Wavelet Packets

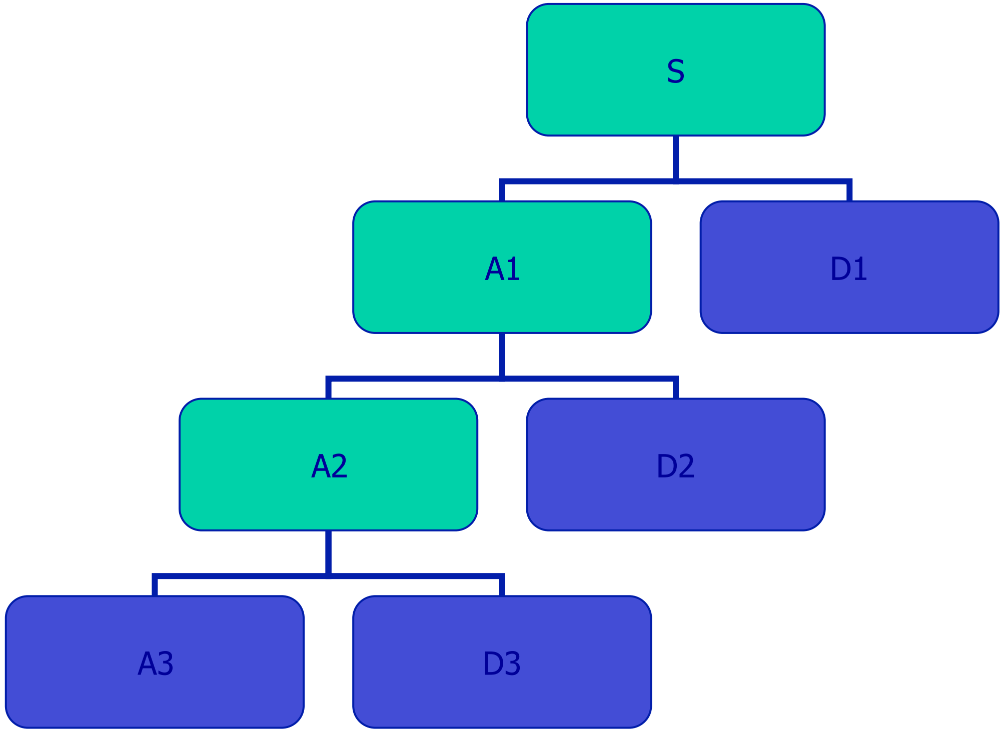
M2009, Chapter 8

Motivation

- Goal
  - Get minimal representation of data relative to particular cost function
- Usage
  - Data compression
  - Noise reduction

Wavelet Transform

- Wavelet transform is applied to low pass results (approximations) only:

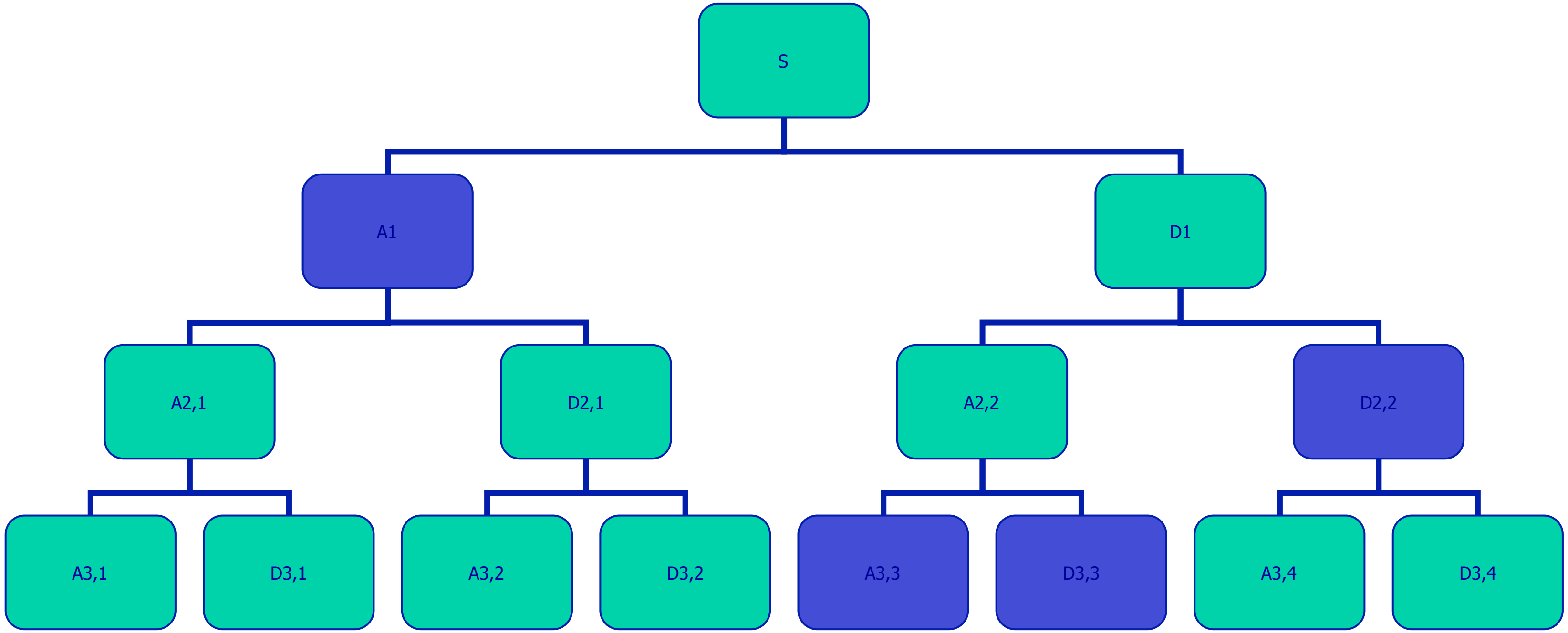


Not optimal

- From the point of view of compression, where we want as many small values as possible, the standard wavelet transform may not produce the best result, since it is limited to wavelet bases that are delated by a power of two with each step.
- It could be that another combination of functions produce a more desirable representation.

Wavelet Packet Transform

Wavelet packet transform is applied to both low pass results (approximations) and high pass results (details)


S

A1

D1

A2,1

D2,1

A2,2

D2,2

A3,1

D3,1

A3,2

D3,2

A3,3

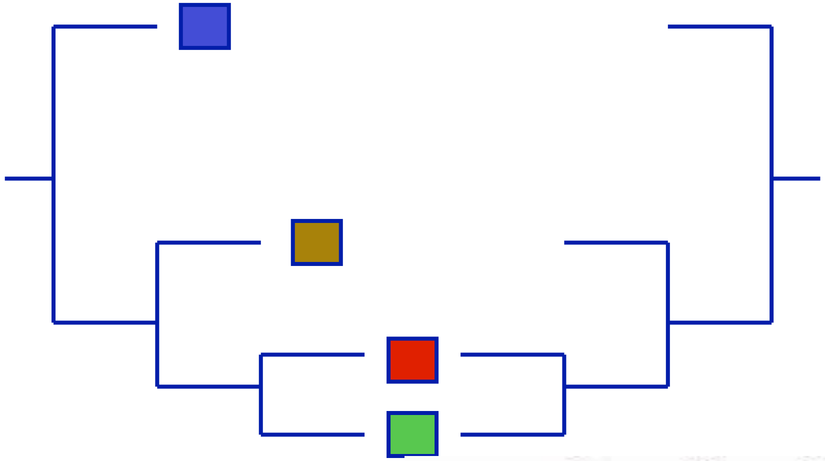
D3,3

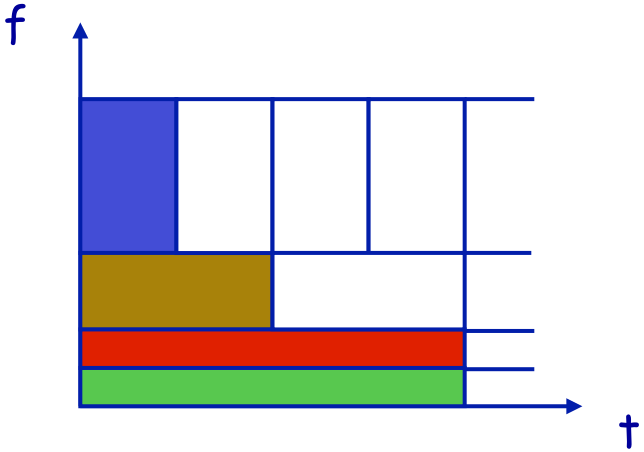
A3,4

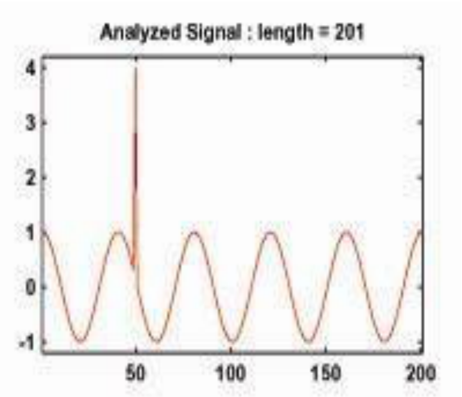
D3,4

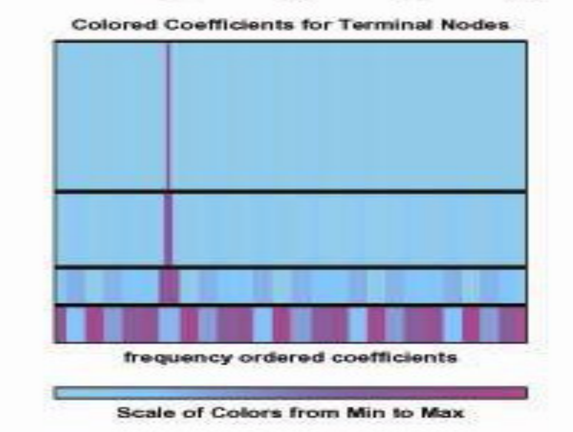
DWT

- Iterate only on the lowpass channel

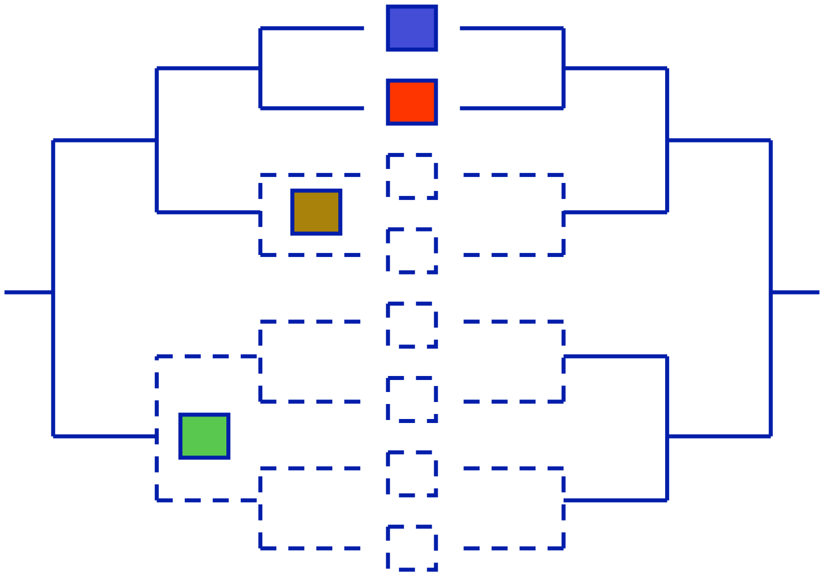


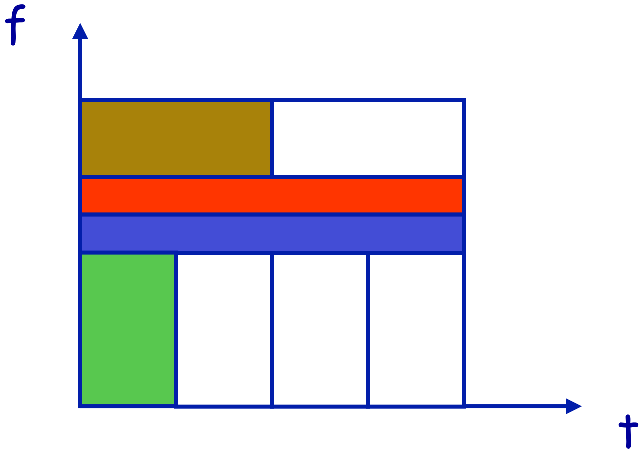






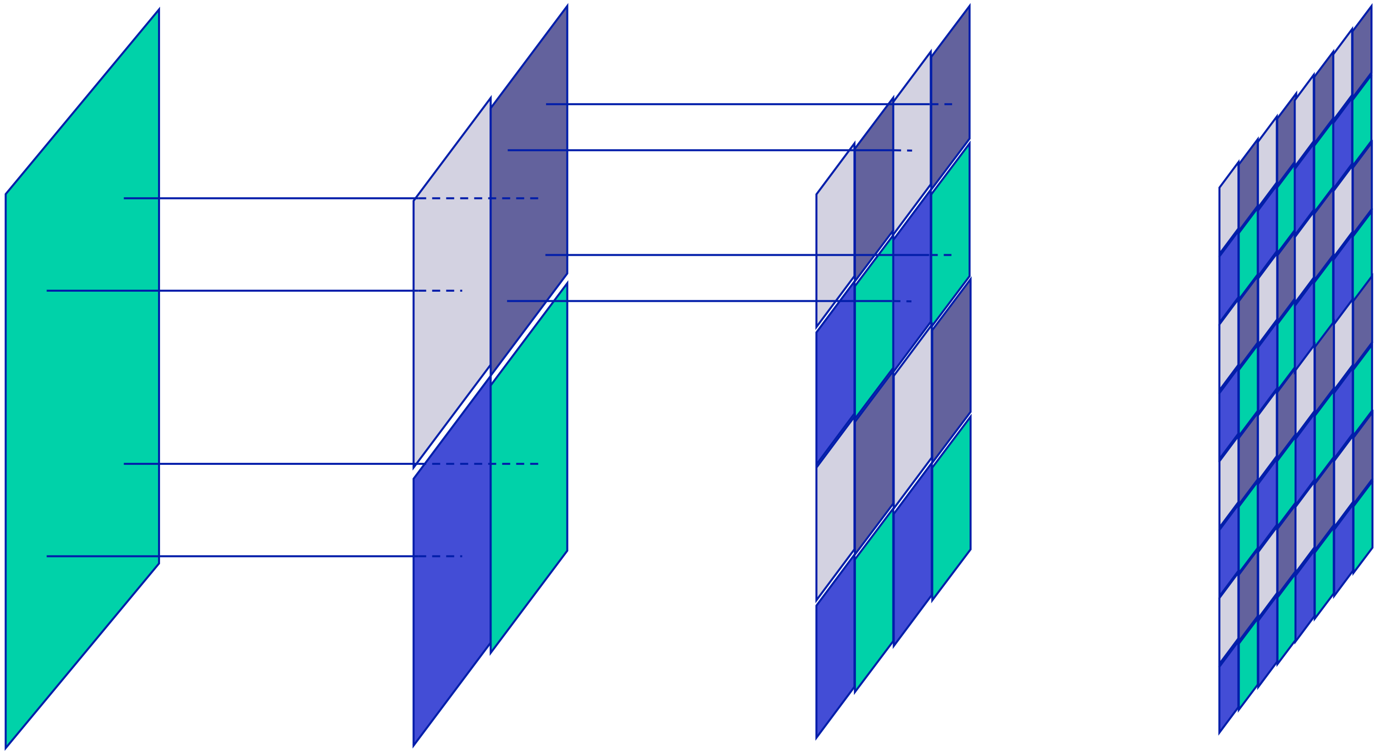
wavelet packet





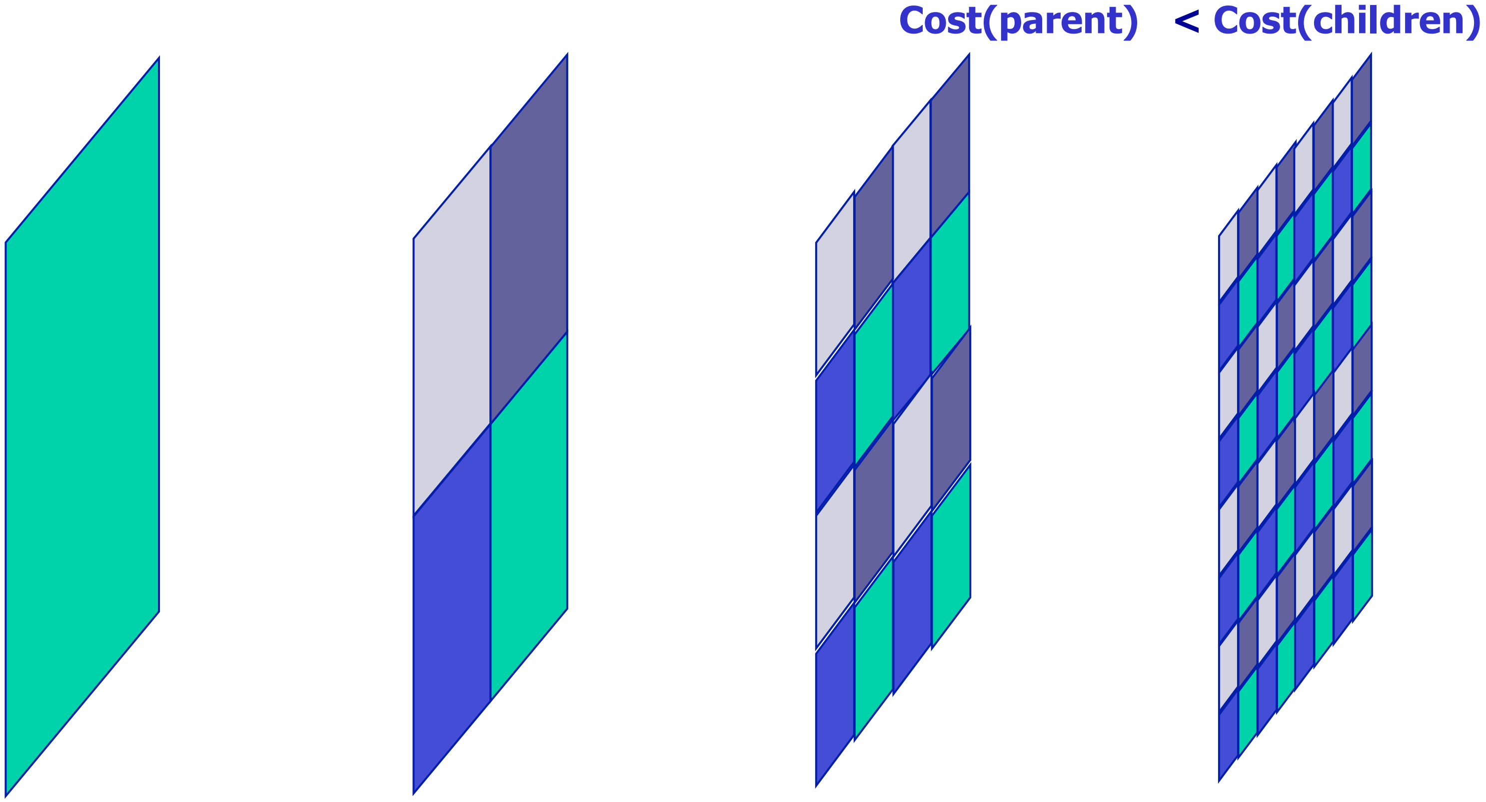
wavelet packet

- First stage: full decomposition



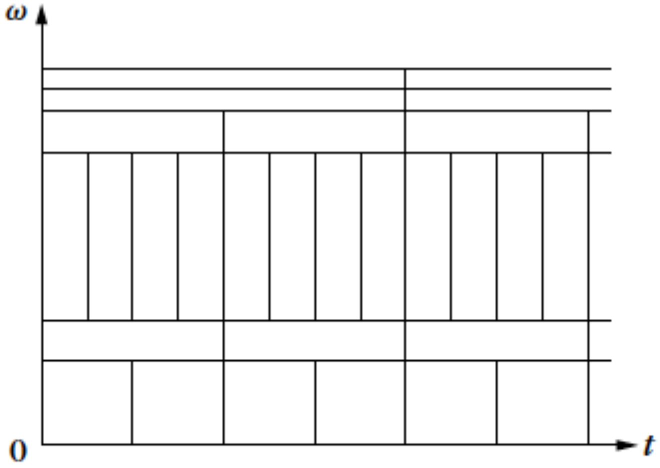
wavelet packet

- Second stage: pruning



Wavelet packets

- Wavelet bases divide the frequency axis into intervals of 1 octave bandwidth. Coifman, Meyer, and Wickerhauser have generalized this construction with bases that split the frequency axis in intervals of bandwidth that may be adjusted.
- Each frequency interval is covered by the Heisenberg time-frequency boxes of wavelet packet functions *translated in time*, in order to cover the whole plane, as shown by Figure 1.7



A wavelet packet basis divides the frequency axis in separate intervals of varying sizes. A tiling is obtained by translating in time the wavelet packets covering each frequency interval.

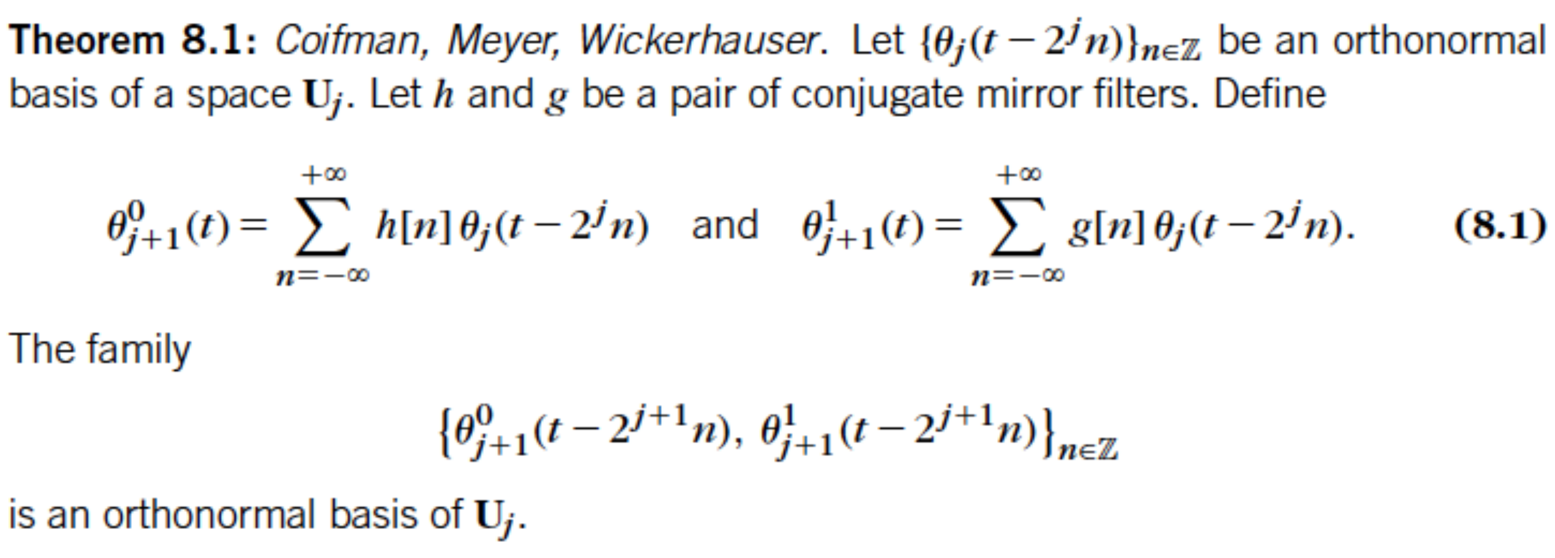
Wavelet packets (Chapt. 9, M2009)

- Wavelet packets were introduced by Coifman, Meyer, and Wickerhauser [182] by generalizing the link between multiresolution approximations and wavelets.
- A space Vj of a multiresolution approximation is decomposed in a lower-resolution space Vj+1 plus a detail space Wj+1 . This is done by dividing the orthogonal basis
- $\{\varphi j(t-2^{j}n)\}n\in Z$ of Vj into two new orthogonal bases
- $\{\varphi j+1(t-2^{j+1}n)\}n\in Z$ of Vj+1 and $\{\psi j+1(t-2^{j+1}n)\}n\in Z$ of Wj+1.
- The decomposition is specified by a pair of CMF h[n] and g[n]

$$g[n]=(-1)^{1-n}\, h[1-n]$$

- Theorem 8.1 generalizes this result to any space Uj that admits an orthogonal basis of functions translated by $n2^{j}$ for $n\in Z$.

Wavelet packets



Theorem 8.1: *Coifman, Meyer, Wickerhauser.* Let $\{\theta_j(t-2^j n)\}_{n\in\mathbb{Z}}$ be an orthonormal basis of a space $\mathbf{U}_j$. Let h and g be a pair of conjugate mirror filters. Define

$$\theta^0_{j+1}(t)=\sum_{n=-\infty}^{+\infty} h[n]\,\theta_j(t-2^j n) \quad \text{and} \quad \theta^1_{j+1}(t)=\sum_{n=-\infty}^{+\infty} g[n]\,\theta_j(t-2^j n). \qquad \textbf{(8.1)}$$

The family

$$\{\theta^0_{j+1}(t-2^{j+1}n),\ \theta^1_{j+1}(t-2^{j+1}n)\}_{n\in\mathbb{Z}}$$

is an orthonormal basis of $\mathbf{U}_j$.

Wavelet packets

Theorem 8.1 proves that conjugate mirror filters transform an orthogonal basis $\{\theta_j(t-2^j n)\}_{n\in\mathbb{Z}}$ in two orthogonal families $\{\theta^0_{j+1}(t-2^{j+1}n)\}_{n\in\mathbb{Z}}$ and $\{\theta^1_{j+1}(t-2^{j+1}n)\}_{n\in\mathbb{Z}}$. Let $\mathbf{U}^0_{j+1}$ and $\mathbf{U}^1_{j+1}$ be the spaces generated by each of these families. Clearly $\mathbf{U}^0_{j+1}$ and $\mathbf{U}^1_{j+1}$ are orthogonal and

$$\mathbf{U}^0_{j+1} \oplus \mathbf{U}^1_{j+1} = \mathbf{U}_j.$$

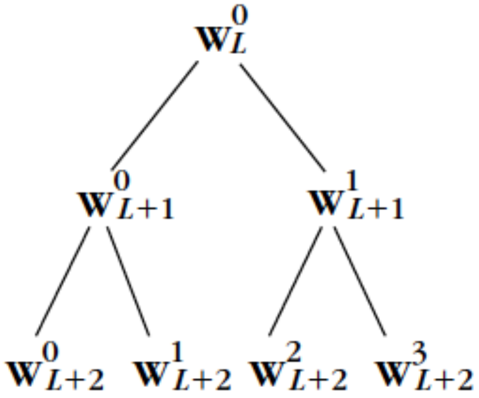
Computing the Fourier transform of (8.1) relates the Fourier transforms of θ^0_{j+1} and θ^1_{j+1} to the Fourier transform of θ_j:

$$\hat{\theta}^0_{j+1}(\omega) = \hat{h}(2^j\omega)\,\hat{\theta}_j(\omega), \quad \hat{\theta}^1_{j+1}(\omega) = \hat{g}(2^j\omega)\,\hat{\theta}_j(\omega). \tag{8.9}$$

Since the transfer functions $\hat{h}(2^j\omega)$ and $\hat{g}(2^j\omega)$ have their energy concentrated in different frequency intervals, this transformation can be interpreted as a division of the frequency support of $\hat{\theta}_j$.

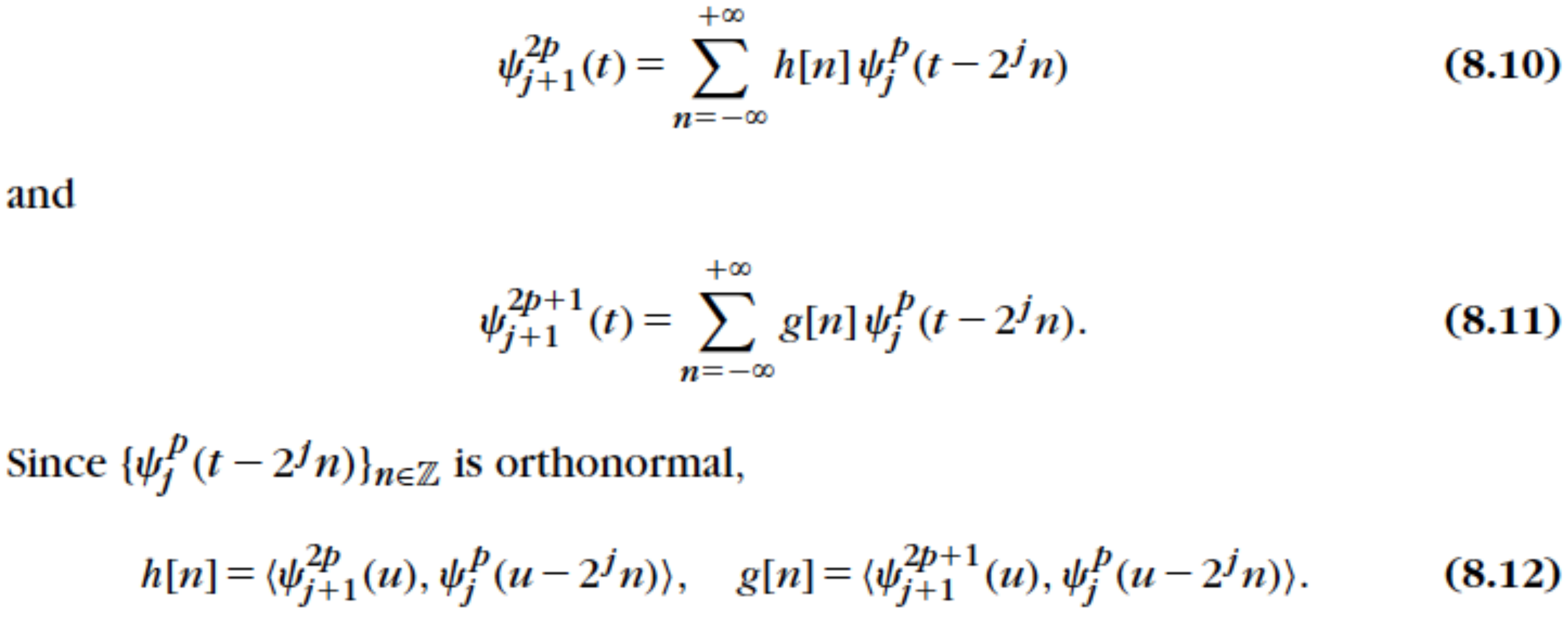
Binary wavelet packet tree

- Instead of dividing only the approximation spaces Vj to construct detail spaces Wj and wavelet bases, Theorem 8.1 proves that we can set Uj=Wj and divide these detail spaces to derive new bases.
- The recursive splitting of vector spaces is represented in a binary tree.
- Any node of the binary tree is labeled by (j, p) , where $j-L \leq 0$ is the depth of the node in the tree, and p is the number of nodes that are on its left at the same depth j-L



Binary wavelet packet tree

- Wavelet packet orthogonal bases at the children node



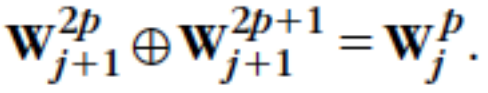
$$\psi_{j+1}^{2p}(t) = \sum_{n=-\infty}^{+\infty} h[n]\, \psi_j^p(t - 2^j n) \qquad \textbf{(8.10)}$$

and

$$\psi_{j+1}^{2p+1}(t) = \sum_{n=-\infty}^{+\infty} g[n]\, \psi_j^p(t - 2^j n). \qquad \textbf{(8.11)}$$

Since $\{\psi_j^p(t - 2^j n)\}_{n \in \mathbb{Z}}$ is orthonormal,

$$h[n] = \langle \psi_{j+1}^{2p}(u), \psi_j^p(u - 2^j n) \rangle, \quad g[n] = \langle \psi_{j+1}^{2p+1}(u), \psi_j^p(u - 2^j n) \rangle. \qquad \textbf{(8.12)}$$



$$\mathbf{W}_{j+1}^{2p} \oplus \mathbf{W}_{j+1}^{2p+1} = \mathbf{W}_j^p .$$

Example

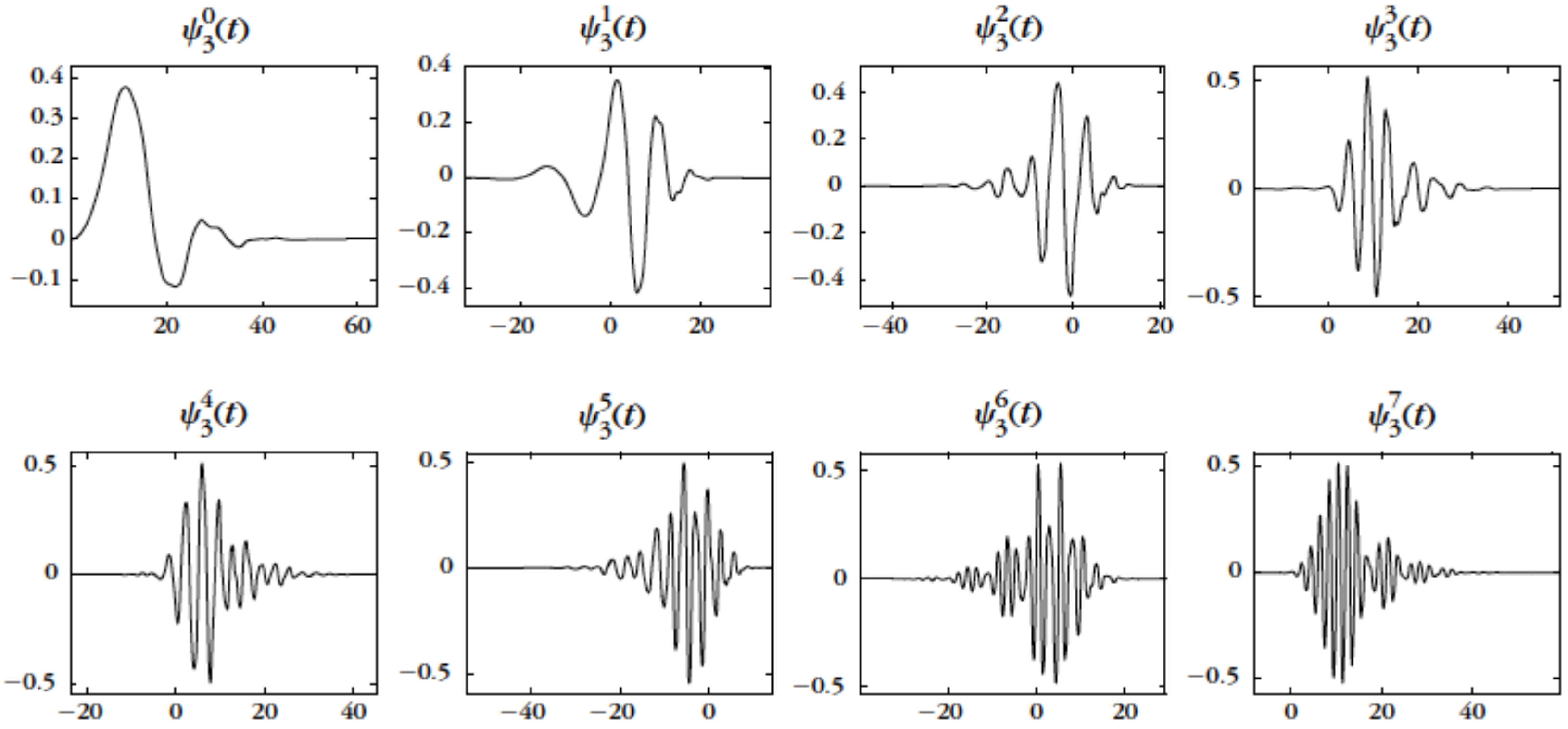
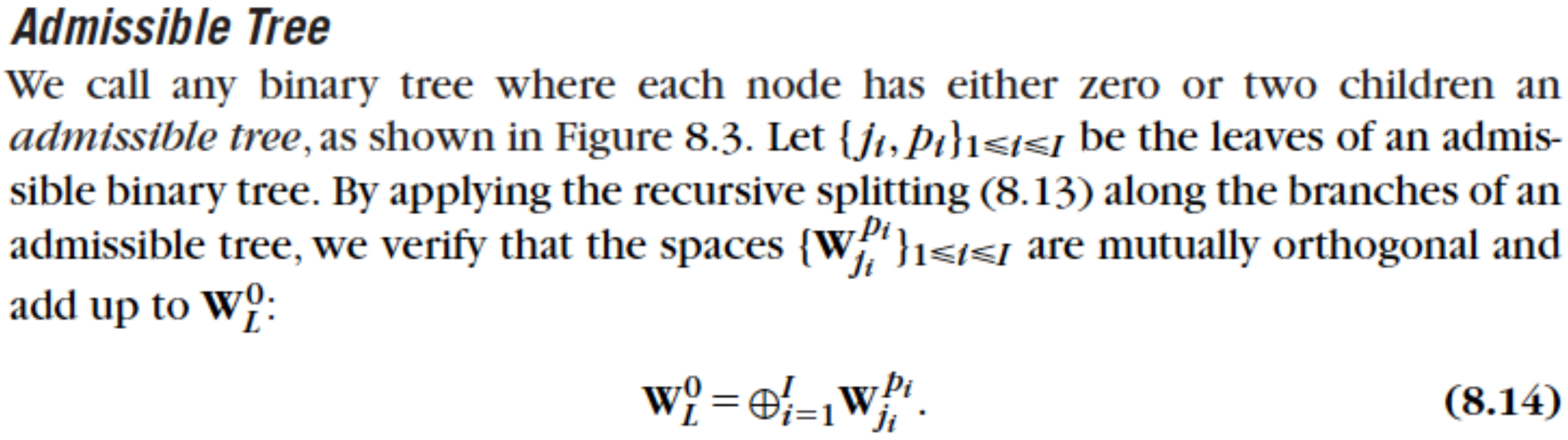


FIGURE 8.2

Wavelet packets computed with a Daubechies 5 filter at the depth $j-L=3$ of the wavelet packet tree, with $L=0$. They are ordered from low to high frequencies.

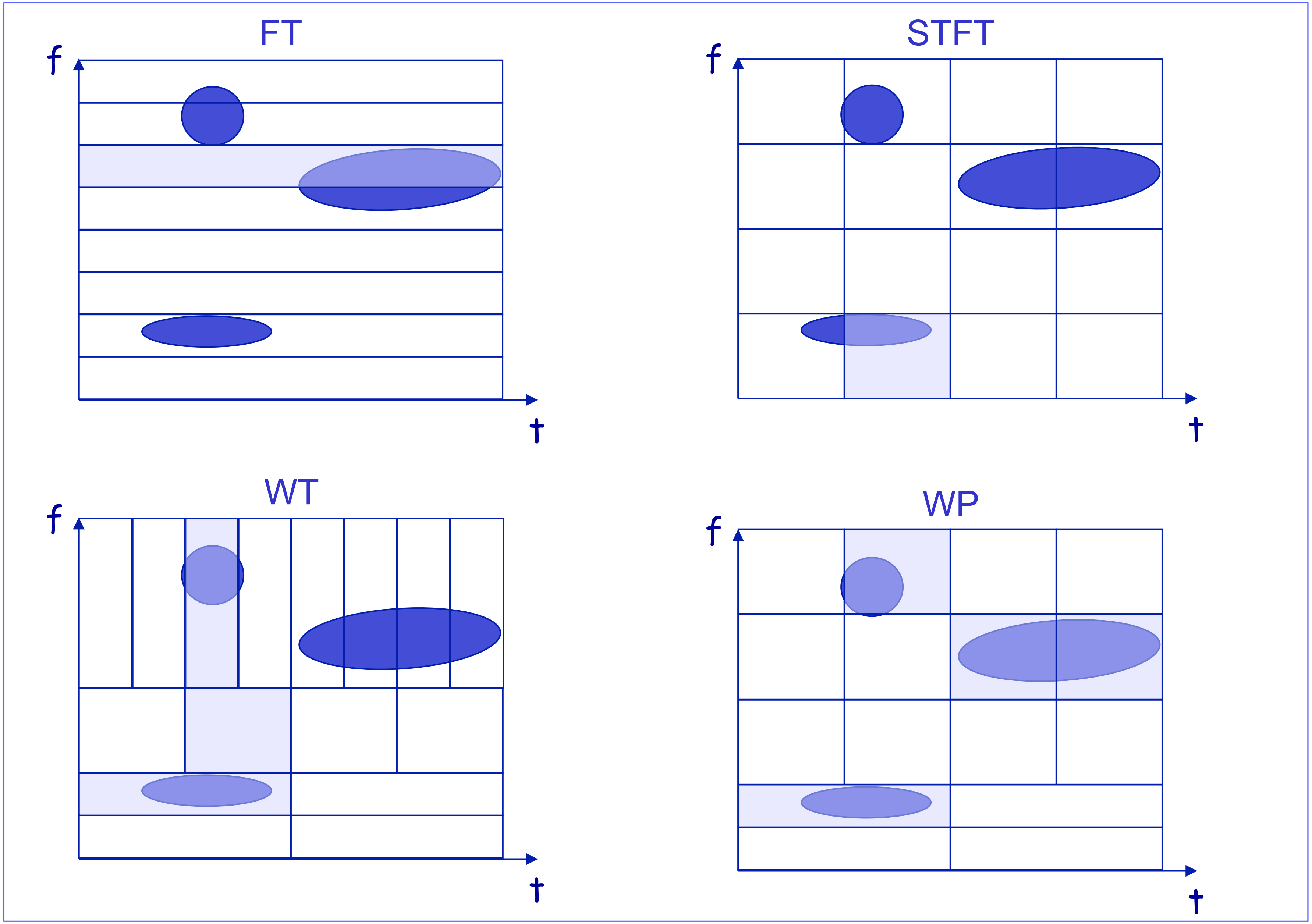
Admissible tree



Admissible Tree

We call any binary tree where each node has either zero or two children an *admissible tree*, as shown in Figure 8.3. Let $\{j_i, p_i\}_{1\leqslant i\leqslant I}$ be the leaves of an admissible binary tree. By applying the recursive splitting (8.13) along the branches of an admissible tree, we verify that the spaces $\{\mathbf{W}_{j_i}^{p_i}\}_{1\leqslant i\leqslant I}$ are mutually orthogonal and add up to $\mathbf{W}_L^0$:

$$\mathbf{W}_L^0 = \oplus_{i=1}^{I} \mathbf{W}_{j_i}^{p_i}. \tag{8.14}$$


FT
f
t
STFT
f
t
WT
f
t
WP
f
t

Best basis

- Among the admissible trees, one can select the “best one” with respect to a predefined criterion (cost function)
- The best basis pursuit algorithm finds a set of wavelet bases basically prunes the complete tree under the guidance of the cost function
- A cost function may be chosen to fit a particular application.
  - For example, in a compression algorithm the cost function might be the number of bits needed to represent the result.

Cost function

- The value of the cost function is a real number.
- Given two vectors of finite length, **a** and **b**, we denote their concatenation by [**a** **b**]. This vector simply consists of the elements in **a** followed by the elements in **b**.
- We require the following two properties:
  - The cost function is additive in the sense that $K([\mathbf{a}\ \mathbf{b}]) = K(\mathbf{a}) + K(\mathbf{b})$ for all finite length vectors **a** and **b**.
  - $K(\mathbf{0}) = 0$, where **0** denotes the zero vector

Cost functions: threshold

- The threshold cost function counts the number of *values* in a wavelet packet tree node whose absolute value is greater than a threshold value ***t***.

$$\text{cost}_{\text{threshold}} = \sum_{i=0}^{N-1} (|s[i]| > t) \; ? \; 1 : 0;$$

➔ Promoting sparsity!

Best basis algorithm

- Compute cost value for each node
- When the wavelet packet tree is constructed, all the leaves are marked with a flag. The best basis calculation is performed bottom up (that is, from the leaves of the tree toward the root):
  - A leaf (a node at the bottom of the tree with no children) returns its cost value.
  - As the calculation recurses up the tree toward the root, if there is a non-leaf node, **v1** is the cost value for that node. The value **v2** is the sum of the cost values of the children of the node.
    - If (**v1** <= **v2**) then we mark the node as part of the best basis set and remove any marks in the nodes in the sub-tree of the current node.
    - If (**v1** > **v2**) then the cost value of the node is replaced with v2.

Best basis contd.

- The best basis set selected by the best basis algorithm is relative to a particular cost function.
- In some cases the best basis set may be the same set yielded by the wavelet transform (in which case we could have used the simpler algorithm).
- In other cases the best basis function may not yield a result that differs from the original data set (e.g., the original data set is already a minimal representation in terms of the cost function).

Other cost functions

- Nonnormalized Shannon (0 log(0)=0)

$$\mathrm{cost}_{shannon} = -\sum_{n} s[n]^2 \log(s[n]^2)$$

  - The Shannon entropy function provides a measure of the economy of representation of a signal

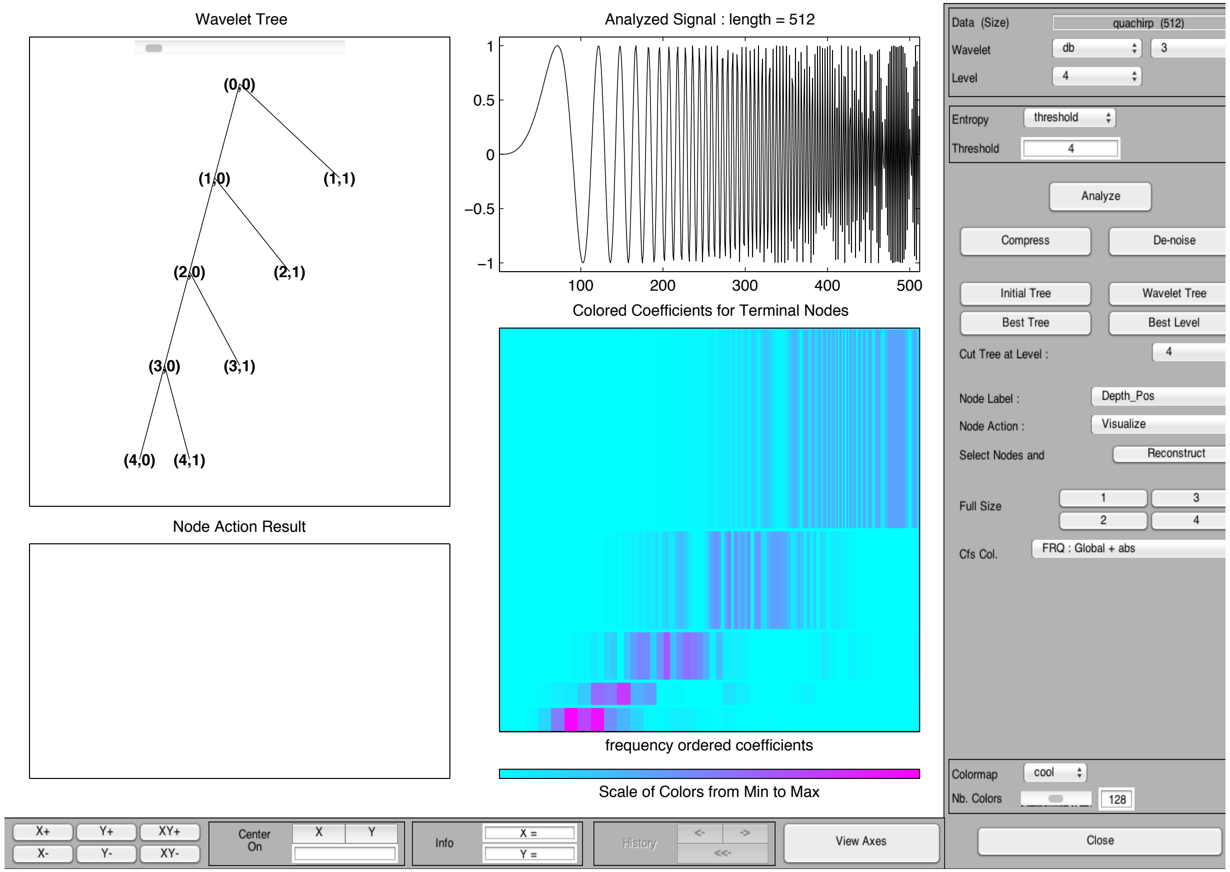
- Concentration in l_p norm $(1 \leq p)$

$$\mathrm{cost}_{norm\ p} = \sum_{n} |s[n]|^p$$

- Logarithm of "energy" (log(0)=0)

$$\mathrm{cost}_{\log} = \sum_{n} \log(s[n]^2)$$

DWT



Data (Size) quachirp (512)

Wavelet db 3

Level 4

Entropy threshold

Threshold 4

Cut Tree at Level : 4

Node Label : Depth_Pos

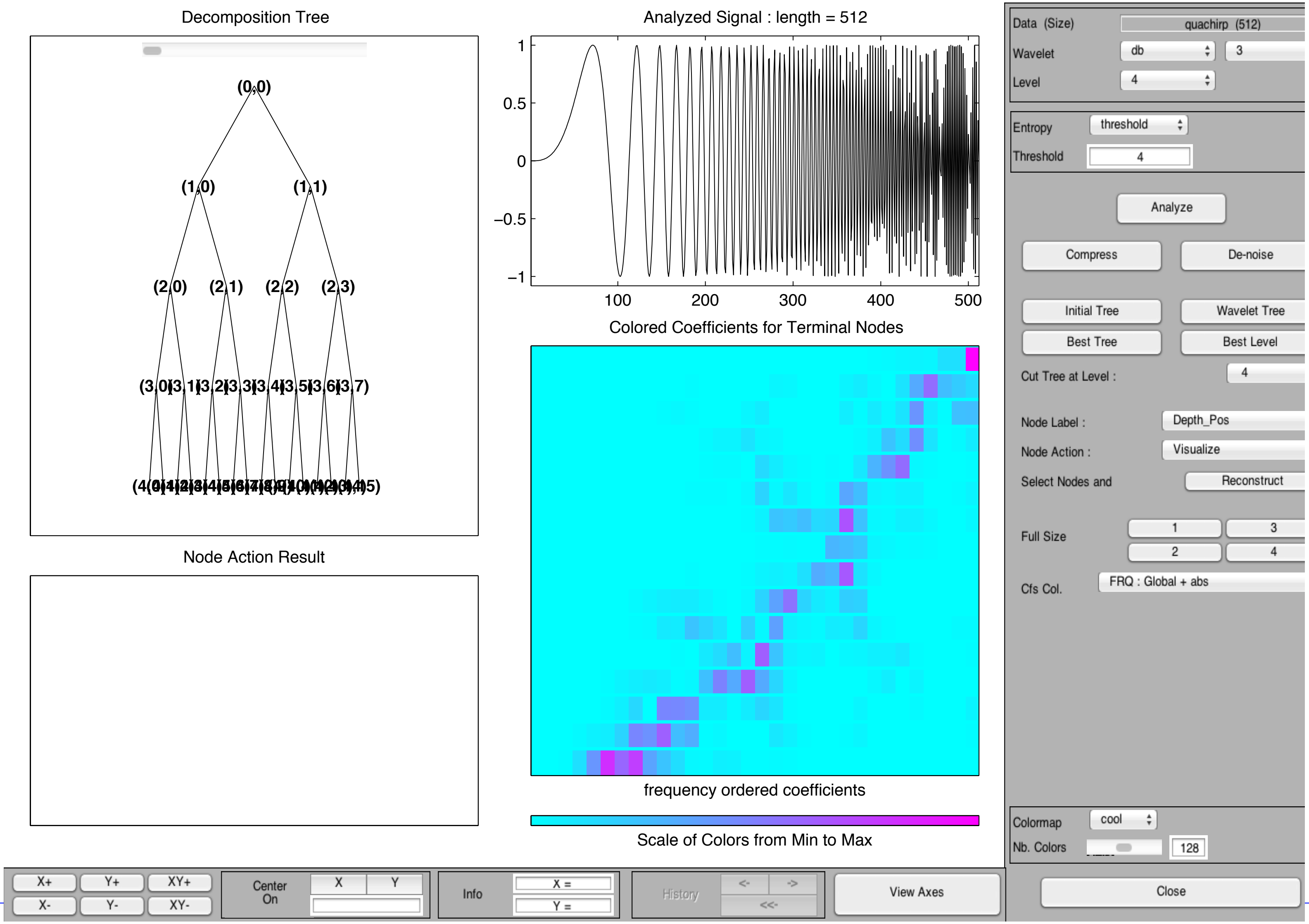
Node Action : Visualize

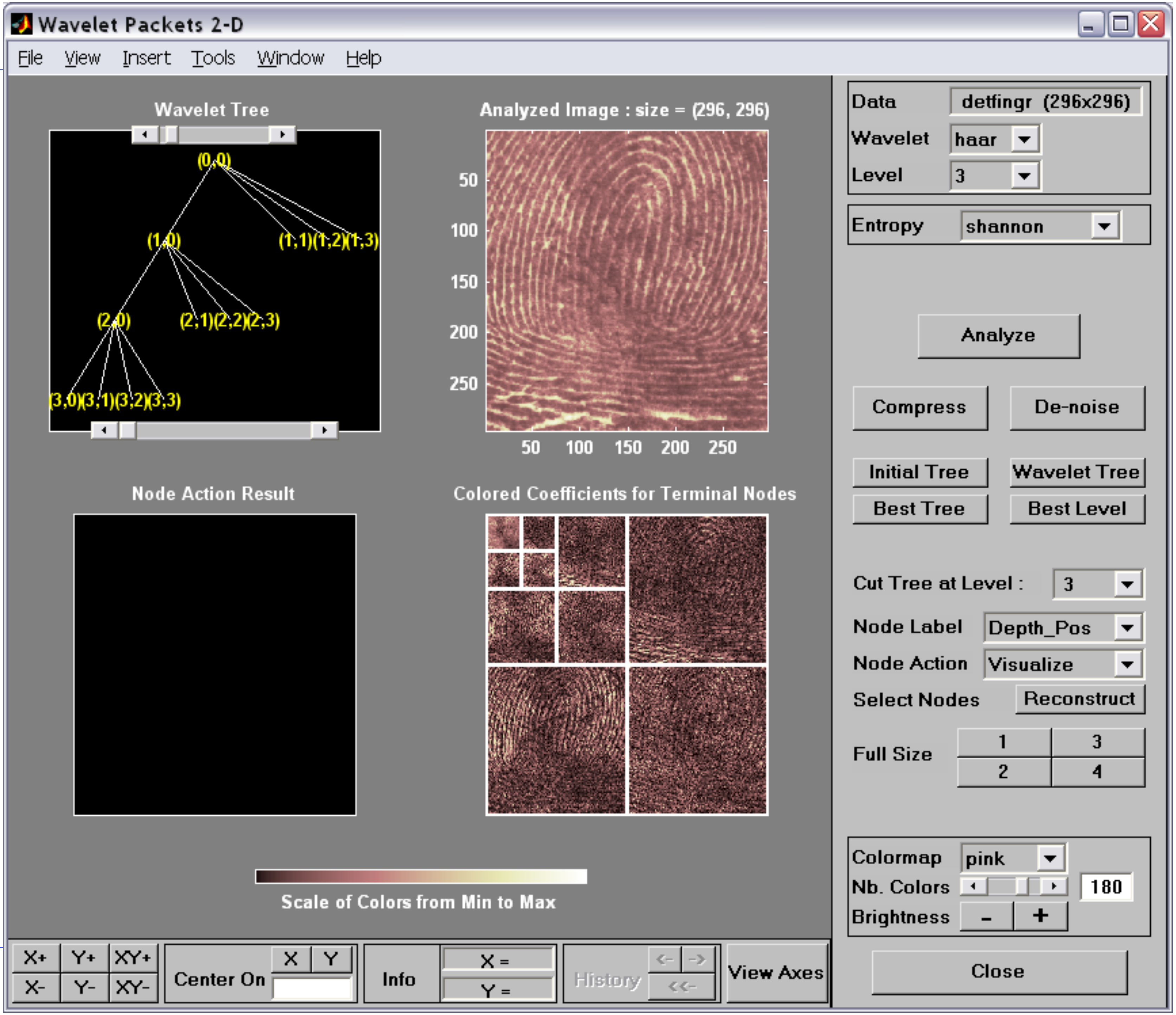
Cfs Col. FRQ : Global + abs

Colormap cool

Nb. Colors 128

WP


Data (Size) quachirp (512)
Wavelet db 3
Level 4
Entropy threshold
Threshold 4
Cut Tree at Level : 4
Node Label : Depth_Pos
Node Action : Visualize
Cfs Col. FRQ : Global + abs
Colormap cool
Nb. Colors 128



Data detfingr (296x296)

Wavelet haar

Level 3

Entropy shannon

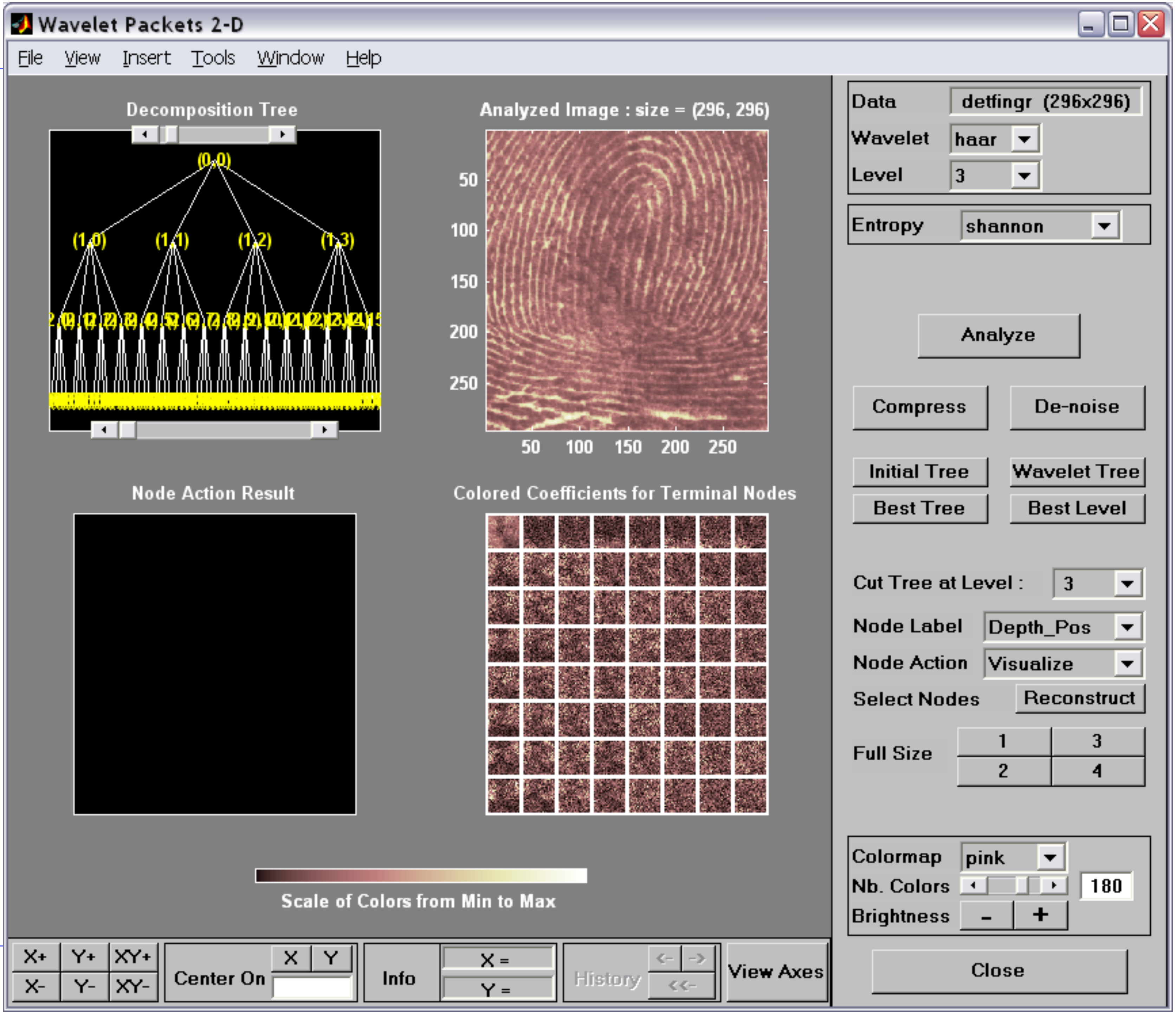
Cut Tree at Level : 3

Node Label Depth_Pos

Node Action Visualize

Colormap pink

Nb. Colors 180



Wavelet Packets 2-D

Data detfingr (296x296)

Wavelet haar

Level 3

Entropy shannon

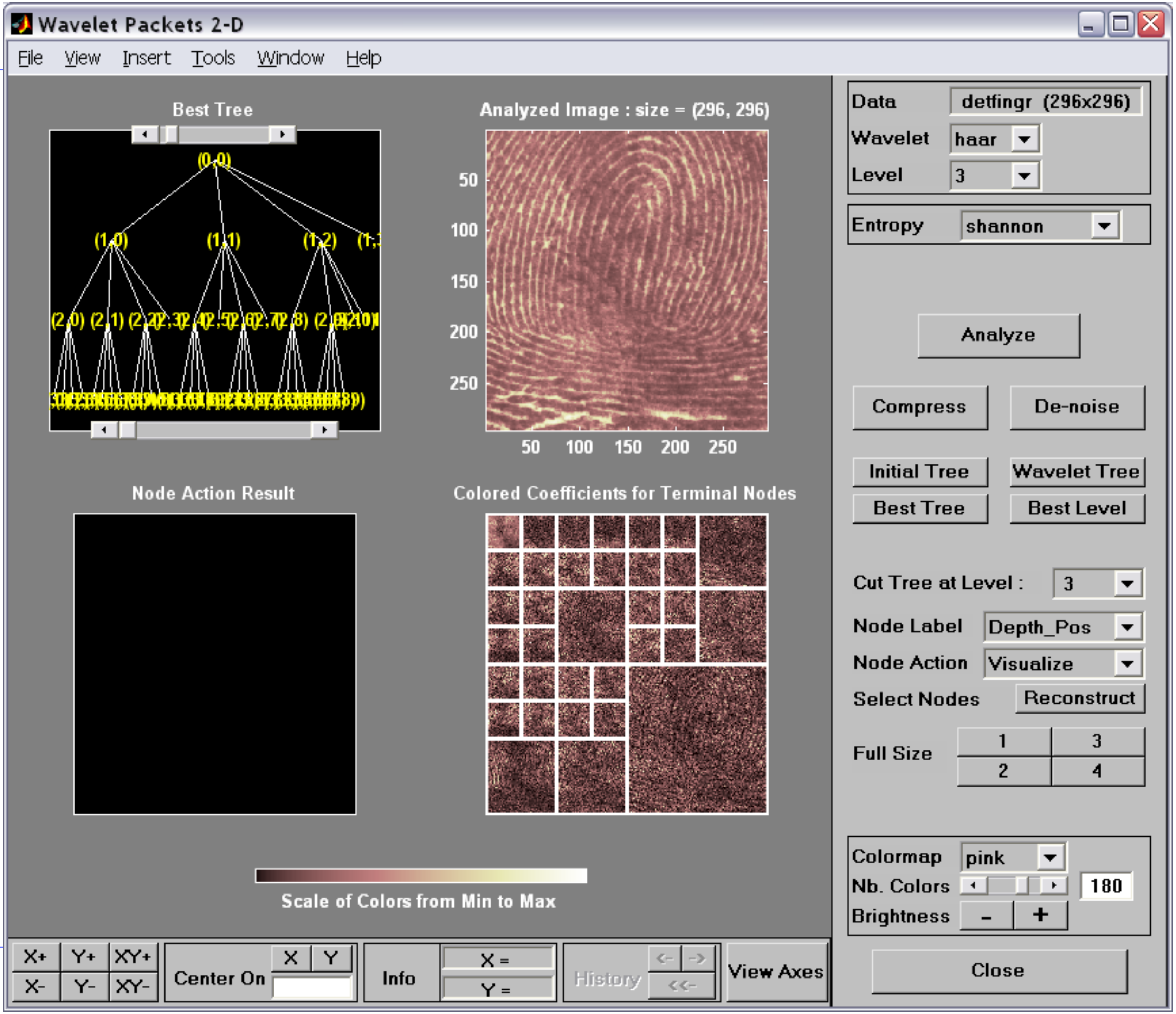
Cut Tree at Level : 3

Node Label Depth_Pos

Node Action Visualize

Colormap pink

Nb. Colors 180



Data: detfingr (296x296)

Wavelet: haar

Level: 3

Entropy: shannon

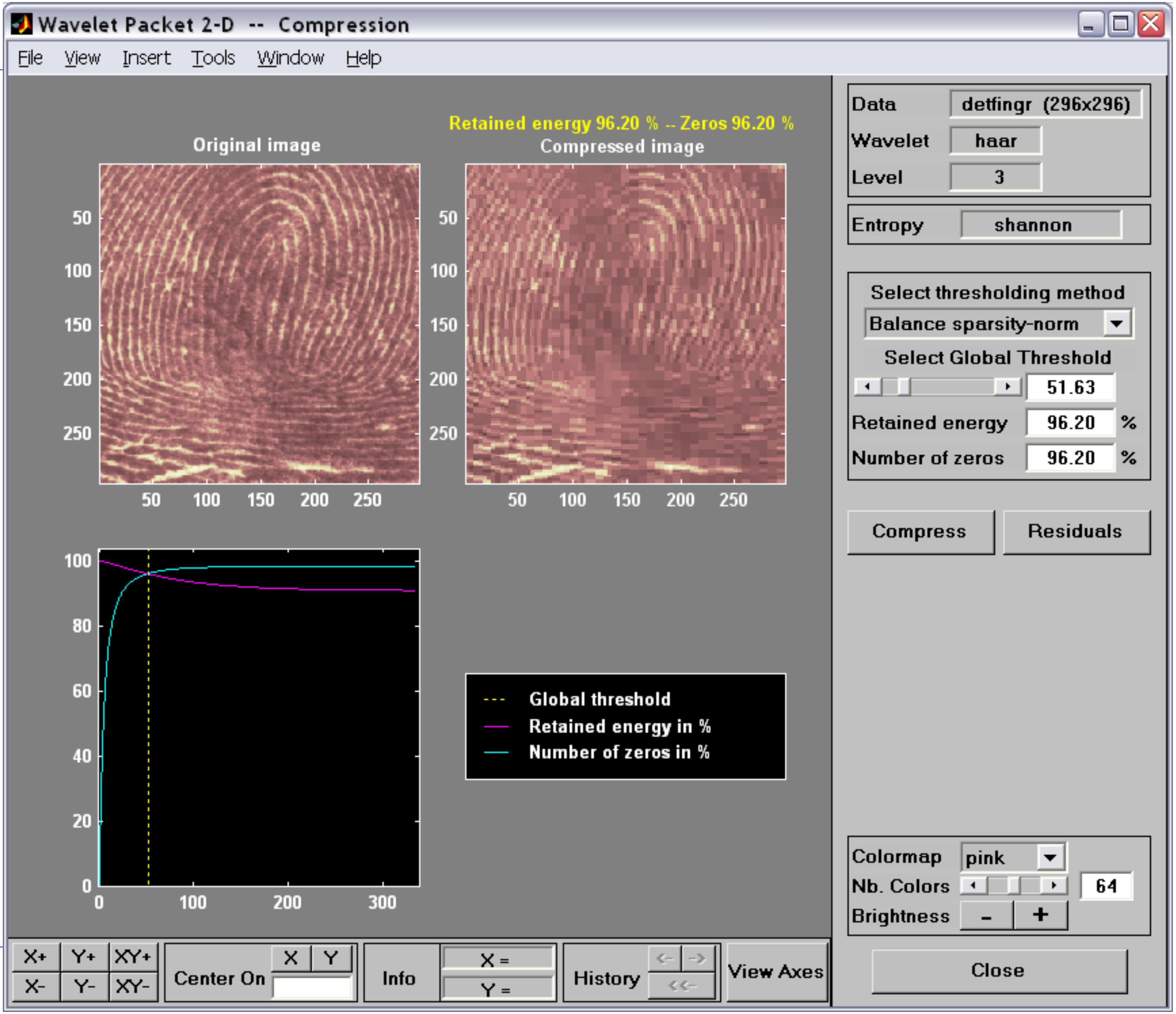
Cut Tree at Level : 3

Node Label: Depth_Pos

Node Action: Visualize

Colormap: pink

Nb. Colors: 180



Data detfingr (296x296)

Wavelet haar

Level 3

Entropy shannon

Select thresholding method

Balance sparsity-norm

Select Global Threshold

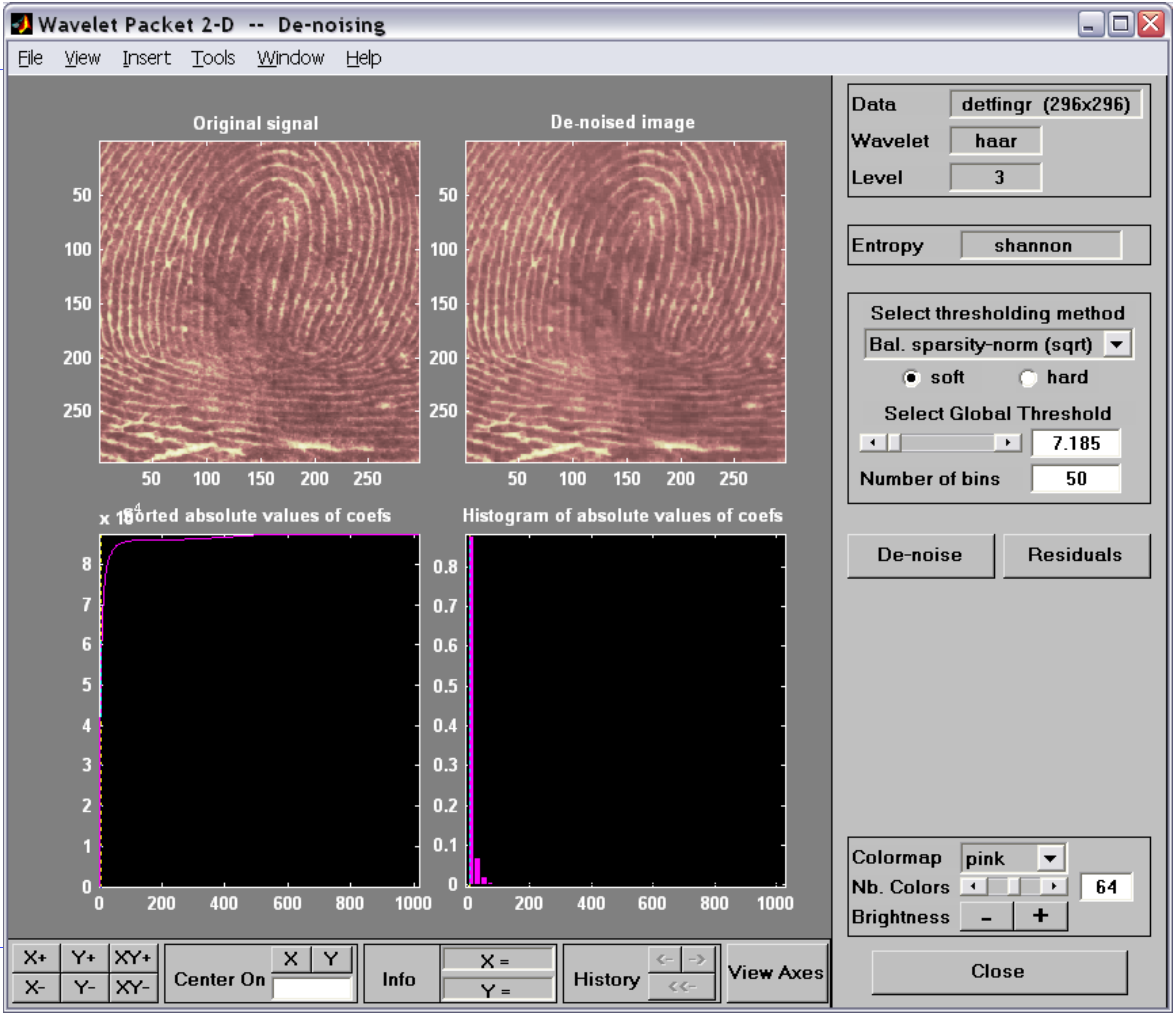
51.63

Retained energy 96.20 %

Number of zeros 96.20 %

Colormap pink

Nb. Colors 64



Data: detfingr (296x296)

Wavelet: haar

Level: 3

Entropy: shannon

Select thresholding method: Bal. sparsity-norm (sqrt)

● soft ○ hard

Select Global Threshold: 7.185

Number of bins: 50

Colormap: pink

Nb. Colors: 64