Image Processing and Analysis – Math Part: Windowing and the Haar Filter Bank

Ivar Farup

Based upon Chapter 5 and parts of Chapter 6 of Broughton and Bryan's *Discrete Fourier Analysis and Wavelets* and Maciej Piętka's Lecture Notes from 2010

Overview

- Nonlocality of the DFT
- Windowing
- The Haar filter bank

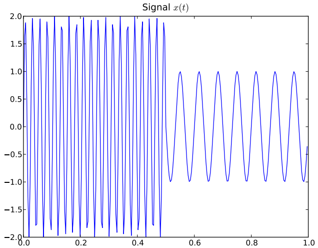
Nonlocality of the DFT

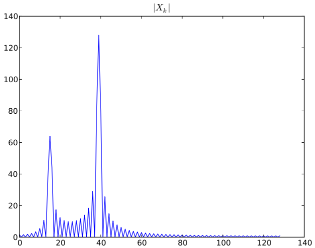
Example: Piecewise Monochromatic Signal

- Consider an audio signal on $t \in [0, 1]$:

$$x(t) = \begin{cases} 2\sin(2\pi \cdot 39t) & 0 \leq t \leq 1/2 \\ sin(2\pi \cdot 15t) & 1/2 < t \leq 1 \end{cases}$$

- Many frequency components needed to synthesize the signal
- DFT is global in nature





Nonlocality of the DFT

Example: Piecewise Monochromatic Signal

- The discrete basic waveform $\mathbf{E}_{N,k}$ is a complex exponential $e^{2\pi ikt/T}$ sampled on $t \in [0, T]$
- In a synthesized signal, it has constant amplitude $|X_k|$
- The waveform $\mathbf{E}_{256,39}$ contributes to the whole synthesized signal
- The global DFT analysis does not perform well on a non-stationary signal

Nonlocality of the DFT

Solutions

Two approaches to solve the problem:

1. Windowing and the short-time Fourier transform
2. Filter banks and the Discrete Wavelet Transform

Windowing

The Approach

- Break the signal into blocks/windows in the time domain
- Certain frequencies may be present in some blocks and not in others
- Block size small enough so that frequency content is relatively stable over the block
- Apply the DFT to each block independently
- Represent the signal as a sequence of short-time DFTs

- Starting position m
- Length M samples, $m + M \leq N$
- All samples x_j with $j < m$ and $j > m + M$ are zeroed out, others unchanged
- The resulting vector $\mathbf{y}$ has components

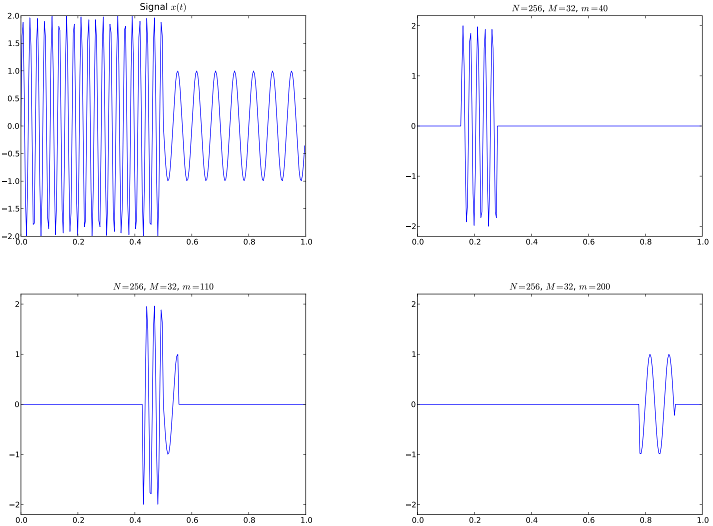
$$y_j = w_j x_h$$

where w_j are the components of the rectangular window $\mathbf{w}$,

$$w_j = \begin{cases} 1, & m \leq j \leq m + M - 1 \\ 0, & \text{otherwise} \end{cases}$$

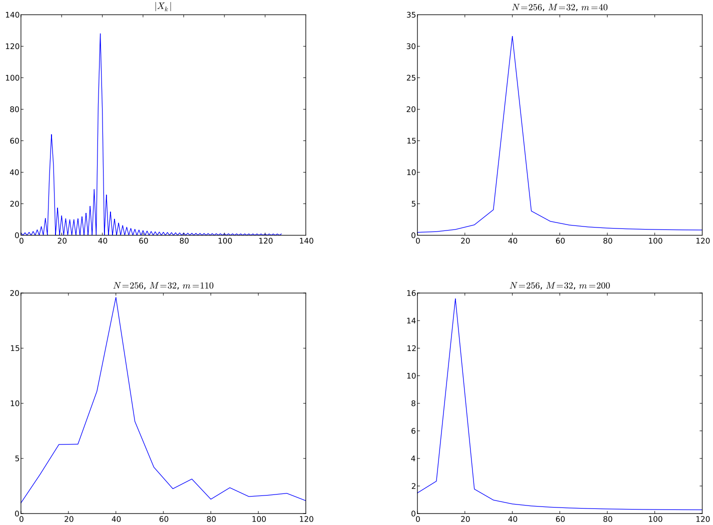
Windowing

A Rectangular Window



Windowing

A Rectangular Window



- Note that the frequency resolution of the DFT for a windowed signal is degraded
- Related to the uncertainty principle:
  - A localised signal in the time domain has a wide DFT spectrum
  - And vice versa
- With windowing, we focus on a local portion of the signal
- Thus, lose the ability to distinguish closely spaced frequencies

$$\Delta x \Delta f \geq \frac{1}{2}$$

- A collection of DFT's computed over windowed portions of the signal is called a *short-time Fourier transform*
- Adjacent windows may overlap
- Let $m = k \times n$ be the starting point of the k-th window for $k = 0, 1, \ldots$
- The integer $n \geq 1$ controls the overlap of adjacent blocks, being the discance from the start of one block to the start of the next block
- No overlap for $n = M$

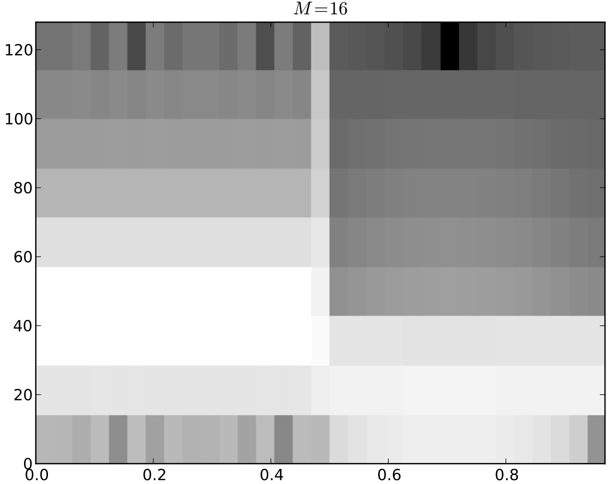
- The k-th block of data is

$$(x_{kn}, x_{kn+1}, \dots, x_{kn+M-1})$$

  for $k = 0, 1, \dots \lfloor \frac{N-M}{N} \rfloor$
- Compute the M-point DFT of each block, plot its magnitude as a k-th column of an intensity image
- The resulting plot of DFT amplitudes vs. time is called a *spectrogram*
- In the example, we take $M = 16$ and $n = 8$
- A total of 31 blocks fit into the $[0, 1]$ time interval sampled at 256 Hz

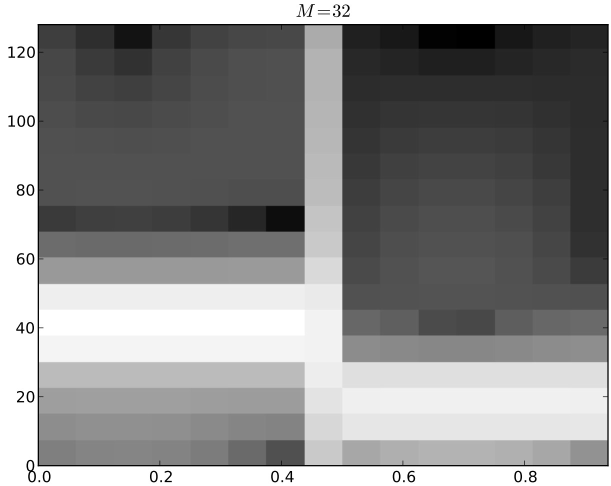
Windowing

Spectrograms



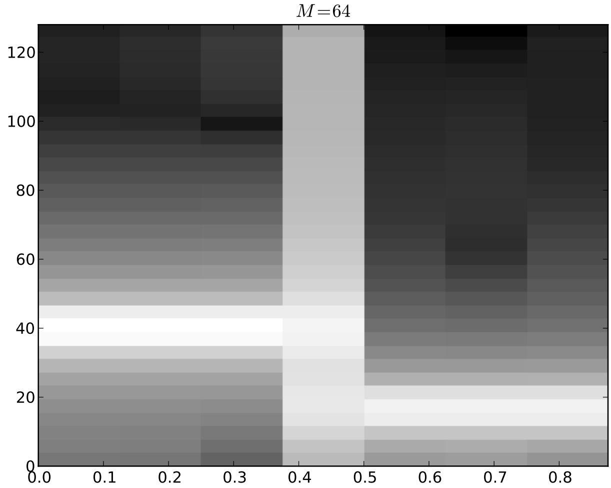
Windowing

Spectrograms



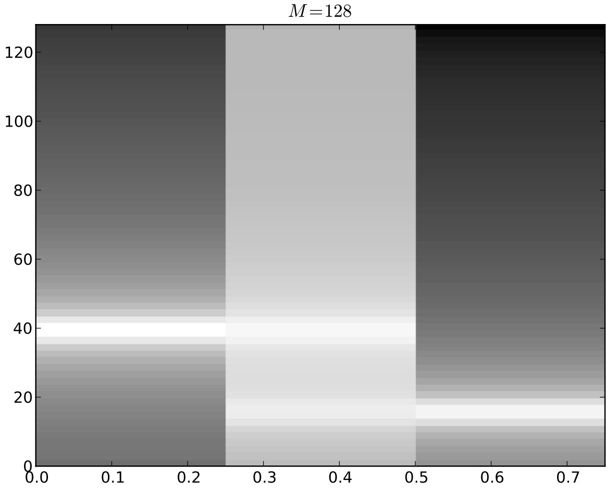
Windowing

Spectrograms



Windowing

Spectrograms



- In the *filter bank* approach, the signal is split into two or more frequency bands
- Each band is downsampled afterwards to remove redundant information
- First, consider bi-infinite sampled signals $\mathbf{x} \in L^2(\mathbb{Z})$
- x_k is defined for $k = 0, \pm 1, \pm 2, \dots, \pm\infty$
- The signal has a finite total energy,

$$\sum_{k=-\infty}^{\infty} |x_k|^2 < \infty$$

- Split the signal $\mathbf{x}$ into two bands by applying a low-pass and a high-pass filter
- Both act on $\mathbf{x}$ by convolving it with vectors $\boldsymbol{\ell}, \mathbf{h} \in L^2(\mathbb{Z})$
- Assume that both $\boldsymbol{\ell}$ and $\mathbf{h}$ have finite number of nonzero elements (*finite impulse response* (FIR) filters)
- Thus, the computation of $\mathbf{x} * \boldsymbol{\ell}$ and $\mathbf{x} * \mathbf{h}$ involves a finite sum
- Both $\mathbf{x} * \boldsymbol{\ell}$ and $\mathbf{x} * \mathbf{h}$ are in $L^2(\mathbb{Z})$: Use linearity and the triangle inequality to show that

$$||\mathbf{x} * \boldsymbol{\ell}|| \leq ||\mathbf{x}|| \sum_k |\ell_k|$$

The Haar Filter Bank

Example

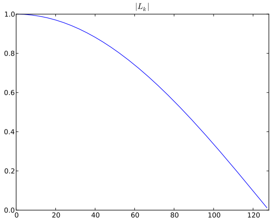
Take $\boldsymbol{\ell}$ to be the two-point averaging filter

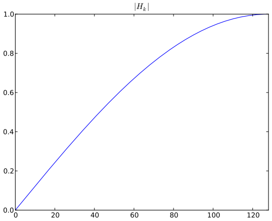
$$\ell_0 = \frac{1}{2}, \quad \ell_1 = \frac{1}{2}, \quad \ell_r = 0 \text{ otherwise}$$

and $\mathbf{h}$ to be the two-point differentiating filter

$$h_0 = \frac{1}{2}, \quad h_1 = -\frac{1}{2}, \quad h_r = 0 \text{ otherwise}$$

These are called the Haar filters

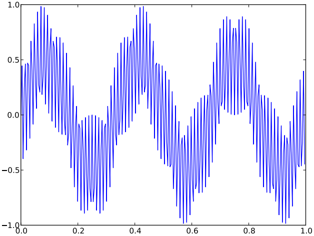


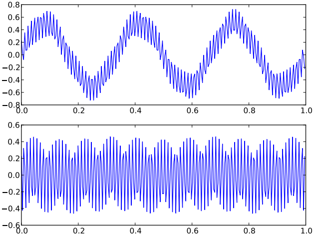


- Let $x(t)$ be the analog signal

$$x(t) = \frac{1}{2}\sin(2\pi \cdot 3t) + \frac{1}{2}\sin(2\pi \cdot 89t)$$

- Sample $x(t)$ at 256 Hz for $t \in [0, 1]$
- Pad with zeros to $\pm\infty$
- Compute $\mathbf{x} * \boldsymbol{\ell}$ and $\mathbf{x} * \mathbf{h}$





The Haar Filter Bank

$$\mathbf{x} = \begin{pmatrix} \vdots \\ x_{-2} \\ x_{-1} \\ x_0 \\ x_1 \\ x_2 \\ \vdots \end{pmatrix}, \quad \mathbf{x} * \boldsymbol{\ell} = \frac{1}{2} \begin{pmatrix} \vdots \\ x_{-2} + x_{-3} \\ x_{-1} + x_{-2} \\ x_0 + x_{-1} \\ x_1 + x_0 \\ x_2 + x_1 \\ \vdots \end{pmatrix}, \quad \mathbf{x} * \mathbf{h} = \frac{1}{2} \begin{pmatrix} \vdots \\ x_{-2} - x_{-3} \\ x_{-1} - x_{-2} \\ x_0 - x_{-1} \\ x_1 - x_0 \\ x_2 - x_1 \\ \vdots \end{pmatrix}$$

- Observe that $(\mathbf{x} * \boldsymbol{\ell} + \mathbf{x} * \mathbf{h})_k = x_k$
- and $(\mathbf{x} * \boldsymbol{\ell} - \mathbf{x} * \mathbf{h})_k = x_{k-1}$
- The transformation $\mathbf{x} \to (\mathbf{x} * \boldsymbol{\ell}, \mathbf{x} * \mathbf{h})$ is invertible
- Too much information; we can drop every other component of the filtered signal and still be able to reconstruct it

- The downsampling operator is defined as

$$(D(\mathbf{x}))_k = x_{2k}$$

- The upsampling operator is defined as

$$(U(\mathbf{x}))_k = \begin{cases} x_{k/2} & \text{when } k \text{ is even} \\ 0 & \text{when } k \text{ is odd} \end{cases}$$

- In other words:

$$\begin{aligned} \mathbf{x} &= (\dots, x_{-4}, x_{-3}, x_{-2}, x_{-1}, x_0, x_1, x_2, x_3, x_4, \dots) \\ D(\mathbf{x}) &= (\dots, x_{-8}, x_{-6}, x_{-4}, x_{-2}, x_0, x_2, x_4, x_6, x_8, \dots) \\ U(\mathbf{x}) &= (\dots, x_{-2}, 0, x_{-1}, 0, x_0, 0, x_1, 0, x_2, \dots) \end{aligned}$$

- Downsampled version of the filtered signal

$$D(\mathbf{x} * \boldsymbol{\ell}) = \frac{1}{2} \begin{pmatrix} \vdots \\ x_{-4} + x_{-5} \\ x_{-2} + x_{-3} \\ x_0 + x_{-1} \\ x_2 + x_1 \\ x_4 + x_3 \\ \vdots \end{pmatrix}, \quad D(\mathbf{x} * \mathbf{h}) = \frac{1}{2} \begin{pmatrix} \vdots \\ x_{-4} - x_{-5} \\ x_{-2} - x_{-3} \\ x_0 - x_{-1} \\ x_2 - x_1 \\ x_4 - x_3 \\ \vdots \end{pmatrix}$$

- Let $\mathbf{X}_\ell = D(\mathbf{x} * \boldsymbol{\ell}), \mathbf{X}_h = D(\mathbf{x} * \mathbf{h})$
- The invertible transform $W : L^2(\mathbb{Z}) \to L^2(\mathbb{Z}) \times L^2(\mathbb{Z})$ defined as

$$W(\mathbf{x}) = (\mathbf{X}_\ell, \mathbf{X}_h) =: \mathbf{X}$$

is called the Haar filter bank transform

The Haar Filter Bank

- The transform W is an example of an *analysis filter bank*
- The coefficients of $\mathbf{X}_\ell$ are the *approximation coefficients*
- The coefficients of $\mathbf{X}_h$ are the *detail coefficients*

$$
\begin{tikzcd}
 & {[\boldsymbol{\ell}]} \arrow[r] & {[\downarrow 2]} \arrow[r] & \mathbf{X}_\ell \arrow[dr] & \\
\mathbf{x} \arrow[ur] \arrow[dr] & & & & \mathbf{X} = (\mathbf{X}_\ell, \mathbf{X}_h) \\
 & {[\mathbf{h}]} \arrow[r] & {[\downarrow 2]} \arrow[r] & \mathbf{X}_h \arrow[ur] &
\end{tikzcd}
$$

One-stage two-channel analysis filter bank

The Haar Filter Bank

The Inverse Filter Bank Transform

1. Upsample $\mathbf{X}_\ell$ and $\mathbf{X}_h$
2. Convolve $U(\mathbf{X}_\ell)$ and $U(\mathbf{X}_h)$ with appropriate *synthesis filters*
3. Combine the resulting vector to recover $\mathbf{x}$

The transformation $\mathbf{X} = (\mathbf{X}_\ell, \mathbf{X}_h) \to \mathbf{x}$ is called the *synthesis filter bank*

The Haar Filter Bank

The Synthesis Filters

- $\boldsymbol{\ell}_s$:
$$(\boldsymbol{\ell}_s)_{-1} = 1, \quad (\boldsymbol{\ell}_s)_0 = 1, \quad (\boldsymbol{\ell}_s)_r = 0 \text{ otherwise}$$
- $\mathbf{h}_s$:
$$(\mathbf{h}_s)_{-1} = -1, \quad (\mathbf{h}_s)_0 = 1, \quad (\mathbf{h}_s)_r = 0 \text{ otherwise}$$

$$\mathbf{x}_\ell = \frac{1}{2} \begin{pmatrix} \vdots \\ x_{-4} + x_{-5} \\ x_{-2} + x_{-3} \\ x_0 + x_{-1} \\ x_2 + x_1 \\ x_4 + x_3 \\ \vdots \end{pmatrix}, \quad \mathbf{x}_h = \frac{1}{2} \begin{pmatrix} \vdots \\ x_{-4} - x_{-5} \\ x_{-2} - x_{-3} \\ x_0 - x_{-1} \\ x_2 - x_1 \\ x_4 - x_3 \\ \vdots \end{pmatrix}$$

The Haar Filter Bank

Upsampling

$$U(\mathbf{X}_\ell) = \frac{1}{2}\begin{pmatrix} \vdots \\ 0 \\ x_{-2} + x_{-3} \\ 0 \\ x_0 + x_{-1} \\ 0 \\ x_2 + x_1 \\ 0 \\ \vdots \end{pmatrix}, \quad U(\mathbf{X}_h) = \frac{1}{2}\begin{pmatrix} \vdots \\ 0 \\ x_{-2} - x_{-3} \\ 0 \\ x_0 - x_{-1} \\ 0 \\ x_2 - x_1 \\ 0 \\ \vdots \end{pmatrix}$$

$$\mathbf{v}_\ell = U(\mathbf{X}_\ell)*\boldsymbol{\ell} = \frac{1}{2}\begin{pmatrix} \vdots \\ x_{-2}+x_{-3} \\ x_{-2}+x_{-3} \\ x_0+x_{-1} \\ x_0+x_{-1} \\ x_2+x_1 \\ x_2+x_1 \\ \vdots \end{pmatrix}, \quad \mathbf{v}_h = U(\mathbf{X}_h)*\mathbf{h} = \frac{1}{2}\begin{pmatrix} \vdots \\ x_{-3}-x_{-2} \\ x_{-2}-x_{-3} \\ x_{-1}-x_0 \\ x_0-x_{-1} \\ x_1-x_2 \\ x_2-x_1 \\ \vdots \end{pmatrix}$$

Note that $\mathbf{v}_\ell + \mathbf{v}_h = \mathbf{x}$

The Haar Filter Bank

Analysis Filter Bank

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

The Haar Filter Bank

Synthesis Filter Bank

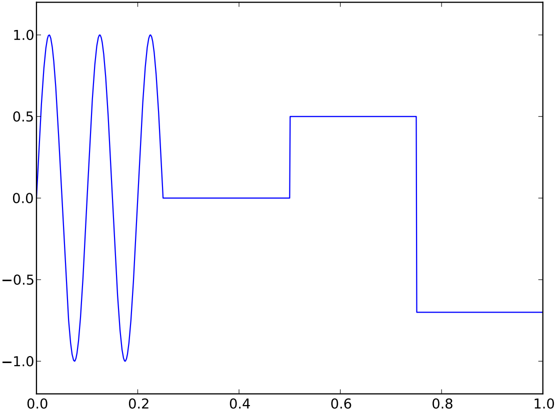
$$
\begin{tikzcd}
 & \mathbf{X}_l \arrow[r] & {[\uparrow 2]} \arrow[r] & {[\boldsymbol{\ell}_s]} \arrow[dr] & \\
\mathbf{X} \arrow[ur] \arrow[dr] & & & & \mathbf{x} \\
 & \mathbf{X}_h \arrow[r] & {[\uparrow 2]} \arrow[r] & {[\mathbf{h}_s]} \arrow[ur] &
\end{tikzcd}
$$

The Haar Filter Bank

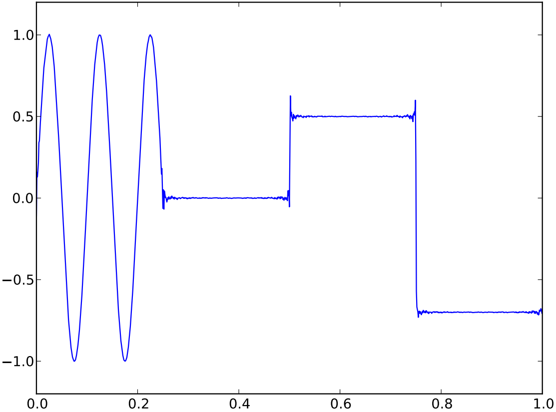
Analysis/Synthesis Filter Bank

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

Signal to be compressed

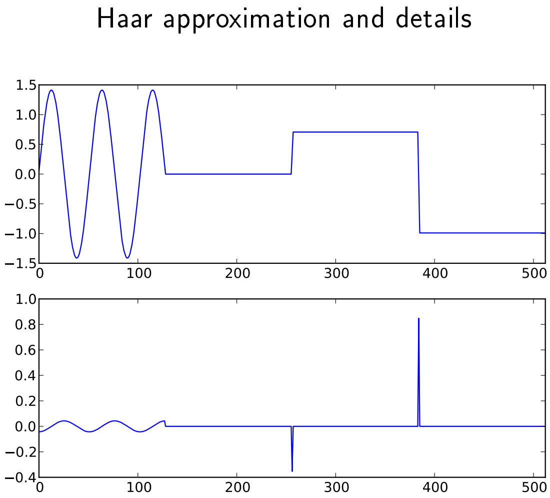


DFT compression (half of the coefficients)



The Haar Filter Bank

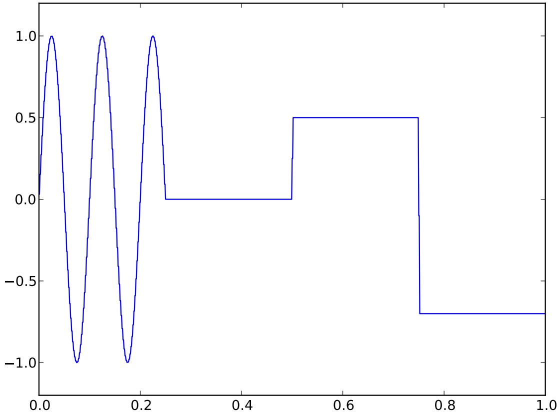
Example



The Haar Filter Bank

Example

Haar compression (remove details)



- The filter bank transform is local in nature. Each approximation and detail coefficient depends on few neighboring samples
- The DFT is global in nature. Each DFT coefficient depends on all samples

Overview

- Nonlocality of the DFT
- Windowing
- The Haar filter bank

Do exercise 5.1 in the text book