Wavelets and thresholding

(blended lecture)

DS-GA 1013 / MATH-GA 2824 Mathematical Tools for Data Science

Carlos Fernandez-Granda

2D multiresolution analysis

Thresholding

Image

Multiresolution analysis

Scale / resolution at which information is encoded is not uniform

Goal: Decompose signals into components at different resolutions

Challenge: Design basis of vectors to achieve this

If vectors are orthogonal, then we can just project onto them to separate contributions of each scale

Multiresolution decomposition

Let $N := 2^K$ for some K, a multiresolution decomposition of $\mathbb{R}^N$ is a sequence of nested subspaces $\mathcal{V}_K \subset \mathcal{V}_{K-1} \subset \ldots \subset \mathcal{V}_0$ satisfying:

- $\mathcal{V}_0 = \mathbb{R}^N$
- If $x \in \mathcal{V}_k$ then x shifted by 2^k is also in $\mathcal{V}_k$ (invariance to translations)
- Dilating $x \in \mathcal{V}_j$ yields vector in $\mathcal{V}_{j+1}$

Example: Vectors that are constant on segments of length 2^k

Multiresolution decomposition in 2D

Let $N := 2^K$ for some K, a multiresolution decomposition of $\mathbb{R}^N \times \mathbb{R}^N$ is a sequence of nested subspaces $\mathcal{V}_K \subset \mathcal{V}_{K-1} \subset \ldots \subset \mathcal{V}_0$ satisfying:

- $\mathcal{V}_0 = \mathbb{R}^N \times \mathbb{R}^N$
- If $x \in \mathcal{V}_k$ then x shifted by 2^k horizontally or vertically is also in $\mathcal{V}_k$ (invariance to translations)
- Dilating $x \in \mathcal{V}_j$ yields vector in $\mathcal{V}_{j+1}$

Example

Subspace $\mathcal{V}_k$ contains vectors that are constant on $2^k \times 2^k$ squares

Do the subspaces satisfy the conditions?

- $\mathcal{V}_0 = \mathbb{R}^N \times \mathbb{R}^N$
- If $x \in \mathcal{V}_k$ then x shifted by 2^k horizontally or vertically is also in $\mathcal{V}_k$ (invariance to translations)
- Dilating $x \in \mathcal{V}_j$ yields vector in $\mathcal{V}_{j+1}$

Example

What basis vectors span $\mathcal{V}_k$? Should we use them to decompose signals?

Solution

Decompose the finer subspaces into a direct sum

$$\mathcal{V}_k = \mathcal{V}_{k+1} \oplus \mathcal{W}_{k+1}, \qquad 0 \leq k \leq K-1,$$

$\mathcal{W}_k$ is the orthogonal complement of $\mathcal{V}_{k+1}$ in $\mathcal{V}_k$, so it captures finest resolution available at level k

We can then decompose $\mathbb{R}^N \times \mathbb{R}^N$ into different scales

$$\mathbb{R}^N \times \mathbb{R}^N = \mathcal{V}_0 =$$

Solution

Decompose the finer subspaces into a direct sum

$$\mathcal{V}_k = \mathcal{V}_{k+1} \oplus \mathcal{W}_{k+1}, \qquad 0 \le k \le K-1,$$

$\mathcal{W}_k$ is the orthogonal complement of $\mathcal{V}_{k+1}$ in $\mathcal{V}_k$, so it captures finest resolution available at level k

We can then decompose $\mathbb{R}^N \times \mathbb{R}^N$ into different scales

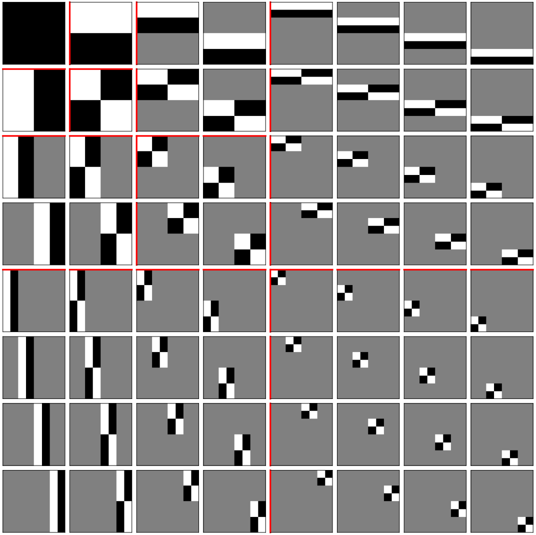
$$\begin{aligned} \mathbb{R}^N \times \mathbb{R}^N = \mathcal{V}_0 &= \mathcal{V}_1 \oplus \mathcal{W}_1 \\ &= \mathcal{V}_2 \oplus \mathcal{W}_2 \oplus \mathcal{W}_1 \\ &= \mathcal{V}_k \oplus \mathcal{W}_k \oplus \cdots \oplus \mathcal{W}_2 \oplus \mathcal{W}_1 \end{aligned}$$

$$\mathbb{R}^4 \times \mathbb{R}^4 = \mathcal{V}_0 = \mathcal{V}_1 \oplus \mathcal{W}_1$$

$$\mathcal{V}_1 = \mathcal{V}_2 \oplus \mathcal{W}_2$$

$$\mathbb{R}^4 \times \mathbb{R}^4 = \mathcal{V}_0 = \mathcal{V}_2 \oplus \mathcal{W}_2 \oplus \mathcal{W}_1$$

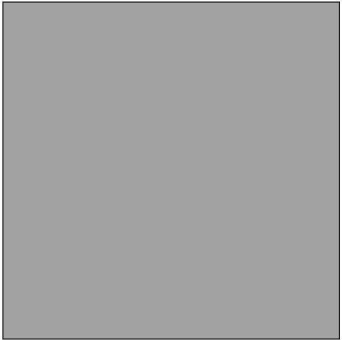
$$\mathbb{R}^{16} \times \mathbb{R}^{16} = \mathcal{V}_0 = \mathcal{V}_2 \oplus \mathcal{W}_3 \oplus \mathcal{W}_2 \oplus \mathcal{W}_1$$



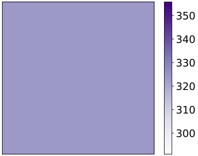
Image

2D Haar wavelet decomposition

Projection onto $\mathcal{V}_9$

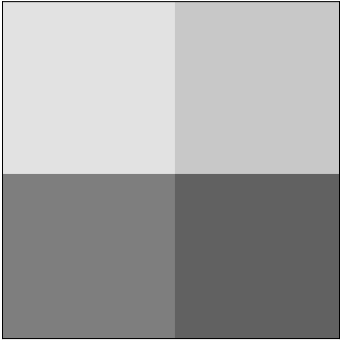


Coefficients for $\mathcal{V}_9$

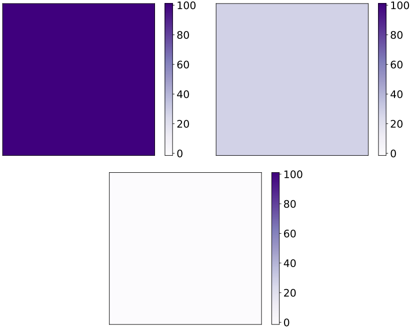


2D Haar wavelet decomposition

Projection onto $\mathcal{V}_8$

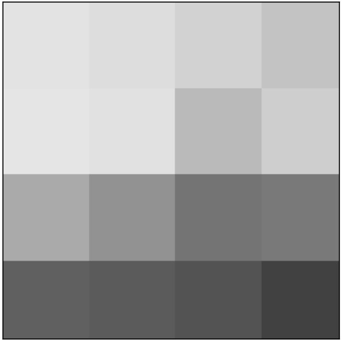


Coefficients for $\mathcal{W}_9$

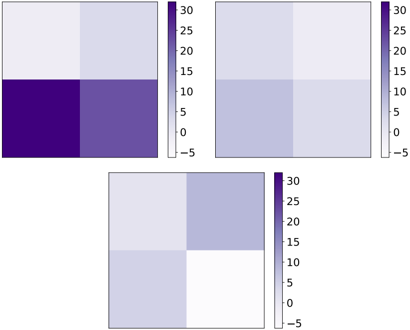


2D Haar wavelet decomposition

Projection onto $\mathcal{V}_7$

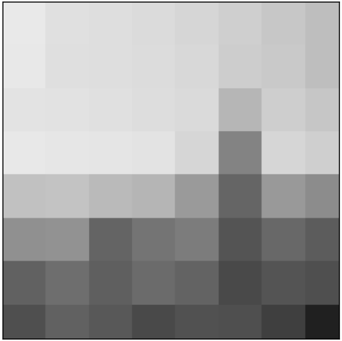


Coefficients for $\mathcal{W}_8$

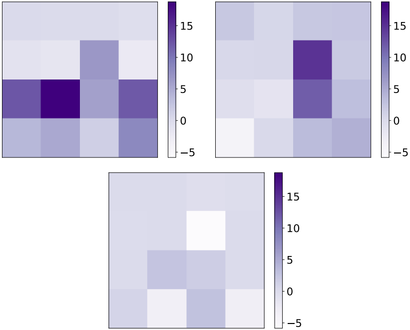


2D Haar wavelet decomposition

Projection onto $\mathcal{V}_6$

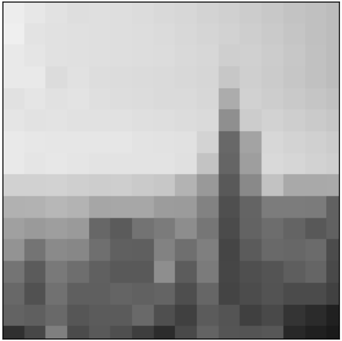


Coefficients for $\mathcal{W}_7$

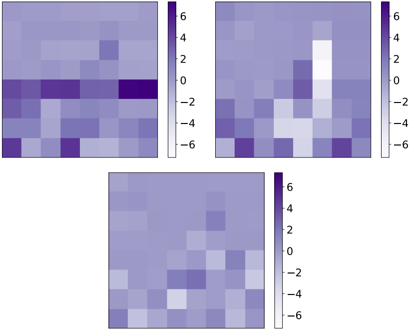


2D Haar wavelet decomposition

Projection onto $\mathcal{V}_5$



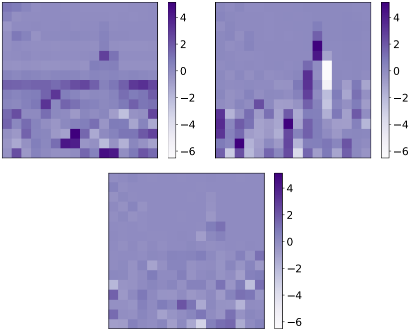
Coefficients for $\mathcal{W}_6$



2D Haar wavelet decomposition

Projection onto $\mathcal{V}_4$

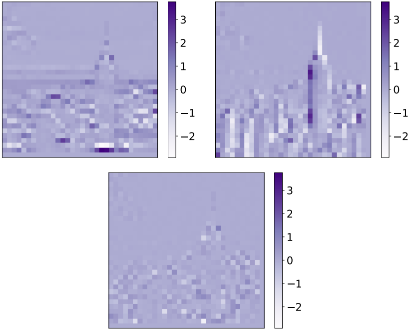
Coefficients for $\mathcal{W}_5$



2D Haar wavelet decomposition

Projection onto $\mathcal{V}_3$

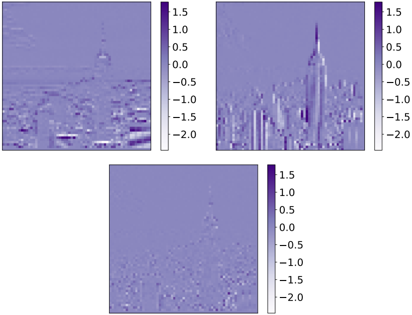
Coefficients for $\mathcal{W}_4$



2D Haar wavelet decomposition

Projection onto $\mathcal{V}_2$

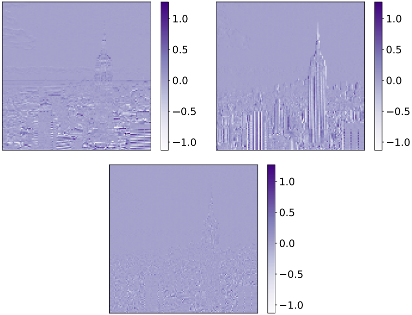
Coefficients for $\mathcal{W}_3$



2D Haar wavelet decomposition

Projection onto $\mathcal{V}_1$

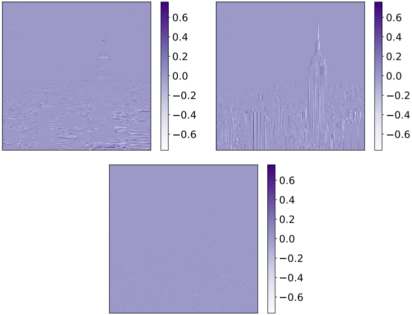
Coefficients for $\mathcal{W}_2$



2D Haar wavelet decomposition

Projection onto $\mathcal{V}_0$

Coefficients for $\mathcal{W}_1$



2D multiresolution analysis

Thresholding

Denoising

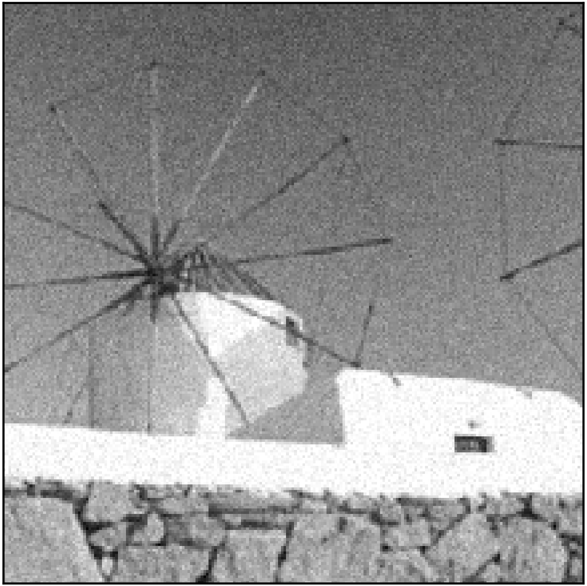
In real applications measurements are usually corrupted by noise

Simple, yet useful, model:

$$\text{data} = \text{signal} + \text{noise}$$

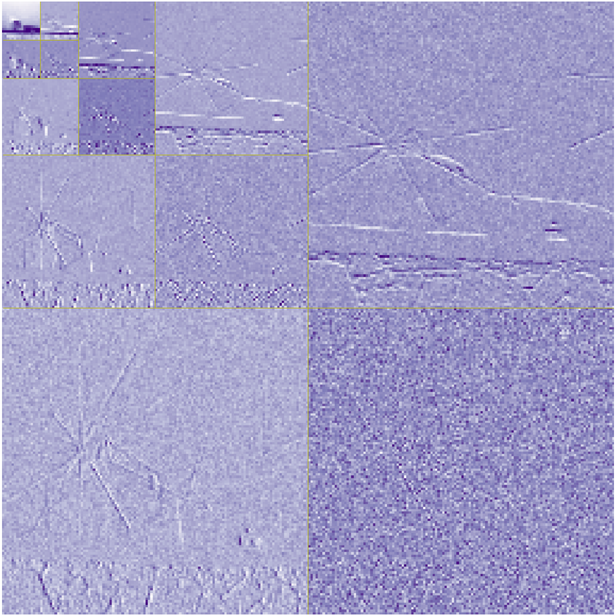
Denoising is the problem of estimating the signal from the noisy data

Noisy image

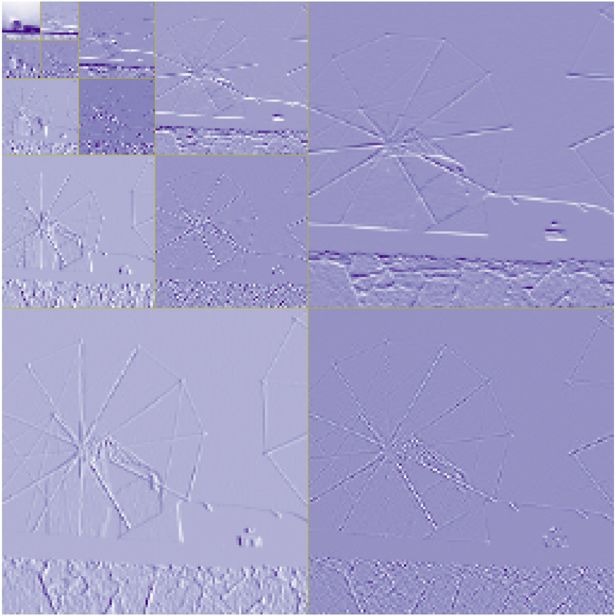


Image

Wavelet coefficients of noisy image



Wavelet coefficients of clean image

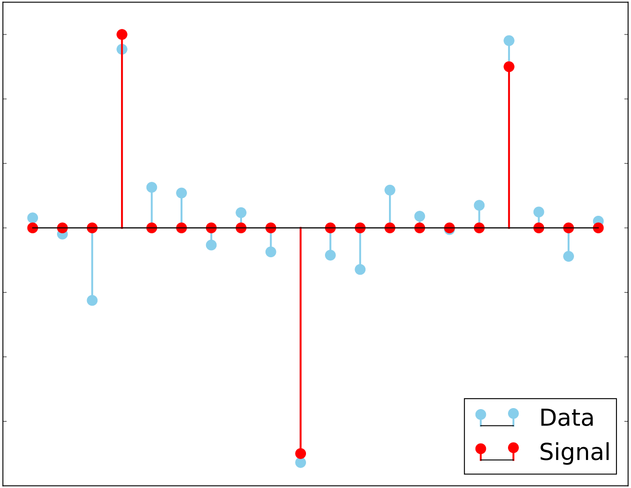


How can we exploit this to denoise?

Wavelet coefficients of signal are sparse

Wavelet coefficients of noise are dense

How would you denoise this signal?

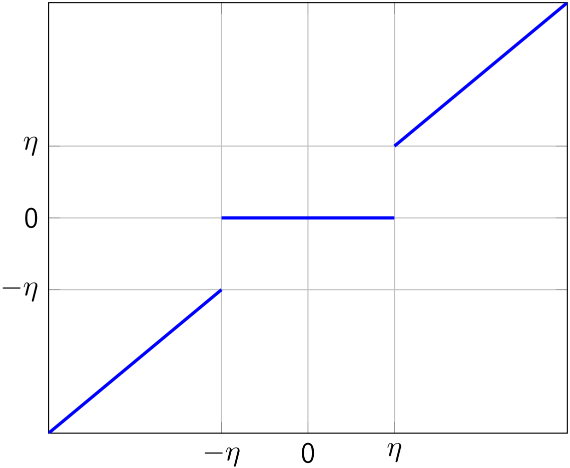


Thresholding

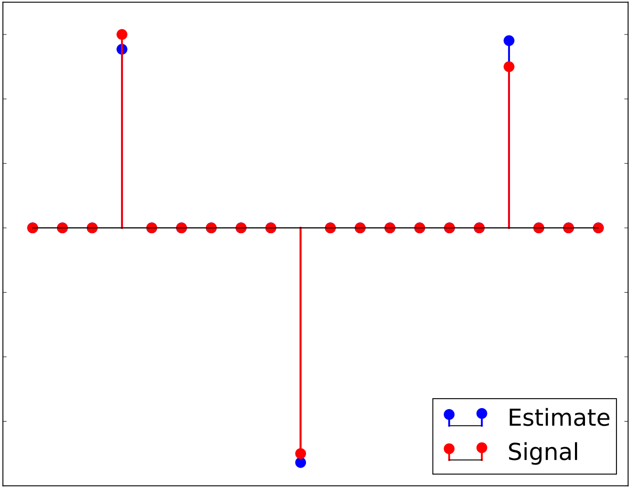
Hard-thresholding operator

$$\mathcal{H}_\eta(v)[j] := \begin{cases} v[j] & \text{if } |v[j]| > \eta \\ 0 & \text{otherwise} \end{cases}$$

Hard-thresholding



Denoising via hard thresholding



Denoising via hard thresholding

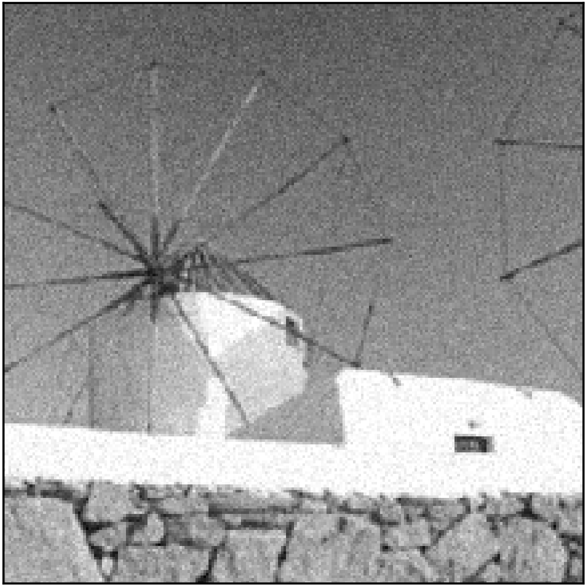
Given data y and a sparsifying linear transform A

1. Compute coefficients Ay
2. Apply the hard-thresholding operator $\mathcal{H}_\eta : \mathbb{C}^N \rightarrow \mathbb{C}^N$ to Ay
3. Invert the transform

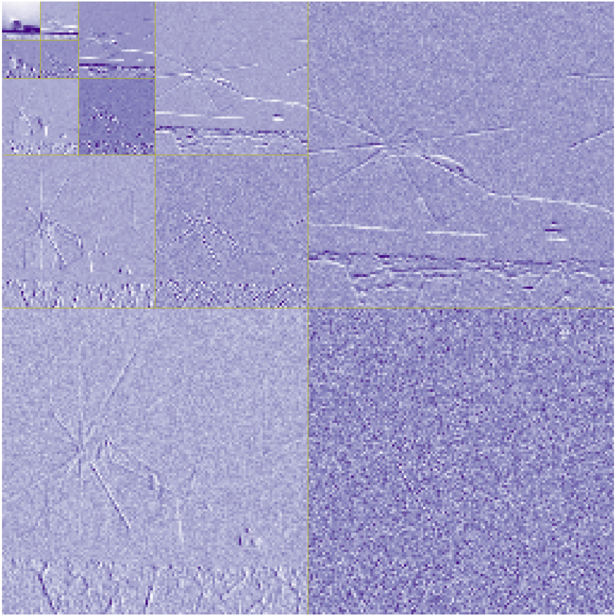
$$x_{\text{est}} := L\,\mathcal{H}_\eta\,(Ay)\,,$$

where L is a left inverse of A

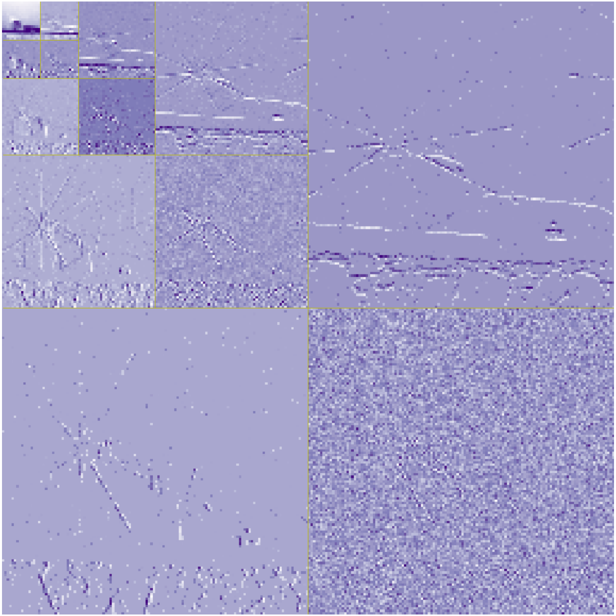
Noisy image



Noisy wavelet coefficients



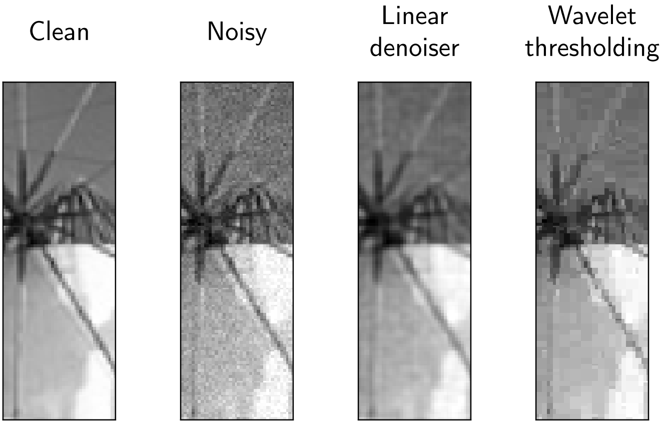
Thresholded wavelet coefficients



Denoised image

Clean image

Comparison



Alternative approach: Use ℓ_1-norm to promote sparsity

Cost function?

Alternative approach: Use ℓ_1-norm to promote sparsity

Cost function?

$$\min_x \frac{1}{2} ||y - x||_2^2 + \lambda ||Ax||_1 = \min_c \frac{1}{2} \left|\left|y - A^T c\right|\right|_2^2 + \lambda ||c||_1$$

where A is an orthogonal wavelet transform

ℓ_1-norm regularization

Subgradients of

$$\min_c \frac{1}{2} \left|\left| y - A^T c \right|\right|_2^2 + \lambda \left|\left|c\right|\right|_1$$

ℓ_1-norm regularization

Subgradients of

$$\min_c \frac{1}{2} \left|\left|y - A^T c\right|\right|_2^2 + \lambda \left|\left|c\right|\right|_1$$

$AA^T c - Ay + \lambda g = c + Ay + \lambda g$ where g is subgradient of ℓ_1 norm

Subgradients of ℓ_1 norm

g is a subgradient of the ℓ_1 norm at $x \in \mathbb{R}^n$ if and only if

$$g[i] = \text{sign}\left(x[i]\right) \qquad \text{if } x[i] \neq 0$$

$$|g[i]| \leq 1 \qquad \text{if } x[i] = 0$$

ℓ_1-norm regularization

Solution to

$$\min_{c} \frac{1}{2} \left|\left| y - A^T c \right|\right|_2^2 + \lambda \left|\left| c \right|\right|_1$$

ℓ_1-norm regularization

Solution to

$$\min_c \frac{1}{2} \left|\left| y - A^T c \right|\right|_2^2 + \lambda \left|\left| c \right|\right|_1$$

$$\lambda g = Ay - c^*$$

If $c^*[j] > 0$

$$c^* = Ay - \lambda$$

If $c^*[j] < 0$

$$c^* = Ay + \lambda$$

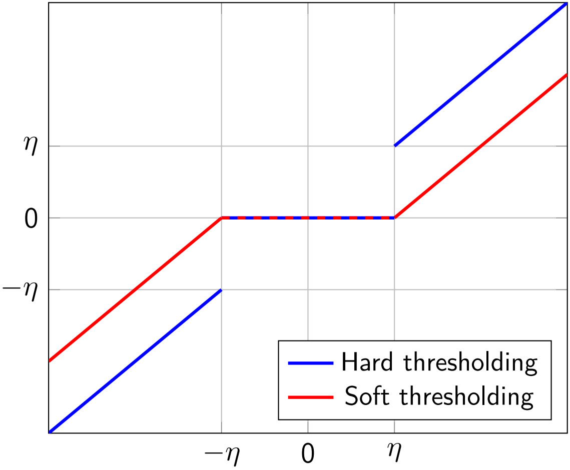
If $c^*[j] = 0$

$$|Ay - c^*| \leq \lambda$$

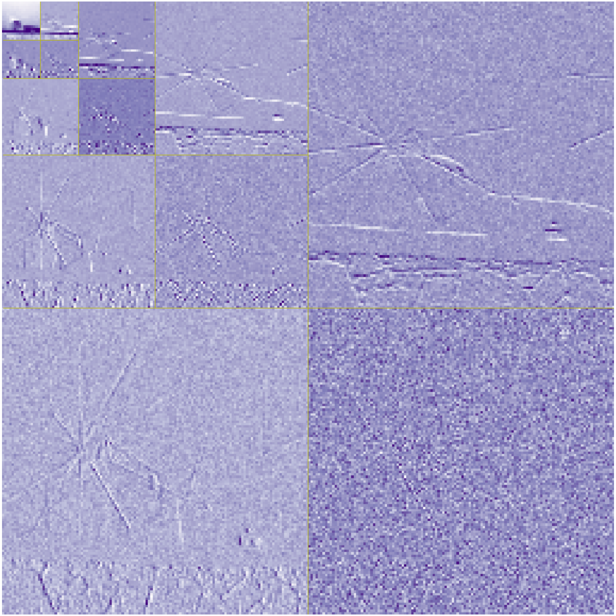
Soft-thresholding operator

$$\mathcal{S}_{\eta}\left(c\right)_i := \begin{cases} y_i - \operatorname{sign}\left(c_i\right)\eta & \text{if } |c_i| \geq \eta \\ 0 & \text{otherwise} \end{cases}$$

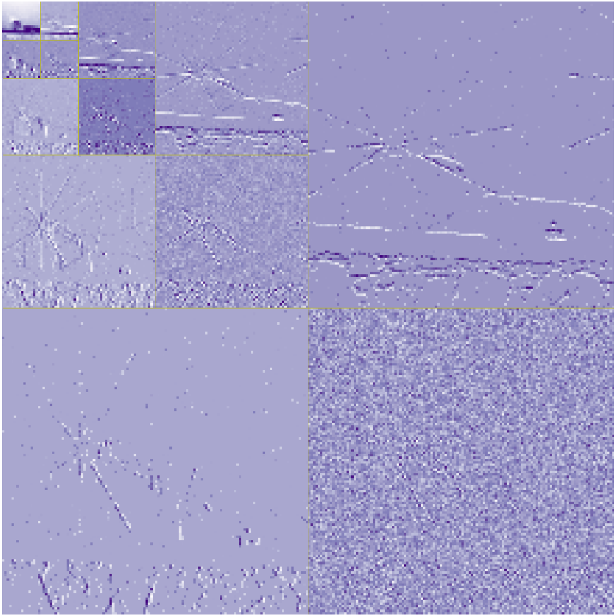
Soft thresholding



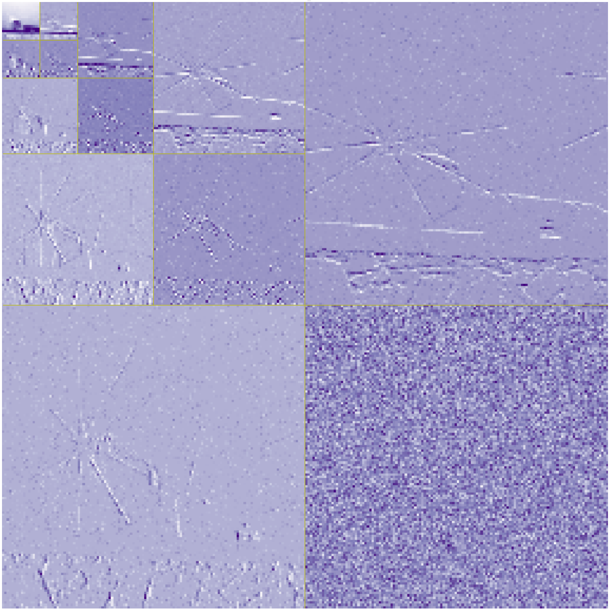
Noisy wavelet coefficients



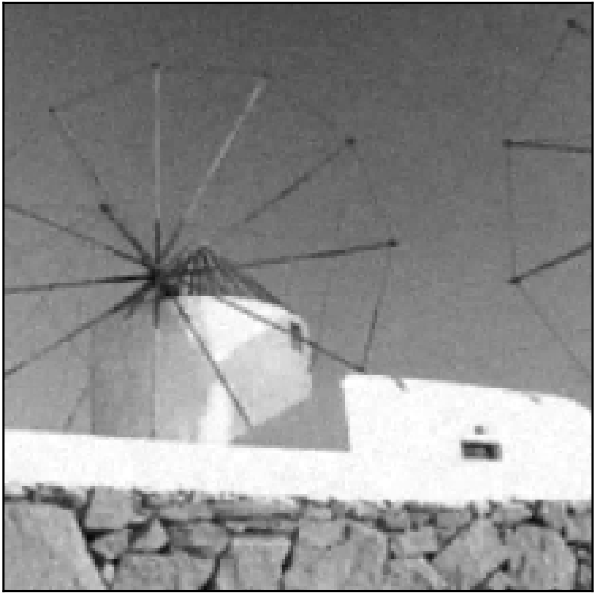
Hard thresholding



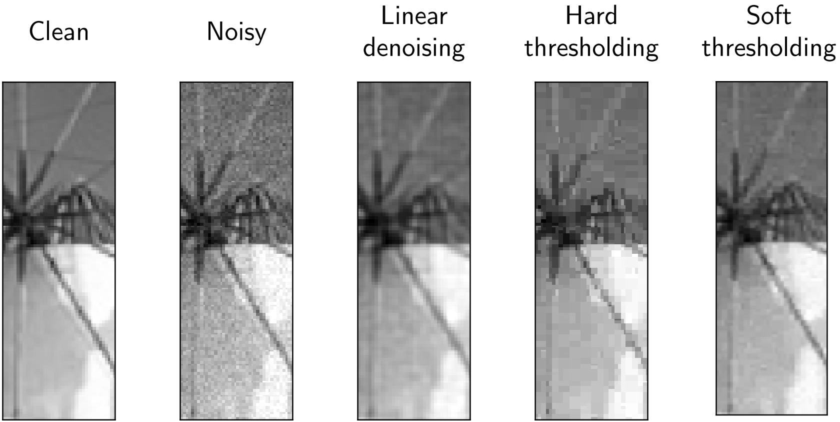
Soft thresholding



Denoised signal



Comparison


Clean
Noisy
Linear denoising
Hard thresholding
Soft thresholding