TELECOM ParisTech
Institut
Mines-Telecom

Wavelet-based image compression

F. Dufaux, M. Cagnazzo

Sigma 207

Outline

Outline

Introduction

Discrete wavelet transform and multiresolution analysis

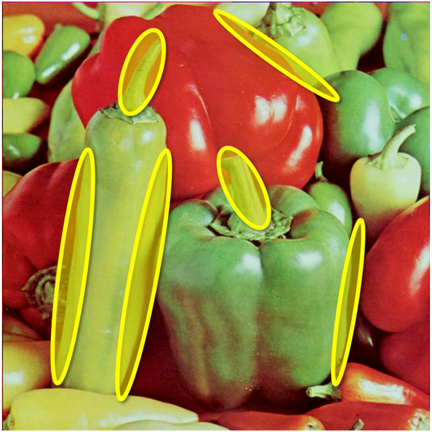
Images compression with wavelets

Wavelets and images : Motivations

- Image model : *trends* + *anomalies*

Wavelets and images : Motivations

- Image model : *trends* + *anomalies*



Wavelets and images : Motivations

- Image model : *trends* + *anomalies*

Wavelets and images : Motivations

- *Anomalies* :
  - Abrupt variations of the signal
  - High frequency contributions
  - Objects' contours
  - Good spatial resolution
  - Rough frequency resolution
- *Trends* :
  - Slow variations of the signal
  - Low frequency contributions
  - Objects' texture
  - Rough spatial resolution
  - Good frequency resolution

Wavelets and images : Motivations

Time-frequency boxes of basis signals

Time analysis:
$\theta_\gamma(t) = \delta(t - t_0)$
Frequency analysis:
$\theta_\gamma(t) = e^{2i\pi f_0 t}$
Short Time Fourier Transform (STFT):

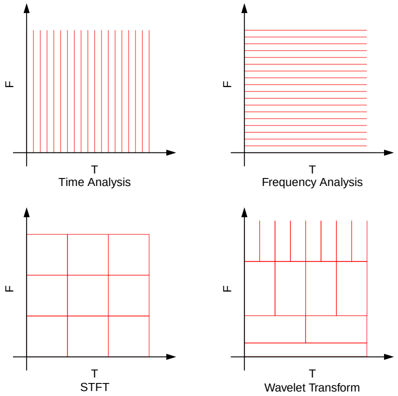
$$\theta_\gamma(t) = g(t - t_0)e^{2i\pi f_0 t}$$

σ_t, σ_ξ independent from γ

Wavelet Transform:

$$\theta_\gamma(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)$$

$\sigma_{t(a,b)} = a\sigma_{t,(1,0)} \quad \sigma_{\xi(a,b)} = a^{-1}\sigma_{\xi,(1,0)}$



Wavelets and images : Motivations

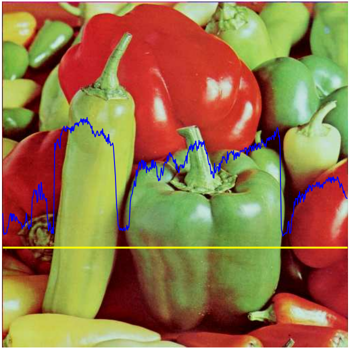
Signal model: an image row

Wavelets and images : Motivations

Signal model: an image row

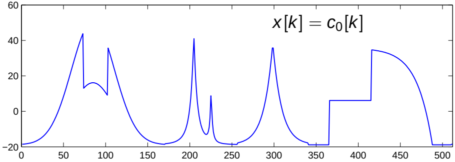
Wavelets and images : Motivations

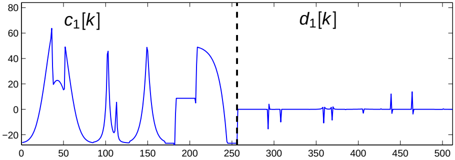
Signal model: an image row



Wavelets and Multiple resolution analysis

- Approximation: low resolution version
- "Details": zeros when the signal is polynomial





Outline

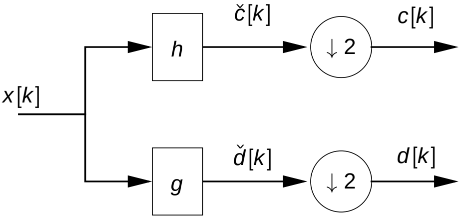
Introduction

Discrete wavelet transform and multiresolution analysis
- Filter banks and DWT
- Multiresolution analysis

Images compression with wavelets

1D filter banks

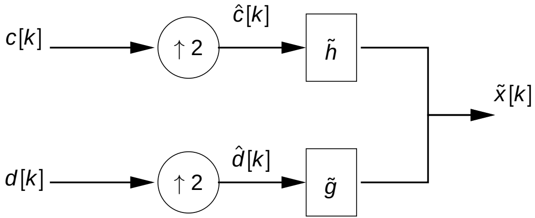
Decomposition



Analysis filter bank

$2\downarrow$: decimation : $c[k] = \check{c}[2k]$

Reconstruction



Synthesis filter bank

$2\uparrow$: interpolation operator, doubles the sample number

$$\hat{c}[k] = \begin{cases} c[k/2] & \text{if } k \text{ is even} \\ 0 & \text{if } k \text{ is odd} \end{cases}$$

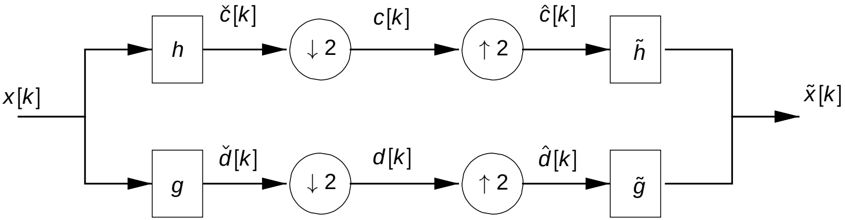
Filter properties

- Perfect reconstruction (PR)
- FIR
- Orthogonality
- Vanishing moments
- Symmetry

Perfect reconstruction conditions

We want PR after synthesis and analysis filter banks :
$\forall k \in \mathbb{Z}$,

$$\widetilde{x}_k = x_{k+\ell} \Longleftrightarrow \widetilde{X}(z) = z^{-\ell} X(z)$$



Z-domain relationships

filter $$\check{C}(z) = \sum_{n=-\infty}^{\infty} \check{c}_n z^{-n} = H(z)X(z)$$

decimation $$C(z) = \frac{1}{2}\left[\check{C}\left(z^{1/2}\right) + \check{C}\left(-z^{1/2}\right)\right]$$

interpolation $$\hat{C}(z) = C\left(z^2\right)$$

output $$\widetilde{X}(z) = \widetilde{H}(z)C\left(z^2\right) + \widetilde{G}(z)D\left(z^2\right)$$

$$\widetilde{X}(z) = \frac{1}{2}\left[\widetilde{H}(z)H(z) + \widetilde{G}(z)G(z)\right]X(z) + \frac{1}{2}\left[\widetilde{H}(z)H(-z) + \widetilde{G}(z)G(-z)\right]X(-z)$$

PR conditions in Z

$\forall k \in \mathbb{Z}$,

$$\widetilde{x}_k = x_{k+\ell} \Longleftrightarrow \widetilde{X}(z) = z^{-\ell} X(z)$$

$$\Updownarrow$$

$$\widetilde{H}(z)H(z) + \widetilde{G}(z)G(z) = 2z^{-\ell} \qquad \text{Non distortion}$$

$$\widetilde{H}(z)H(-z) + \widetilde{G}(z)G(-z) = 0 \qquad \text{Non aliasing}$$

Perfect reconstruction conditions

Matrix form

For simplicity, we ignore the delay, $\ell = 0$
If the analysis filter bank is given, the synthesis one is determined by:

$$\begin{bmatrix} H(z) & G(z) \\ H(-z) & G(-z) \end{bmatrix} \cdot \begin{bmatrix} \widetilde{H}(z) \\ \widetilde{G}(z) \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

We assume that the *modulation matrix* is invertible.

Perfect reconstruction conditions

Synthesis filter bank

Modulation matrix determinant :

$$\Delta(z) = H(z)G(-z) - G(z)H(-z)$$

$$\widetilde{H}(z) = \frac{2}{\Delta(z)}G(-z)$$

$$\widetilde{G}(z) = -\frac{2}{\Delta(z)}H(-z)$$

Perfect reconstruction with FIR filters

Finite impulse response filters:
It can be shown that in this case the PR condition is equivalent to the alterning signs condition. Example:

$$h(k) = \boxed{a}\,\boxed{b}\,\boxed{c} \qquad \widetilde{h}(k) = \boxed{p}\,\boxed{-q}\,\boxed{r}\,\boxed{-s}\,\boxed{t}$$

$$g(k) = \boxed{p}\,\boxed{q}\,\boxed{r}\,\boxed{s}\,\boxed{t} \qquad \widetilde{g}(k) = \boxed{-a}\,\boxed{b}\,\boxed{-c}$$

Orthogonality

Orthogonality assures energy conservation:

$$\sum_{k=-\infty}^{\infty} (x_k)^2 = \sum_{k=-\infty}^{\infty} (c_k)^2 + \sum_{k=-\infty}^{\infty} (d_k)^2$$

$\Rightarrow$ reconstruction error = quantization error on DWT coefficients
For non orthogonal filters, the reconstruction errors is a weighted sum of the quantization errors on the DWT subbands, with suitable weights ω_i

Vanishing moments

- Vanishing moments (VM) represent filter ability to reproduce polynomials: a filter with p VM can represent polynomials with degree $< p$
- The High-pass filter will not respond to a polynomial input with degree $< p$
- In this case all the signal information is preserved in the approximation signal (half the samples)
- A filter with p VM has at least $2p$ taps

Borders problem

- Filterbank properties such as we saw, are valid for infinite-size signals
- We are interested in finite support signals
- How to interpret the previous results for finite support signals?

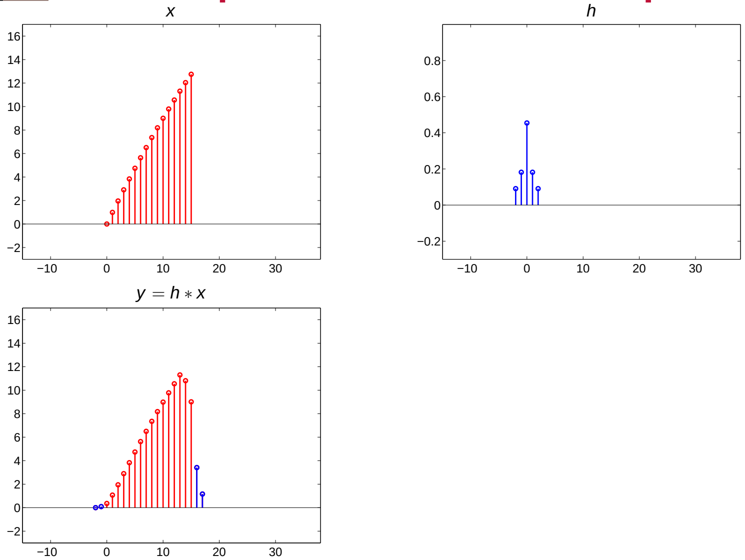
Borders problem

- Filterbank properties such as we saw, are valid for infinite-size signals
- We are interested in finite support signals
- How to interpret the previous results for finite support signals?
- Zero padding would introduce a coefficient expansion
- Filtering an N-size signal with an M-size produces a signal with size $N + M - 1$

Borders problem

- Filterbank properties such as we saw, are valid for infinite-size signals
- We are interested in finite support signals
- How to interpret the previous results for finite support signals?
- Zero padding would introduce a coefficient expansion
- Filtering an N-size signal with an M-size produces a signal with size $N + M - 1$
- Periodization?
- Symmetrization?

Borders problem: Coefficient expansion

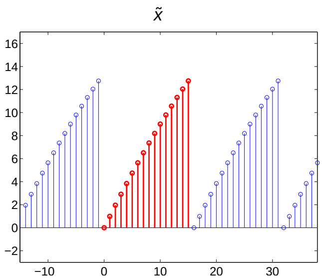


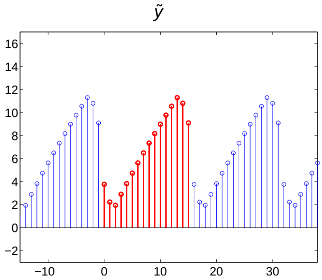
TELECOM ParisTech

Borders problem: Periodization

- A signal x of support N is considered as a periodic signal $\tilde{x}$ of period N
- Filtering $\tilde{x}$ with h results into a periodic output $\tilde{y}$
- $\tilde{y}$ has the same period N as $\tilde{x}$
- So we need to compute just N samples of $\tilde{y}$
- However, periodization introduces “jumps” in a regular signal

Borders problem: Periodization





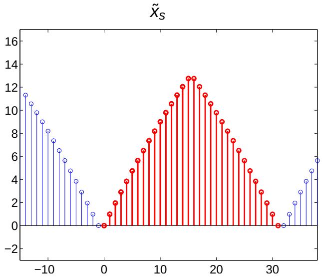
Borders problem: Symmetry

- Symmetrization before periodization reduces the impact on signal regularity
- But it doubles the number of coefficients...

Borders problem: Symmetry

- Symmetrization before periodization reduces the impact on signal regularity
- But it doubles the number of coefficients...
- Unless the filters are symmetric, too
  - We use x as half-period of $\tilde{x}_s$
  - $\tilde{x}_s$ has a period of $2N$ samples
  - Filtering $\tilde{x}_s$ with h, produces $\tilde{y}_s$
  - If h is symmetric, $\tilde{y}_s$ is periodic and symmetric, with period $2N$: we only need to compute N samples

Borders problem: Symmetry



Haar filter

$$h(k) = \boxed{1}\,\boxed{1} \qquad \tilde{h}(k) = \boxed{1}\,\boxed{1}$$

$$g(k) = \boxed{1}\,\boxed{-1} \qquad \tilde{g}(k) = \boxed{-1}\,\boxed{1}$$

- Symmetric
- Orthogonal
- VM = 1
  - Only capable to represent piecewise constant signals

Summary: perfect reconstruction and borders

- Convolution involves coefficient expansion
- Solution: circular convolution
  - Circular convolution allows to reconstruct an N-samples signal with N wavelet coefficients
  - The periodization generates borders discontinuities, i.e. spurious high frequencies coefficients that demand a lot of coding resources
- Solution: Symmetric periodization
  - No discontinuities
  - Does it double the coefficient number?
  - No, if the filter is **symmetric**!

Bad news: the only orthogonal symmetric FIR filter is Haar!

Biorthogonal filters

Cohen-Daubechies-Fauveau filters

With biorthogonal (i.e. PR) filters, if h has p VM and $\widetilde{h}$ has $\widetilde{p}$ VM, the filter has at least $p+\widetilde{p}-1$ taps.

The CDF filters have the following properties:

- They are symmetric (linear phase)
- They maximize the VM for a given filter length
- They are close to orthogonality (weights ω_i are close to one)

They are by far the most popular in image compression

9/7 biorthogonal filters

Filter coefficients:

n	0	± 1	± 2	± 3	± 4
$h[l]$	0.852699	0.377403	-0.110624	-0.023849	0.037828
$\widetilde{h}[l]$	0.788486	0.418092	-0.040689	-0.064539	

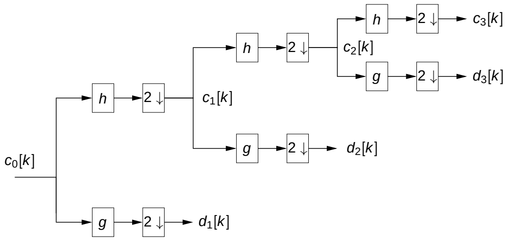
Impulse response of low-pass filters

For high pass filters, we have (alterning sign condition):

$$g[l] = (-1)^{l+1}\,\widetilde{h}\,[l-1] \qquad \text{and} \qquad \widetilde{g}[l] = (-1)^{l-1}\,h\,[l+1]\,.$$

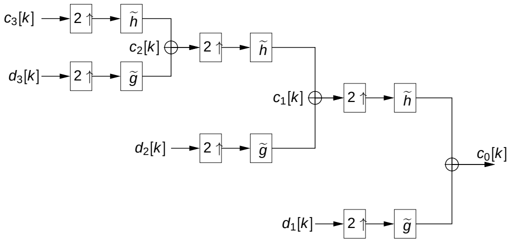
1D Multiresolution analysis

Decomposition



Three level wavelet decomposition structure

Reconstruction

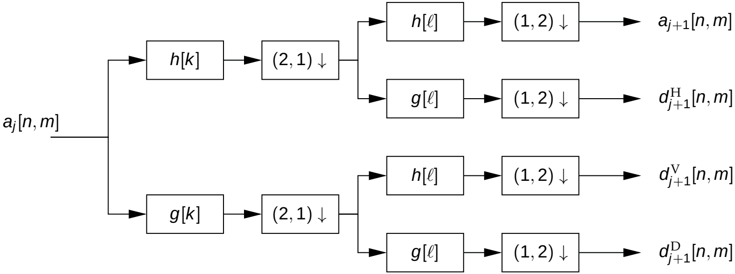


Reconstruction from wavelet coefficients

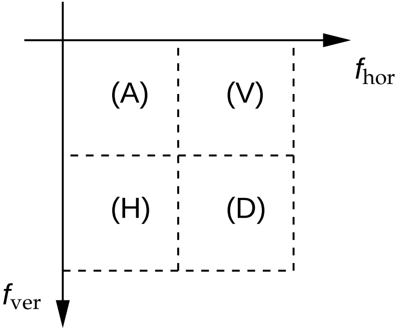
2D AMR

2D Filter banks for separable transform

One decomposition level

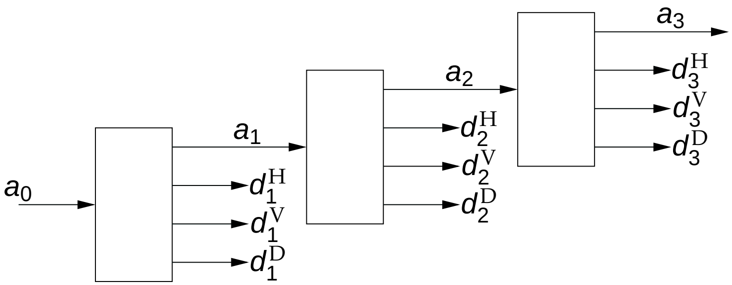


2D-DWT subbands: orientations



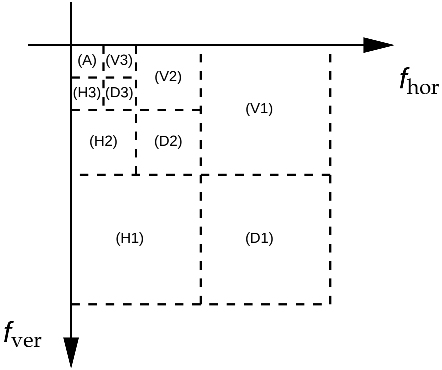
(A), (H), (V) and (D) respectively correspond to approximation coefficients, horizontal, vertical and diagonal detail coefficients.

2D AMR: multiple levels

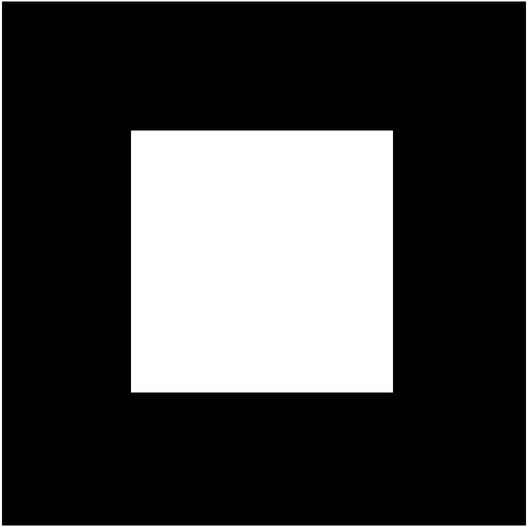


Three levels of separable 2D-AMR.

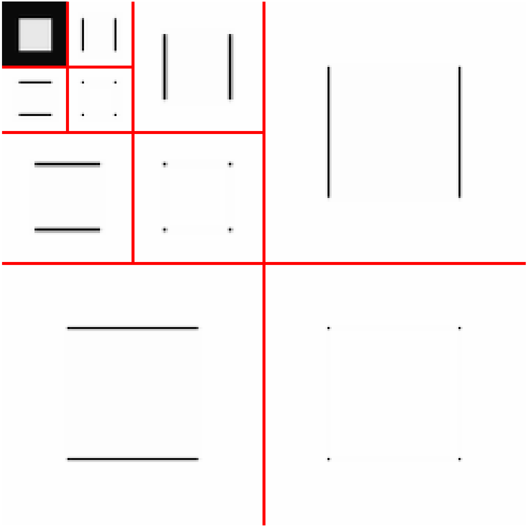
2D-DWT subbands: orientations



Example

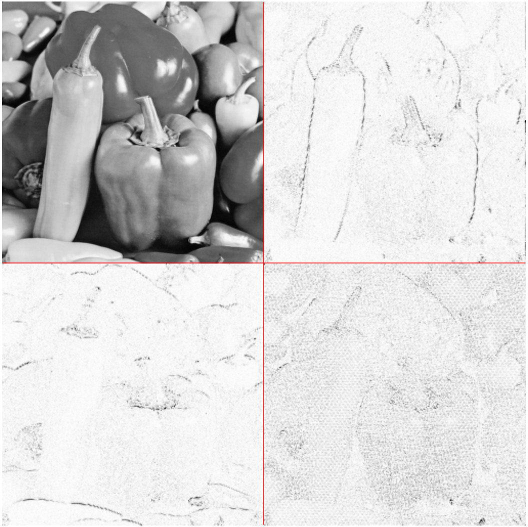


Example

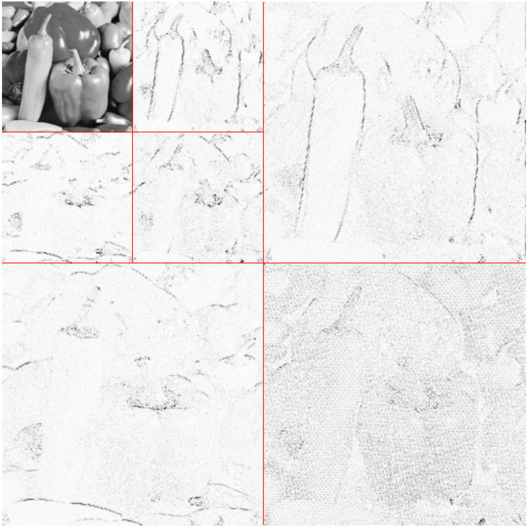


Example

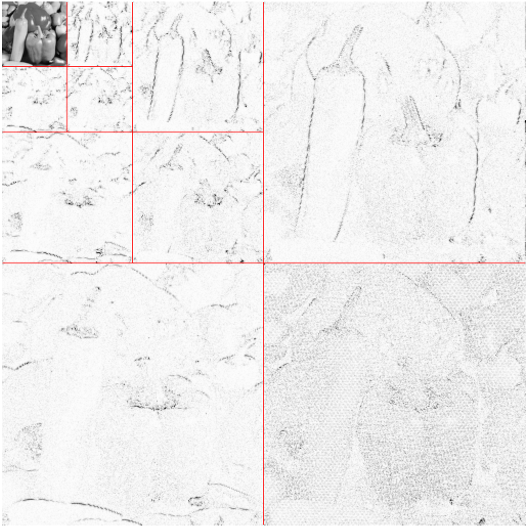
Example



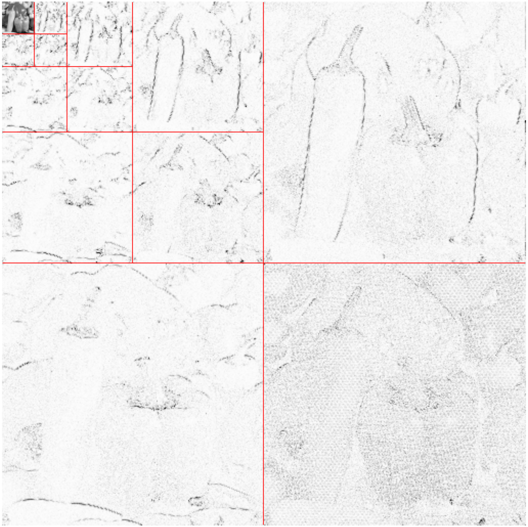
Example



Example



Example



Outline

Introduction

Discrete wavelet transform and multiresolution analysis

Images compression with wavelets
- EZW
- JPEG 2000

Compression with DWT

- Methods based on inter-scale dependencies:
  - EZW (Embedded Zerotrees of Wavelet coefficients),
  - SPIHT (Set Partitioning in Hierarchical Trees)
  - Tree-based representation of dependencies
  - Advantages: good exploitation of inter-scale dependencies, low complexity
  - Disadvantage: no resolution scalability
- Methods not based on inter-scale dependencies
  - Explicit bit-rate allocation among subbands
  - Entropy coding of coefficients
  - Advantages: Good exploitation of intra-scale dependencies, random access, resolution scalability
  - Disadvantage: no exploitation of inter-scale dependencies

Embedded Zerotrees of Wavelet coefficients

Main characteristics

- Quality scalability (i.e. progressive representation)
- Lossy-to-lossless coding
- Small complexity
- Rate-distortion performance much better than JPEG **above all at small rates**

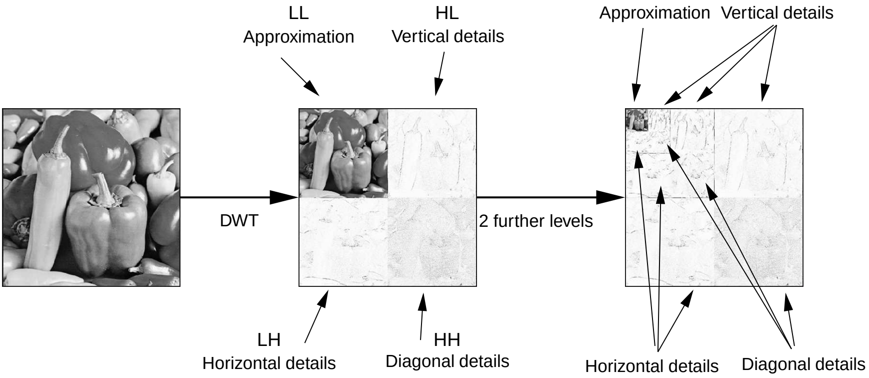
Progressive representation of DWT coefficients

- Each new coding bit must convey the maximum of information

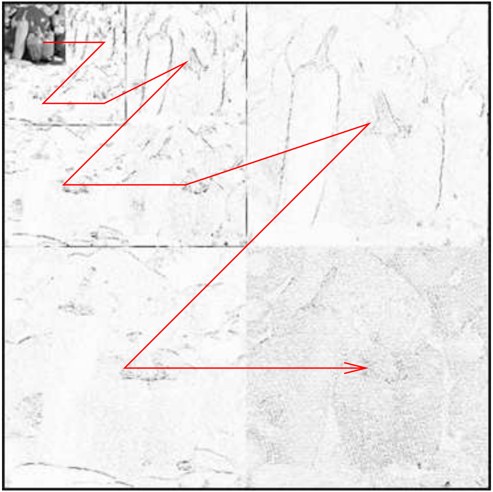
$$\Updownarrow$$

- Each new coding bit must reduce as much as possible distortion
- We first send the largest coefficients
- Problem: localization overhead

Example: an image and its wavelet coefficients



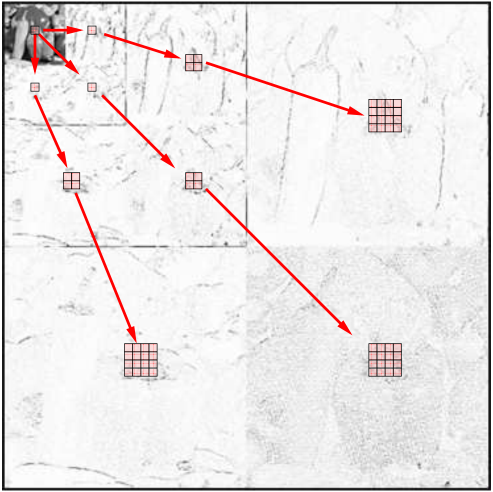
Progressive representation: subband order



EZW Algorithm

- The subband scan order alone is not enough to assure that largest coefficients are sent first
- We need to localize the largest coefficients
- Without having to send explicit localization information
- Idea: to exploit the inter-band correlation to predict the position of non-significant coefficients
- If the prediction is correct we save many coding bits (for all the predicted coefficients)

Zero-tree of wavelet coefficients



EZW idea

- **Auto-similarity** : When a coefficient is small (below a threshold) it is probable that its descendants are small as well
- In this case we use a single coding symbol to represent the coefficient and all its descendants. If c and all its descendants are smaller than the threshold, c is called a zero-tree root
- With just one symbol, (ZT) we code $(1 + 4 + 4^2 + \ldots + 4^{N-n})$ coefficients
- The localization information is implicit in the significance information

EZW algorithm

1. $k = 0$
2. $n = \lfloor \log_2 \left(|c|_{\max} \right) \rfloor$
3. $T_k = 2^n$
4. while (rate < available rate)
   - Dominant pass
   - Refining pass
   - $T_{k+1} \leftarrow T_k / 2$
   - $k \leftarrow k + 1$
5. end while

Dominant pass

- For each coefficient c (in the scan order)
- If $|c| \geq T_n$, the coefficient is significant
  - If $c > 0$ we encode SP (Significant Positive)
  - If $c < 0$ we encode SN (Significant Negative)
- If $|c| < T_n$, we compare all its descendants with the threshold
  - If no descendant is significant, c is coded as a zero-tree root (ZT)
  - Otherwise the coefficient is coded as Isolated Zero (IZ)

Refining pass

- We encode a further bit for all significant coefficients
- This is equivalent to halve the quantization step

Iteration and termination

- The k-th dominant pass allows to encode the k-th bit-plane
- A significant coefficient c is such that $2^k < |c| < 2^{k+1}$
- For the next step we halve the threshold: it is equivalent to pass to the next bitplane
- Algorithm stops when
  - the bit budget is exhausted; or when
  - all the bitplanes have been coded

EZW Algorithm: summary

- Bitplane coding: at the k-th pass, we encode the bitplane $\log_2 T_k$
- Progressive coding: each new bitplane allows refining the coefficients quantization
- Lossless coding of significance symbols
- Lossless-to-lossy coding: When an integer transform is used, and all the bitplanes are coded, the original image can be restored with zero distortion

EZW Algorithm: Example

26	6	13	10
-7	7	6	4
4	-4	4	-3
2	-2	-2	0

$$T_0 = 2^{\lfloor \log_2 26 \rfloor} = 16$$

EZW Algorithm: Example

26	6	13	10
-7	7	6	4
4	-4	4	-3
2	-2	-2	0

$$T_0 = 2^{\lfloor \log_2 26 \rfloor} = 16$$

Bitstream:

SP	ZR	ZR	ZR	1								
IZ	ZR	ZR	ZR	SP	SP	IZ	IZ	0	1	0		
SP	IZ	SP	SI	SP	SP	SN	IZ	IZ	SP	IZ	IZ	IZ

JPEG2000

- JPEG2000 aims at challenges unresolved by previous standards:
- Low bit-rate coding: JPEG has low quality for $R < 0.25$ bpp
- Synthetic images compression
- Random access to image parts
- Quality and resolution scalability

New functionalities

- Region-of-interest (ROI) coding
- Quality and resolution scalability
- Tiling
- Exact coding rate
- Lossy-to-lossless coding

Algorithm

JPEG2000 is made up of two *tiers*

- First tier
  - DWT and quantization
  - Lossless coding of codeblocks
- Second tier
  - **EBCOT**: embedded block coding with optimized truncation
  - Scalability (quality, resolution) and ROI management

Quantization in JPEG2000

- DWT coefficients are encoded with a very fine quantization step
- For the lossless coding case, DWT coefficients are integers, and they are not quantified
- In summary, it is not in the quantization step that the really lossy operations are performed
- The lossy coding is performed by the bitstream truncation of Tier 2

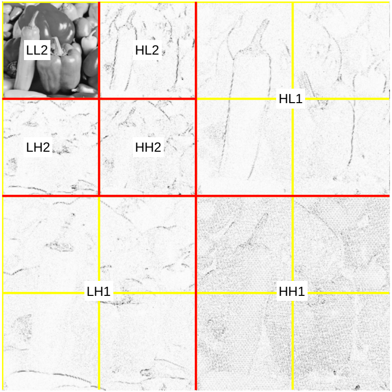
EBCOT

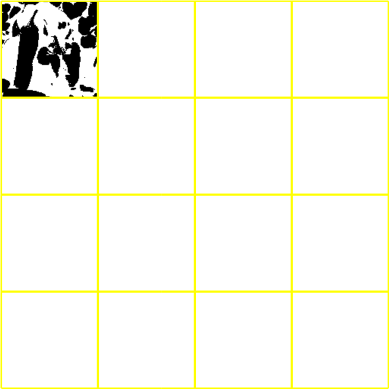
Embedded Block Coding with Optimized Truncation

- Each subband is split in equally sized blocks of coefficients, called codeblocks
- The codeblocks are losslessly and independently coded with an arithmetic coder
- We generate as much bitstreams as codeblocks in the image

Bitplane coding

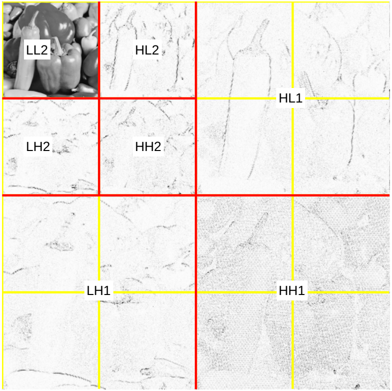
Most significant bitplane

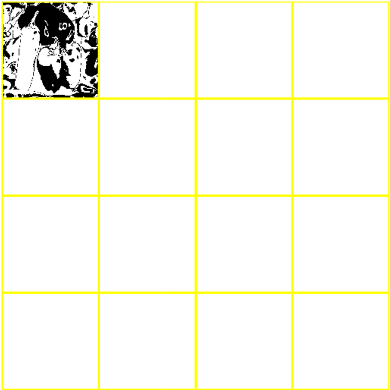




Bitplane coding

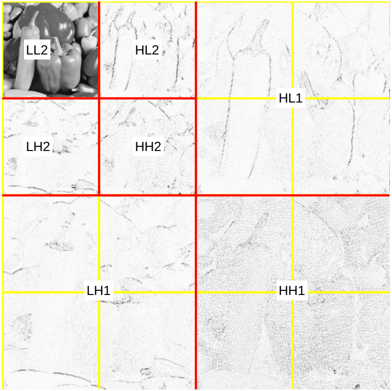
Second bitplane

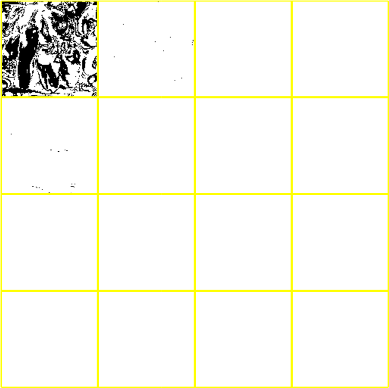




Bitplane coding

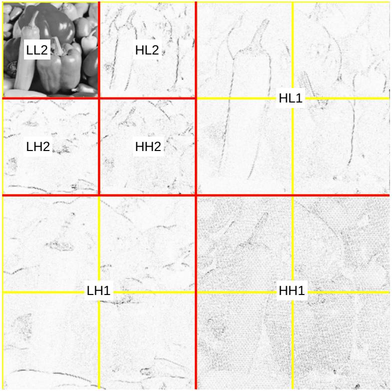
Third bitplane

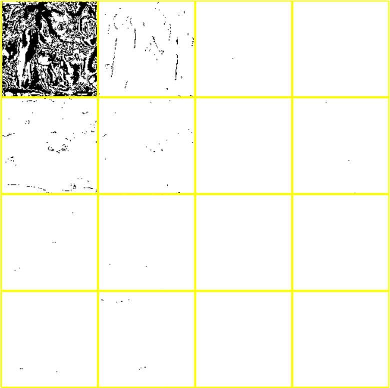




Bitplane coding

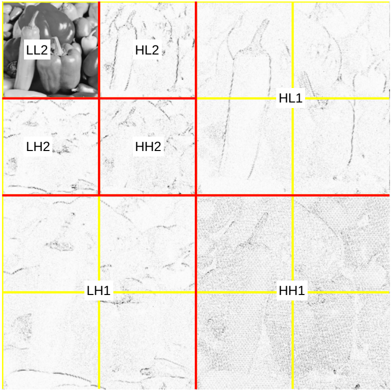
Fourth bitplane

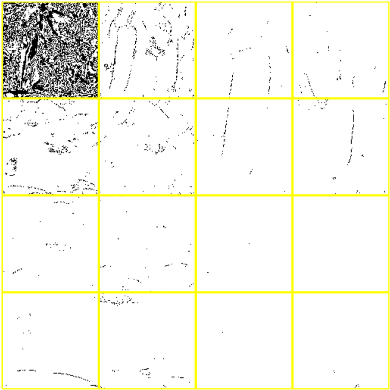




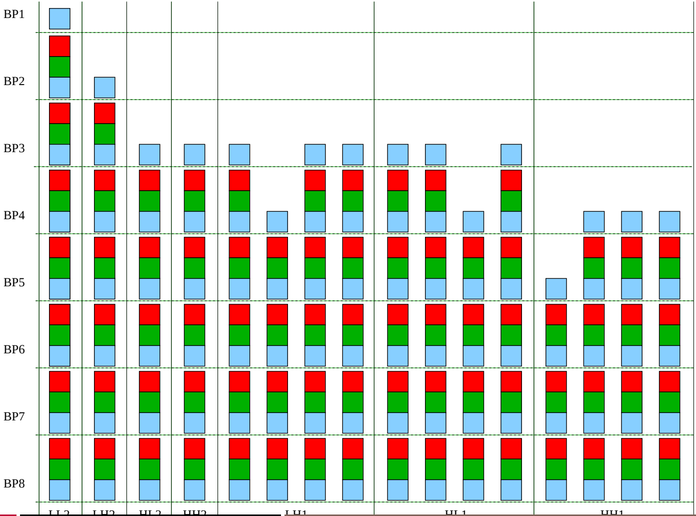
Bitplane coding

Fifth bitplane





Example of bitstreams associated to codeblocks



EBCOT

Optimization

- If we keep all the bitstreams of all the codeblocks, we end up with a huge bitrate
- We have to truncate the bitstream to attain the target bit-rate
- Problem: how to truncate the bitstreams with a minimum resulting distortion?

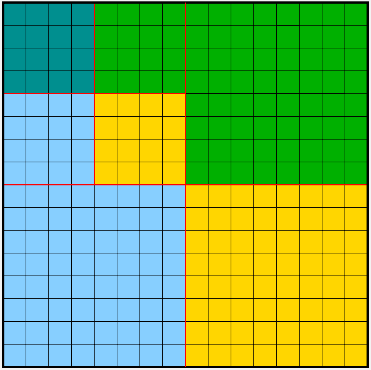
$$\min \sum_i D_i \qquad \text{subject to} \qquad \sum_i R_i \leq R_{\mathrm{tot}}$$

- Solution : Lagrange multiplier

$$J = \sum_i D_i + \lambda \left(\sum_i R_i - R \right)$$

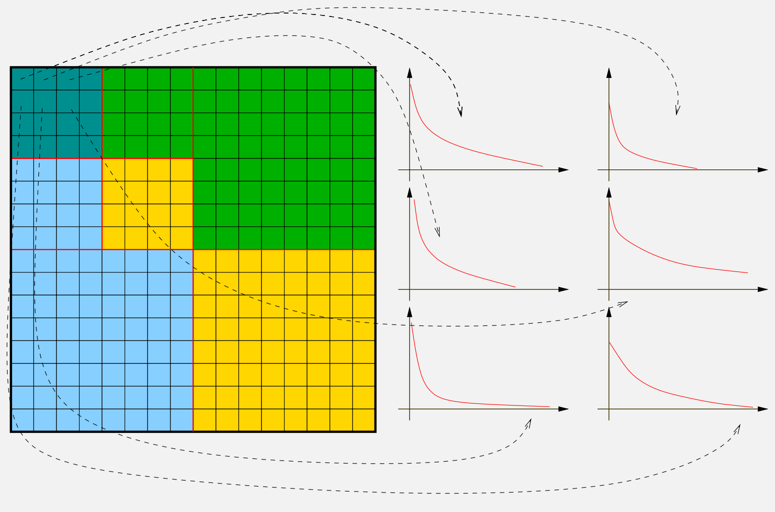
EBCOT

Rate-distortion curve per each codeblock



EBCOT

Rate-distortion curve per each codeblock



EBCOT

Embedded block coding with optimized truncation

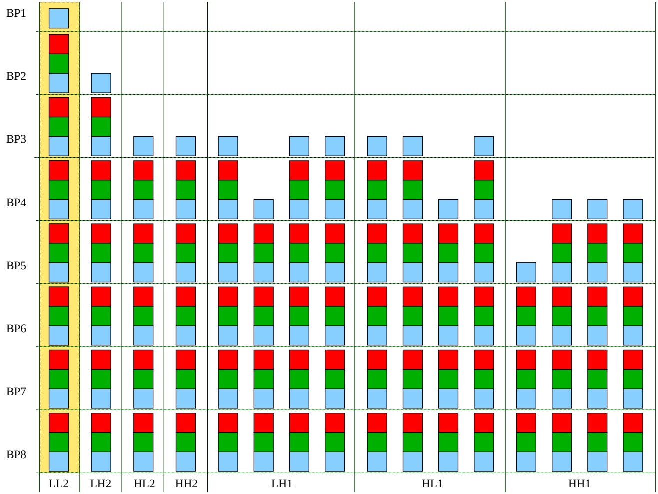
- Optimal truncation point:

$$\frac{\partial D_i}{\partial R_i} = -\lambda$$

- The value of the Lagrange multiplier can be find by an iterative algorithm.
- We can have several truncations for several target rates (quality scalability)

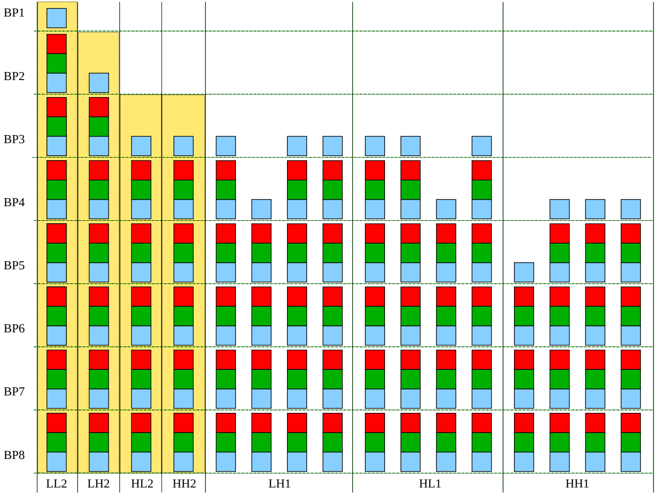
Example of bit-rate allocation with EBCOT

Allocation for maximal quality and minimal resolution



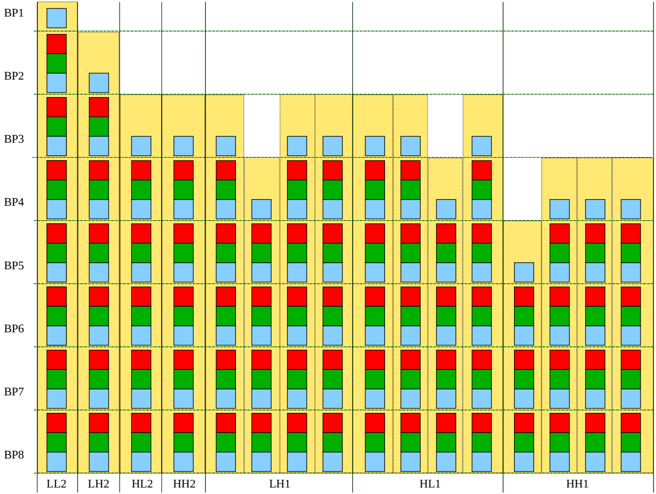
Example of bit-rate allocation with EBCOT

Allocation for maximal quality and medium resolution



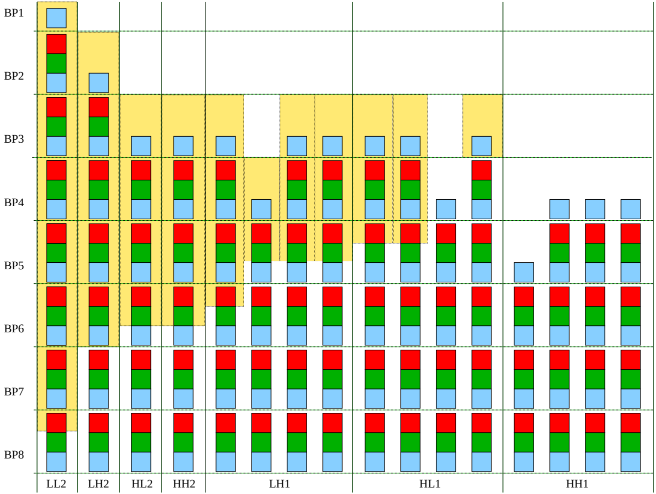
Example of bit-rate allocation with EBCOT

Allocation for maximal quality and maximal resolution



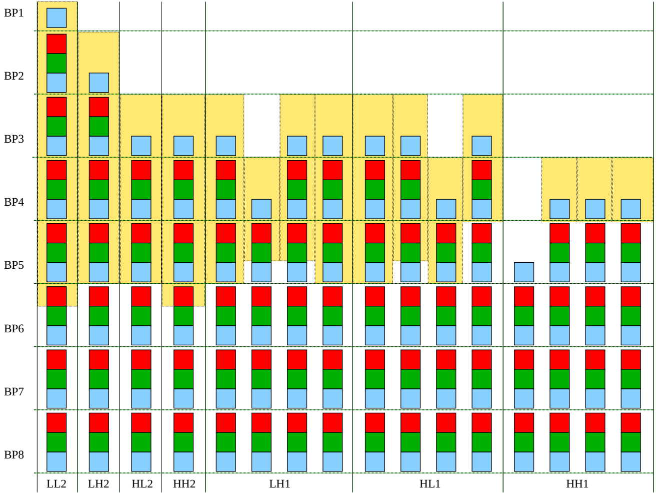
Example of bit-rate allocation with EBCOT

Allocation for perceptual quality and maximal resolution



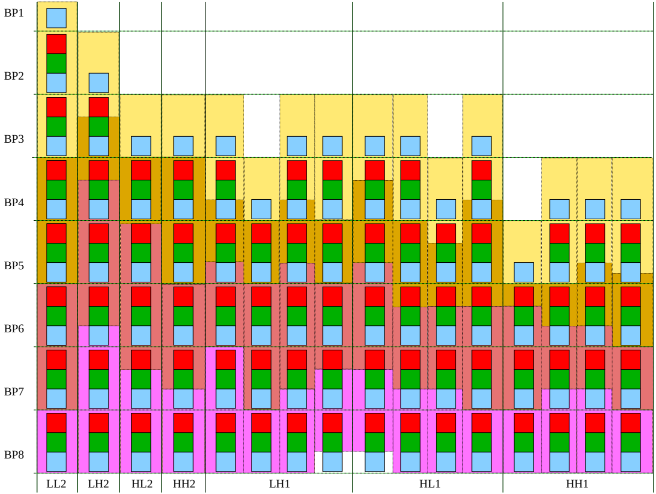
Example of bit-rate allocation with EBCOT

Allocation for a given bit-rate, maximal quality and resolution



Example of bit-rate allocation with EBCOT

Allocation pour several layers and maximal resolution



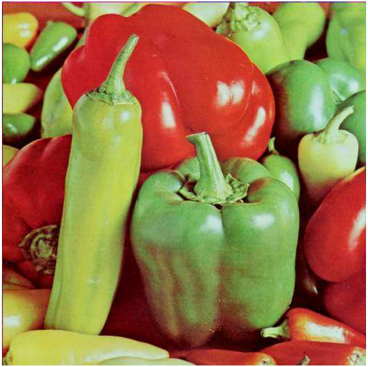
JPEG

Comparison JPEG / JPEG2000

Image Originale, 24 bpp

Comparison JPEG / JPEG2000

Rate: 1bpp



Comparison JPEG / JPEG2000

Rate: 0.75bpp

Comparison JPEG / JPEG2000

Rate: 0.5bpp

Comparison JPEG / JPEG2000

Rate: 0.3bpp

Comparison JPEG / JPEG2000

Rate: 0.2bpp

Comparison JPEG / JPEG2000

Rate: 0.2bpp pour JPEG, 0.1 pour JPEG2000


Error effect: JPEG

JPEG, $p_E = 10^{-4}$

JPEG, $p_E = 10^{-4}$

Error effect: JPEG and JPEG 2000

JPEG, $p_E = 10^{-4}$

JPEG 2000, $p_E = 10^{-4}$

Error effect: JPEG and JPEG 2000

JPEG, $p_E = 10^{-3}$

JPEG 2000, $p_E = 10^{-3}$

Image coding and robustness

- Markers insertion
- Markers period
- Marker emulation prevention
- Trade-off between robustness and rate

Error robustness in JPEG2000

- Data priorization is possible
- No dependency among codeblocks
  - No error propagation
- No block-based transform
  - No *blocking* artifacts