

Wavelets and Applications

Kévin Polisano

kevin.polisano@univ-grenoble-alpes.fr

M2 MSIAM & Ensimag 3A MMIS

December 3, 2021



LABORATOIRE
JEAN KUNTZMANN
MATHÉMATIQUES APPLIQUÉES - INFORMATIQUE

The 2D Discrete Wavelet Transform

The 2D Haar Basis

From $\varphi(x)$ and $\psi(x)$ one can define the bidimensional functions:

$$\begin{aligned}\Phi(x, y) &= \varphi(x)\varphi(y), & \Psi^1(x, y) &= \psi(x)\varphi(y) \\ \Psi^2(x, y) &= \varphi(x)\psi(y), & \Psi^3(x, y) &= \psi(x)\psi(y)\end{aligned}$$

The values of Ψ^1 , Ψ^2 and Ψ^3 on $[0, 1] \times [0, 1]$ are:

1	-1
---	----

1
-1

1	-1
-1	1

For $i = 1, 2, 3$, $j \geq 0$ et $\mathbf{k} = (k_x, k_y)$, $0 \leq k_x, k_y \leq 2^j - 1$:

$$\Psi_{j,\mathbf{k}}^i(x, y) = 2^{\frac{j}{2}} \Psi^i(2^j x - k_x, 2^j y - k_y)$$

The family $\{\Phi, \Psi_{j,\mathbf{k}}^1, \Psi_{j,\mathbf{k}}^2, \Psi_{j,\mathbf{k}}^3\}$ is an **orthonormal basis** of $L^2([0, 1]^2)$.

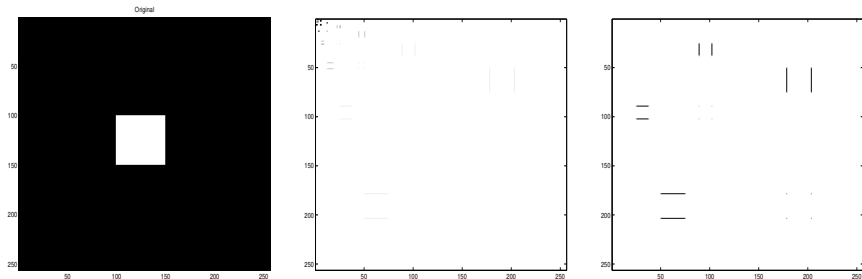
2D Haar expansion

$$f = C_0 + \sum_{j=0}^{+\infty} \sum_{k_x, k_y=0}^{2^j-1} \left(D_{j,k}^1 \Psi_{j,k}^1 + D_{j,k}^2 \Psi_{j,k}^2 + D_{j,k}^3 \Psi_{j,k}^3 \right)$$

with $C_0 = \iint_{[0,1]^2} f$ and $D_{j,k}^i = \iint_{[0,1]^2} f \Psi_{j,k}^i$

C_{J-2}	D_{J-2}^1	D_{J-1}^1
D_{J-2}^2	D_{J-2}^3	
D_{J-1}^2		D_{J-1}^3

Example: square

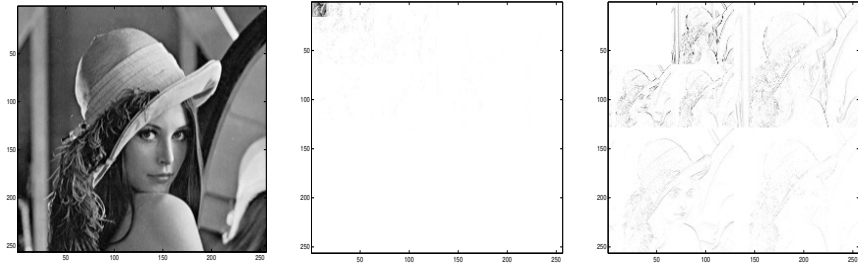


Left figure: *Original image.*

Middle figure: *entire Haar coefficient map.*

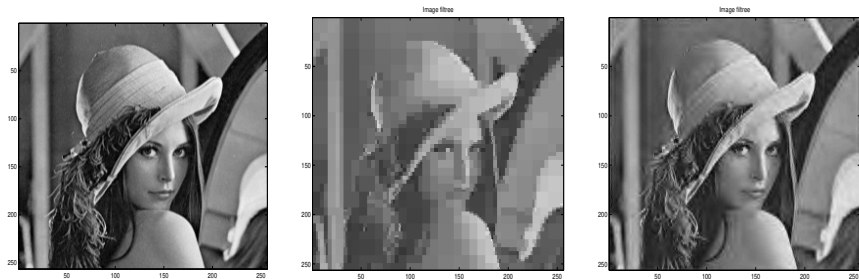
Right Figure: *coefficients of the two finest scales.*

Image decomposition on the Haar basis



Left figure: *Original image (256^2 pixels) and its Haar coefficients:*
Middle figure: *entire Haar coefficient map.*
Right Figure: *coefficients of the two finest scales.*

Image compression with the Haar basis



Original image and compressed images:

Middle figure: *keeping the 1024 largest coefficients (compression 98,4%).*

Right Figure: *keeping the 3467 largest coefficients (compression 94,7%).*

Image decomposition

The wavelet bases 2D are constructed by tensor product of 1D bases. Let φ and ψ be the scaling function and wavelet of a MRA. Two constructions are available:

(1) Pure tensor product wavelet bases (anisotropic) :

$$\Psi_{k,k'}^{j,j'}(x,y) = \psi_{j,k}(x)\psi_{j',k'}(y), \quad j, j' \in \mathbb{Z}, k, k' \in \mathbb{Z}$$

The associated 2D wavelet transform uses 1D FWT on lines followed by 1D FWT on columns of the image.

(2) Wavelets arising from tensor product 2D MRA of $L^2(\mathbb{R}^2)$:

- **Approximation space** (scaling functions) $\mathcal{V}_j = V_j \otimes V_j$
- **Detail spaces** (wavelets) \mathcal{W}_j defined by $\mathcal{V}_{j+1} = \mathcal{V}_j \oplus \mathcal{W}_j$

$$\begin{aligned}\mathcal{V}_{j+1} &= V_{j+1} \otimes V_{j+1} \\ &= (V_j \oplus W_j) \otimes (V_j \oplus W_j) \\ &= (V_j \otimes V_j) \oplus (W_j \otimes V_j) \oplus (V_j \otimes W_j) \oplus (W_j \otimes W_j)\end{aligned}$$

Separable scaling function and wavelets

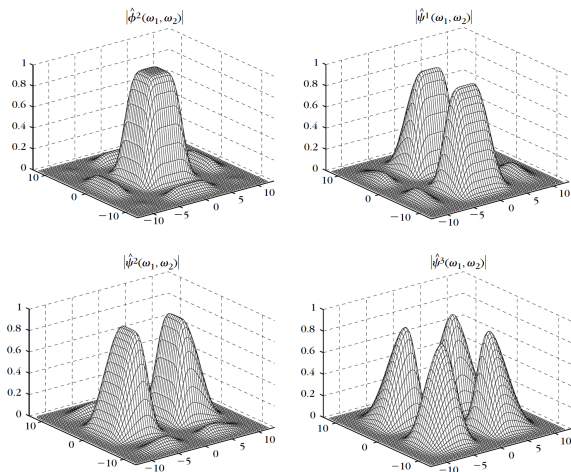


Figure: Fourier transforms of a separable scaling function and of three separable wavelets calculated from a one-dimensional Daubechies 4 wavelet

2D multi-resolutions

$$\Phi(x, y) = \varphi(x)\varphi(y), \quad \Psi^1(x, y) = \psi(x)\varphi(y)$$

$$\Psi^2(x, y) = \varphi(x)\psi(y), \quad \Psi^3(x, y) = \psi(x)\psi(y)$$

$$\mathcal{V}_{j+1} = \underbrace{(V_j \otimes V_j)}_{\text{coarse approx}} \oplus \underbrace{(W_j \otimes V_j)}_{\text{horiz. details}} \oplus \underbrace{(V_j \otimes W_j)}_{\text{vert. details}} \oplus \underbrace{(W_j \otimes W_j)}_{\text{diag. details}}$$

$\mathcal{V}_j = \text{Span}\{\Phi_{j,k}\}_k$
 $\mathcal{W}_j^1 = \text{Span}\{\Psi_{j,k}^1\}_k$
 $\mathcal{W}_j^2 = \text{Span}\{\Psi_{j,k}^2\}_k$
 $\mathcal{W}_j^3 = \text{Span}\{\Psi_{j,k}^3\}_k$

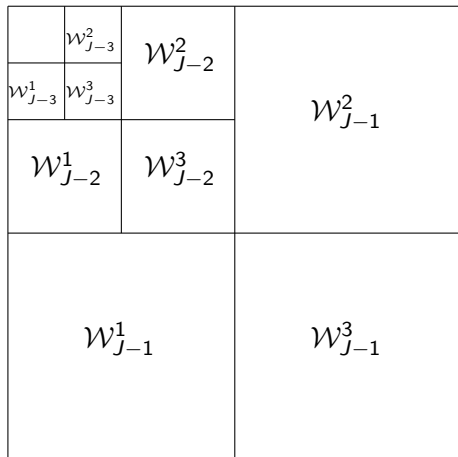
Image $C_J[k] = \langle f, \Phi_{J,k} \rangle$, wavelet coefficients: $D_j^i[k] = \langle f, \Psi_{j,k}^i \rangle$



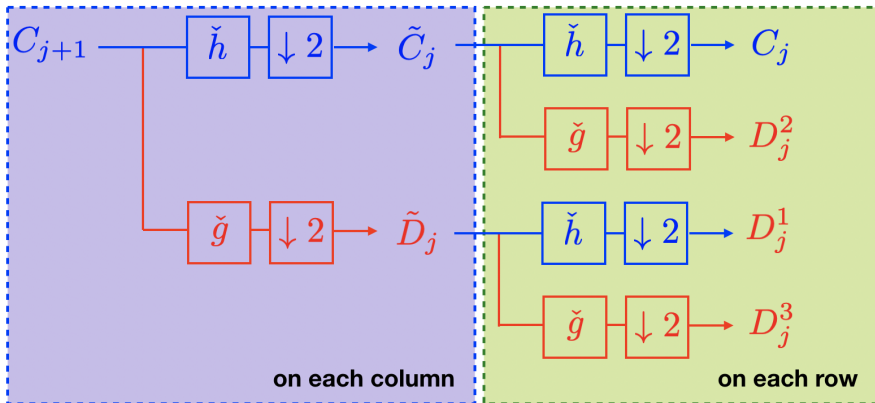
Figure: (Haar approximation) $\mathbf{P}_{\mathcal{V}_j} f = \sum_k \langle f, \Phi_{j,k} \rangle \Phi_{j,k} = \sum_k C_{j,k} \Phi_{j,k}$

Image decomposition

$$\mathcal{W}_j = \text{Vec}\{\psi_{j,k}(x)\varphi_{j,k'}(y); \varphi_{j,k}(x)\psi_{j,k'}(y); \psi_{j,k}(x)\psi_{j,k'}(y) \mid (k, k') \in \mathbb{Z}^2\}$$



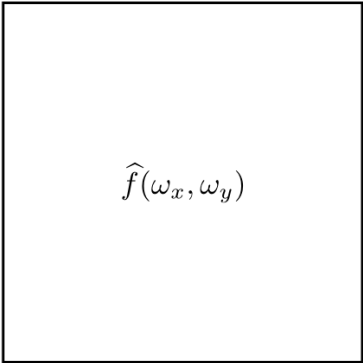
Fast 2D Wavelet Transform



Fast 2D Wavelet Transform

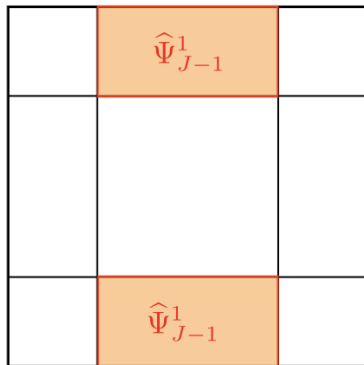
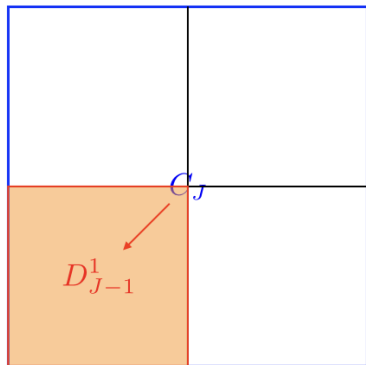


C_J

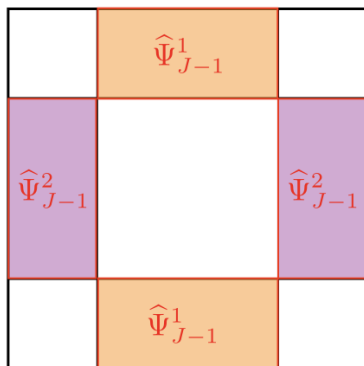
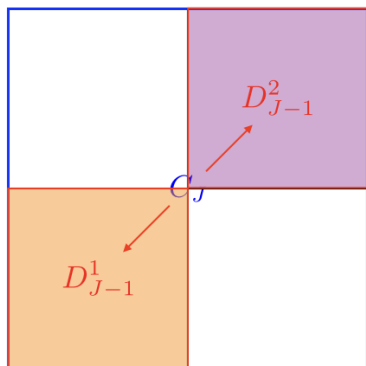


$\hat{f}(\omega_x, \omega_y)$

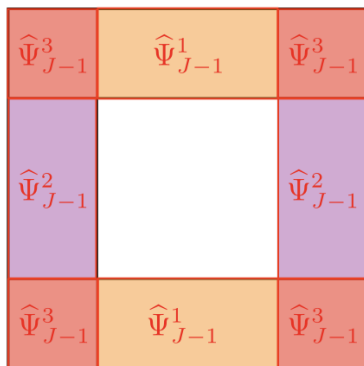
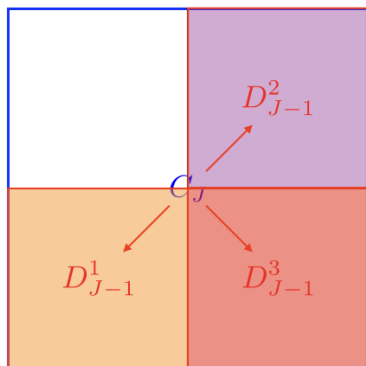
Fast 2D Wavelet Transform



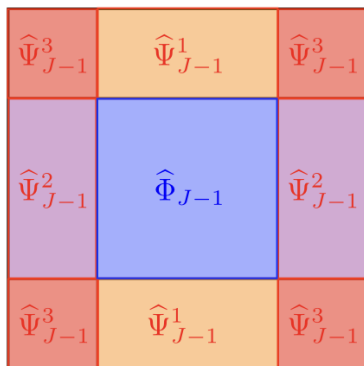
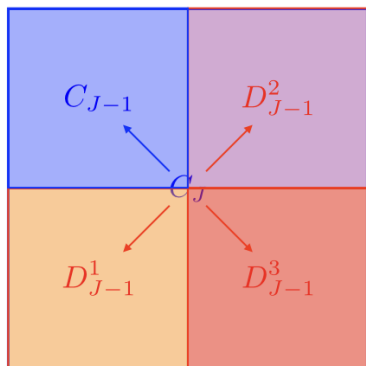
Fast 2D Wavelet Transform



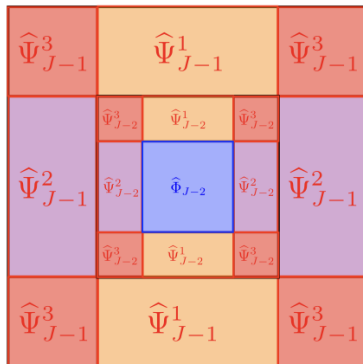
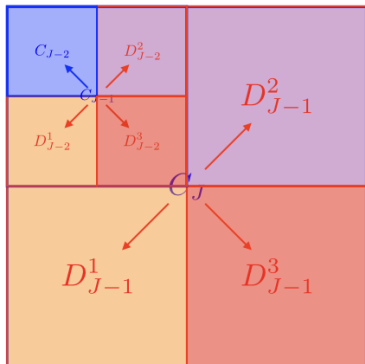
Fast 2D Wavelet Transform



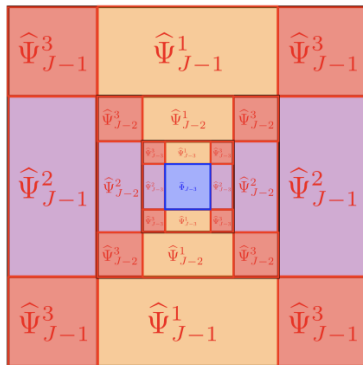
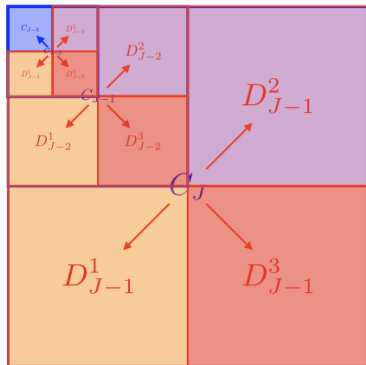
Fast 2D Wavelet Transform



Fast 2D Wavelet Transform



Fast 2D Wavelet Transform



Lena multi-resolution

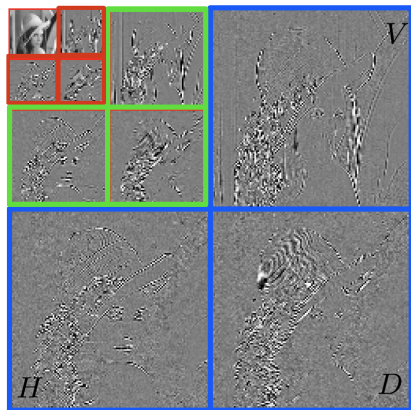
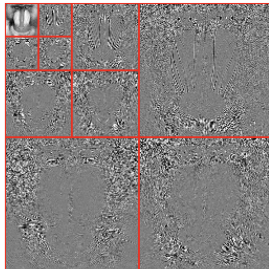
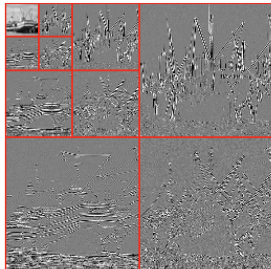
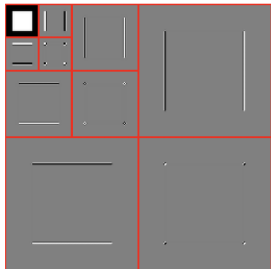
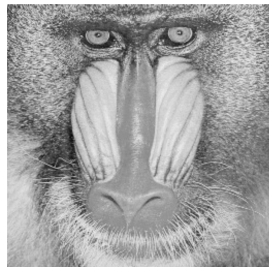
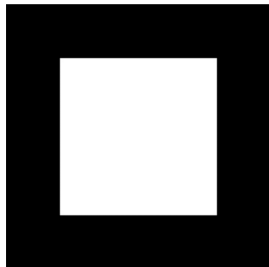


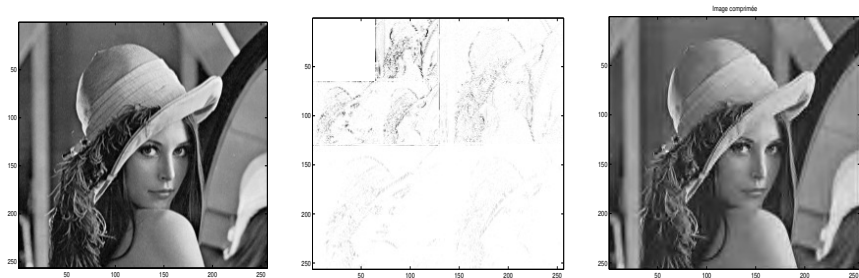
Figure: (Left) original Lena C_0 . (Right) coarse approximation C_6 (small Lena on the top left corner in red) and the discrete wavelets coefficients D_j^i for $i \in \{1, 2, 3\}$ and at scales $j = 8$, $j = 7$ and $j = 6$

Other examples



Credits: G. Peyré

Example: Lena compression



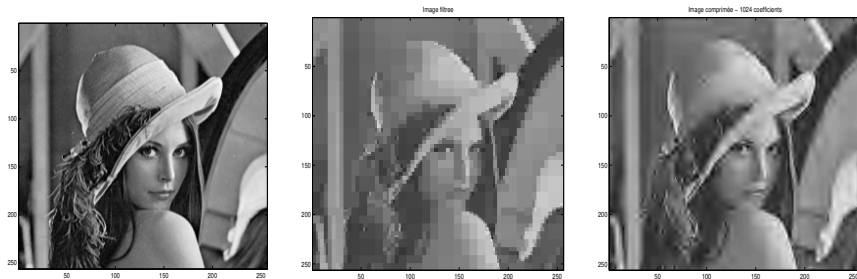
Left figure: *Image* $256 \times 256 = 65536$ pixels, 256 grey levels.

Middle figure: *wavelet coefficients of the 2 finest scales*

Right figure: *Reconstructed image from its 4000 largest wavelet coefficients (4 vanishing moments).*

Compression factor = $(65536 - 4000) * 100 / 65536 = 93,9 \%$

Comparison Haar basis vs. regular wavelets



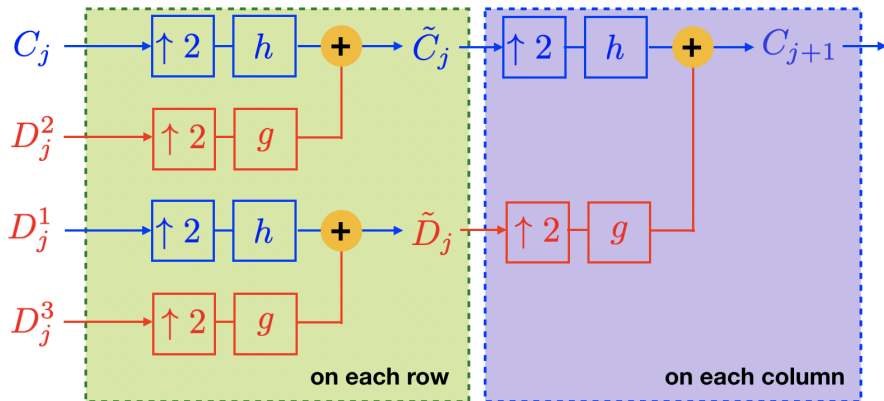
Left figure: *Image* $256 \times 256 = 65536$ valeurs, 256 niveaux de gris.

Middle figure: *reconstructed image from its 1024 largest wavelet coefficients in the Haar basis.*

Right figure: *reconstructed image from its 1024 largest wavelet coefficients in a wavelet bases with 4 vanishing moments.*

Compression factor = $(65536 - 1024) * 100 / 65536 = 98,44 \%$

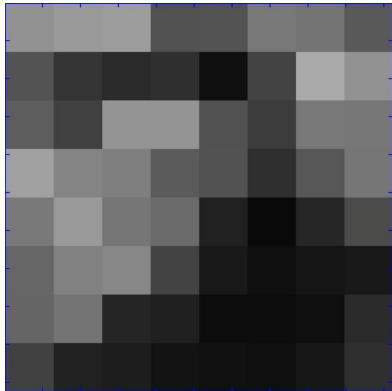
Fast 2D Inverse Wavelet Transform



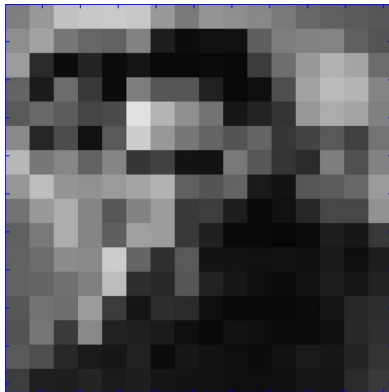
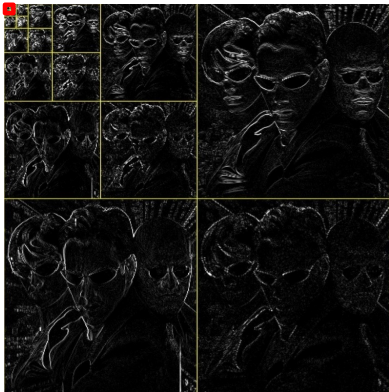
JPEG-2K Compression



JPEG-2K Compression



JPEG-2K Compression



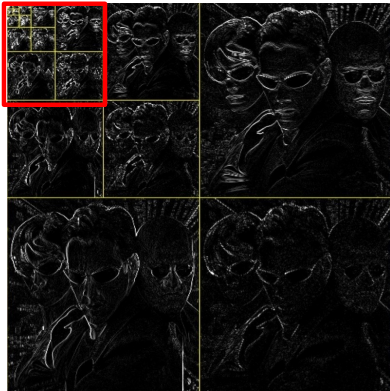
JPEG-2K Compression



JPEG-2K Compression



JPEG-2K Compression



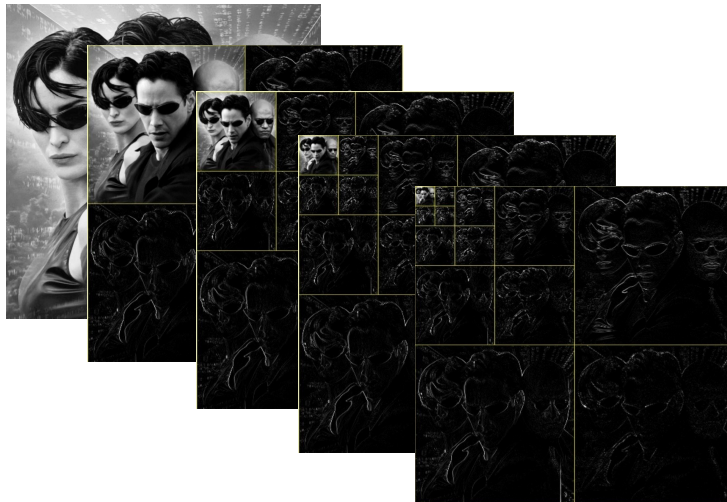
JPEG-2K Compression



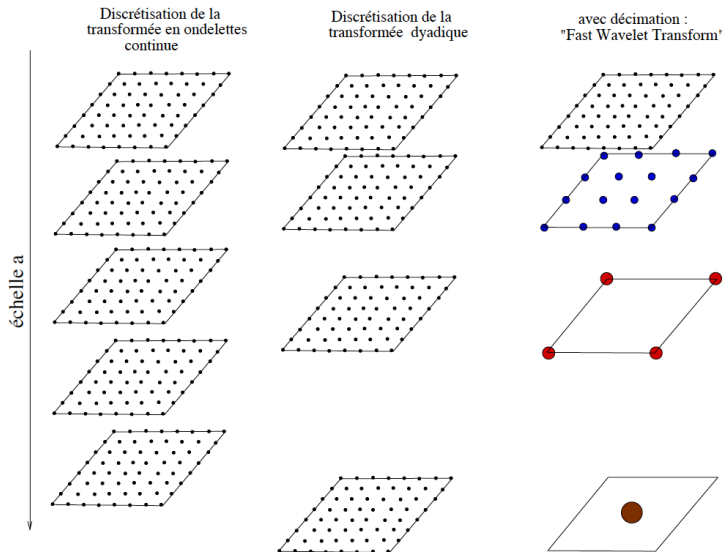
JPEG-2K Compression



JPEG-2K Compression



CWT-2D sampled vs. FWT-2D



Credits: Le Cadet

Take home message

Question: CWT (discretized) or DWT?

Answer: depends on the application

- **CWT** for feature detection (no *a priori* choice for a, b): more flexible, more robust to noise, but only **frames** in general.
- **DWT** for large amount of data, data compression: **bases**, faster, but more rigid (need generalization)

Generalizations

- Biorthogonal wavelets
- Wavelet packets
- Continuous wavelet packets (integrated wavelets)
- Redundant WT (on a rectangular lattice)
- Second generation wavelets (lifting scheme)
- ...