Introduction to Wavelets
Discrete Wavelet Transform

- A wavelet is a function of zero average centered in the neighborhood of $t=0$ and is normalized:
  \[
  \int_{-\infty}^{+\infty} \psi(t)dt = 0
  \]
  \[
  \|\psi\| = 1
  \]

- The translations and dilations of the wavelet generate a family of functions over which the signal is projected:
  \[
  \psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right)
  \]

- Wavelet transform of $f$ in $L^2(\mathbb{R})$ at position $u$ and scale $s$ is:
  \[
  Wf(u,s) = \langle f, \psi_{u,s} \rangle = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}}\psi^*\left(\frac{t-u}{s}\right)dt
  \]
  \[
  s = 2^j
  \]
  \[
  u = k \cdot 2^j
  \]
Wavelet transform

\[ \Psi_{0,s}(t) \]

\[ \Psi_{u,s}(t) \]

\[ Wf(0,s) \Leftrightarrow \text{correlation for } u=0 \]
Wavelet transform

\[ \Psi_{n2^j,s}(t) \]

\[ Wf(n 2^i, s) \leftrightarrow \text{correlation for } u=n 2^i \]
Wavelet transform

\[ \Psi_{(n+1)2^j,s}(t) \]

\[ Wf((n+1)2^j,s) \Leftrightarrow \text{correlation at } u=(n+1)2^j \]
Changing the scale

\[ \Psi_{u,s}(t) \]

\[ s = 2^{j+1} \]

multiresolution

\[ \Psi_{u,s}(t) \]

\[ s = 2^{j+2} \]

coarser
Fourier versus Wavelets

\[ F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} \, dt \]

\[ C(\text{scale, position}) = \int_{-\infty}^{\infty} f(t)\psi(\text{scale, position, } t) \, dt \]

- Sine Wave
- Wavelet (db10)

Wavelet Transform

Signal

Constituent wavelets of different scales and positions
Scaling

\[ f(t) = \psi(t) \quad ; \quad a = 1 \]

\[ f(t) = \psi(2t) \quad ; \quad a = \frac{1}{2} \]

\[ f(t) = \psi(4t) \quad ; \quad a = \frac{1}{4} \]
Shifting

Wavelet function $\psi(t)$

Shifted wavelet function $\psi(t - k)$
Recipe

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. More precisely, if the signal energy and the wavelet energy are equal to one, $C$ may be interpreted as a correlation coefficient.

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

\[ C = 0.0102 \]
Recipe

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.

5 Repeat steps 1 through 4 for all scales.
Wavelet Zoom

- WT at position $u$ and scale $s$ measures the local correlation between the signal and the wavelet

Thus, there is a correspondence between wavelet scales and frequency as revealed by wavelet analysis:

(small) • Low scale $a \Rightarrow$ Compressed wavelet $\Rightarrow$ Rapidly changing details $\Rightarrow$ High frequency $\omega$.

(large) • High scale $a \Rightarrow$ Stretched wavelet $\Rightarrow$ Slowly changing, coarse features $\Rightarrow$ Low frequency $\omega$. 
Frequency domain

- Parseval

\[
Wf(u,s) = \int_{-\infty}^{+\infty} f(t)\psi_{u,s}^*(t)dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega)\Psi_{u,s}^*(\omega)d\omega
\]

The wavelet coefficients \( Wf(u,s) \) depend on the values of \( f(t) \) (and \( F(\omega) \)) in the time-frequency region where the energy of the corresponding wavelet function (respectively, its transform) is concentrated

- time/frequency localization
- The position and scale of high amplitude coefficients allow to characterize the temporal evolution of the signal

- Time domain signals (1D) : Temporal evolution
- Spatial domain signals (2D) : Localize and characterize spatial singularities

\[
\psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right) \Leftrightarrow \Psi_{u,s}(\omega) = \sqrt{s}\Psi(s\omega)e^{-j\omega s}
\]

Stratching in time ↔ Shrinking in frequency (and viceversa)
Wavelet representation = approximation + details

approximation ↔ scaling function
details ↔ wavelets

Example
A different perspective

\[ A_{2^0}^d f = A_{2^1}^d f + d_{2^1} \]

approximation at resolution \(2^0\)

approximation at resolution \(2^1\)
detail signal
Haar pyramid [Haar 1910]

Haar basis function

$\varphi_2^0$

Haar wavelet

details

$\text{signal} = \text{approximation at scale } n + \text{details at scales 1 to } n$
What wavelets can do?
Wavelets and linear filtering

- The WT can be rewritten as a convolution product and thus the transform can be interpreted as a linear filtering operation:

\[
Wf(u,s) = \langle f, \psi_{u,s} \rangle = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}} \psi^* \left( \frac{t-u}{s} \right) dt = f * \overline{\psi}_s(u)
\]

\[
\overline{\psi}_s(t) = \frac{1}{\sqrt{s}} \psi^* \left( \frac{-t}{s} \right)
\]

\[
\widehat{\psi}_s(\omega) = \sqrt{s} \psi^* (s\omega)
\]

\[
\hat{\psi}(0) = 0
\]

→ band-pass filter
Wavelets & filterbanks

Quadrature Mirror Filter (QMF)
Analysis or decomposition

- Filters
  - low-pass
  - high-pass

- S ➔ A
- S ➔ D

- 1000 samples ➔ D ➔ ~1000 samples
- 1000 samples ➔ A ➔ ~1000 samples

- S ➔ cD ➔ ~500 coeffs
- S ➔ cA ➔ ~500 coeffs
Analysis or decomposition

Diagram showing the analysis or decomposition process with a tree structure and graphs illustrating the components cA_1, cA_2, cA_3, cD_1, cD_2, and cD_3.
Synthesis or reconstruction

\[ S = A_1 + D_1 \]
\[ = A_2 + D_2 + D_1 \]
\[ = A_3 + D_3 + D_2 + D_1 \]

Upsampling
Multi-scale analysis

Analysis
Decomposition
DWT

Wavelet
Coefficients

Synthesis
Reconstruction
IDWT

1000

H
L

H
L

H
L

H
L

~500

~250
Famous wavelets

- **Haar**

- **Mexican hat**
Daubechies’s

db2  db3  db4  db5  db6

db7  db8  db9  db10
Bi-dimensional wavelets

\[
\varphi(x, y) = \varphi(x)\varphi(y) \\
\psi^1(x, y) = \varphi(x)\psi(y) \\
\psi^2(x, y) = \psi(x)\varphi(y) \\
\psi^3(x, y) = \psi(x)\psi(y)
\]

\[
\frac{1}{\sqrt{a_1 a_2}} \psi\left(\frac{x_1 - b_1}{a_1}, \frac{x_2 - b_2}{a_2}\right) \text{ where } (x = (x_1, x_2) \in \mathbb{R}^2)
\]
Fast wavelet transform algorithm (DWT)

Decomposition step

where $X$ Convolve with filter $X$.

Keep the even indexed elements (see dyaddown).
Fast wavelet transform algorithm (DWT)

Reconstruction Step

- **cA_j**
  - **upsample**
  - **2**
  - **low-pass**
  - **Lo_R**
  - **wkeep**
  - **cA_{j-1}**

- **cD_j**
  - **upsample**
  - **2**
  - **high-pass**
  - **Hi_R**
  - **level j**

where:
- **2**
  - Insert zeros at odd-indexed elements.
- **X**
  - Convolve with filter X.
- **wkeep**
  - Take the central part of U with the convenient length.
Filters
Fast DWT for images

Decomposition Step

where

- \( \begin{array}{c} 2 \downarrow 1 \\ 1 \downarrow 2 \end{array} \) - Downsample rows: keep the even indexed rows.
- \( \begin{array}{c} \text{rows} \\ \text{columns} \end{array} \) - Convolve with filter X the rows of the entry.
- \( \begin{array}{c} \text{columns} \\ \text{columns} \end{array} \) - Convolve with filter X the columns of the entry.

Initialization \( CA_0 = s \) for the decomposition initialization.
Fast DWT for images

Reconstruction Step

Two-Dimensional IDWT

\( cA_{j+1} \)

\( cD_{j+1}^{(h)} \)

\( cD_{j+1}^{(v)} \)

\( cD_{j+1}^{(d)} \)

\( \times \)

\( \times \)

where

\( \underline{2 \uparrow 1} \)

Upsample columns: insert zeros at odd-indexed columns.

\( \underline{1 \uparrow 2} \)

Upsample rows: insert zeros at odd-indexed rows.

\( \underline{\times} \)

Convolve with filter X the rows of the entry.

\( \underline{\times} \)

Convolve with filter X the columns of the entry.
Subband structure for images

\[ s \leftarrow cA_1 \leftarrow cA_2 \leftarrow cD_1^{(h)} \leftarrow cD_1^{(d)} \leftarrow cD_1^{(v)} \]

\[ \leftarrow cA_2 \leftarrow cD_2^{(h)} \leftarrow cD_2^{(d)} \leftarrow cD_2^{(v)} \]

\[ \leftarrow cA_2 \leftarrow cD_2^{(h)} \leftarrow cD_2^{(d)} \leftarrow cD_2^{(v)} \]