{I}_{N/2} \end{pmatrix} \mathbf{W}_N^a$$

- An r-stage IDWT matrix

$$\mathcal{W}_r^s = \mathbf{W}_N^s \begin{pmatrix} \mathbf{W}_{N/2}^s & 0 \\ 0 & \mathbf{I}_{N/2} \end{pmatrix} \cdots \begin{pmatrix} \mathbf{W}_{N/2^{r-1}}^s & 0 \\ 0 & \mathbf{I}_{N(1-1/2^{r-1})} \end{pmatrix}$$

The Discrete Wavelet Transform

Example: Multistage Transform



- In analogy with the 2D DFT and the 2D DCT, the 2D DWT can be computed as the matrix product

$$\hat{\mathbf{A}} = \mathbf{W}_m^a \mathbf{A} (\mathbf{W}_n^a)^T$$

  where $\mathbf{W}_m^a$ and $\mathbf{W}_n^a$ are the one-dimensional DCT matrices
- The inverse transform is

$$\mathbf{A} = \mathbf{W}_m^s \hat{\mathbf{A}} (\mathbf{W}_n^s)^T$$

Interpretation of the result

$$\begin{aligned}\hat{\mathbf{A}} &= \mathbf{W}_m^a \mathbf{A} (\mathbf{W}_n^a)^T \\ &= \begin{pmatrix} \text{LL (approximation)} & \text{HL (vertical details)} \\ \text{LH (horizontal details)} & \text{HH (diagonal details)} \end{pmatrix}\end{aligned}$$

(where L represents low-pass, and H high-pass filtering)

The 2D Discrete Wavelet Transform

Example, One-Stage Transform

- The multistage 2D DWT can be computed as the matrix product

$$\hat{\mathbf{A}} = \mathcal{W}_m^a \mathbf{A} (\mathcal{W}_n^a)^T$$

  where $\mathcal{W}_m^a$ and $\mathcal{W}_n^a$ are the one-dimensional multistage DCT matrices
- The inverse transform is

$$\mathbf{A} = \mathcal{W}_m^s \hat{\mathbf{A}} (\mathcal{W}_n^s)^T$$

The 2D Discrete Wavelet Transform

Example, Multi-Stage Transform

The 2D Discrete Wavelet Transform

Example, Multi-Stage Transform

The 2D Discrete Wavelet Transform

Example, Multi-Stage Transform

The 2D Discrete Wavelet Transform

Example, Multi-Stage Transform

The 2D Discrete Wavelet Transform

Example, Multi-Stage Compression, 'haar'

The 2D Discrete Wavelet Transform

Example, Multi-Stage Compression, 'db2'

4-fold 16-fold

64-fold 256-fold

The 2D Discrete Wavelet Transform

Example, Multi-Stage Compression, 'bior3.3'

4-fold

16-fold

64-fold

256-fold

Transform	1D	2D
DFT	$\mathbf{X} = \mathbf{F}_N \mathbf{x}$	$\hat{\mathbf{A}} = \mathbf{F}_N \mathbf{A} \mathbf{F}_M^T$
DCT	$\mathbf{X} = \mathcal{C}_N \mathbf{x}$	$\hat{\mathbf{A}} = \mathcal{C}_N \mathbf{A} \mathcal{C}_M^T$
DWT	$\mathbf{X} = \mathcal{W}_N^a \mathbf{x}$	$\hat{\mathbf{A}} = \mathcal{W}_N^a \mathbf{A} {\mathcal{W}_M^a}^T$

Overview

- Review of the Haar filter bank
- The general one-stage two-channel filter bank
- Multistage filter banks
- The discrete wavelet transform, DWT
- The 2D DWT

Do exercise 6.3 in the text book