Introduction to information theory and coding

Louis WEHENKEL

Set of slides No 6

Overview of (irreversible) image compression

- Motivations
- Image representations
- Sources of redundancy
- Image compression systems
- Brief introduction to wavelets

Motivations for image compression (and to some extent for sound compression)

Avantage of digital image representations : immunity to noise

Disadvantages : huge volumes of data if not compressed

Examples

A high resolution image $=$ some MB.

A video sequence : ≈ 20 images/s : 1minute $\approx$ 1GB.

Present day image compression techniques

$\Rightarrow$ compression rates $\rightarrow$ 100.

$\Rightarrow$ makes possible what would be impossible otherwise.

$\Rightarrow$ image processing : one of the most used techniques in many fields (more and more).

$\Rightarrow$ Multimedia DB, medical applications, legal, digital archives...

What's a (monochrome) image ?

Mathematical model

Positive, real-valued function of two arguments

$f(x,y)\ :[0,x_{max}]\times[0,y_{max}]\longrightarrow[0,f_{max}],$

Sampled version :

$f(x,y)$ = $N \times M$ matrix : x = line index, y = column index.

NB: to simplify $N \times N$. **(**(x,y) = ***pixel*****)**

Discretized (quantized) version : $f(x,y)$ = integer number with fixed number of bits.

Examples: photo, a component of a color image, a function of two variables (scalar field)

Stochastic image models

They exist (e.g. Markov fields), but we will not talk about them

Image transforms

NB: generalization of the Fourier transform

Goal: represent image in a way well suited for a class of operations.

E.g.: Fourier transform “makes easy” linear operations (convolution).

Here: goal $=$ facilitate data compression (reversible or irreversible).

Reminder (in the temporal domain = unidimensional, sampled)

$$t(w) = \sum_{t=0}^{N-1} f(t)g(t,w)$$

$g(t,w)$ = kernel (family of N basis functions indexed by w)

Inverse transform (when it exists) $f(t) = \sum_{w=0}^{N-1} t(w)h(t,w)$.

Vector representation :

$g(t, w)$ = $N \times N$ matrix, $f(t)$ and $t(w)$ line vectors.

$\Rightarrow$ transformation = matrix product

$\boldsymbol{t} = \boldsymbol{f}\boldsymbol{G}$, $\boldsymbol{f} = \boldsymbol{t}\boldsymbol{H} \Rightarrow \boldsymbol{H} = \boldsymbol{G}^{-1}$

Orthogonal Bases : orthogonal matrices $\boldsymbol{G}^{-1} = \boldsymbol{G}^{T}$.

In the complex case : $\boldsymbol{G}^{-1} = \boldsymbol{G}^{*}$.

$\Rightarrow$ transformation = change of basis

NB: continuous case $\sum_0^{N-1} \rightarrow \int_0^T$...

Anyway : transformation = **linear operation**
$\Rightarrow$ transform of linear combination = linear combination of transforms.

Physical interpretation :

$t(w)$ measures similarity of $f(t)$ and $g(t, w)$ (analog to dictionary match)

Generalization to images

The transform $\mathcal{G}$ of an image $f(x, y)$ (dimensions $N \times N$) using *kernel* $g(\cdot, \cdot, \cdot, \cdot)$ is the new image $N \times N$

$$\mathcal{G}(f) = T(u, v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) g(x, y, u, v). \qquad (1)$$

The transform is non-singular, and has an inverse transform $\mathcal{H}$ which kernel is $h(\cdot, \cdot, \cdot, \cdot)$, if $\forall f(x, y)$

$$f(x, y) = \mathcal{H}(\mathcal{G}(f)) = \mathcal{H}(T) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} T(u, v) h(x, y, u, v). \qquad (2)$$

The functions $g(i, j, \cdot, \cdot)$ and $h(\cdot, \cdot, i, j)$ may be interpreted as a set of N^2 "basis functions" in a series expansion.

Construction of image transforms

Kernel $g(\cdot,\cdot,\cdot,\cdot)$ is separable if $g(x,y,u,v) = g_1(x,u)g_2(y,v)$.
It is said to be symmetric if we can take $g_1(\cdot,\cdot) = g_2(\cdot,\cdot)$. (Same for h)

$\Rightarrow$ Multi-dimensional transforms are obtained by successive applications of unidimensional ones.

One first transforms the N lines, then the N columns of the result : $\boldsymbol{T} = \boldsymbol{G}^T \boldsymbol{F}\boldsymbol{G}$.

$$T(u,v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)g(x,y,u,v) \quad (3)$$

$$= \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)g_1(x,u)g_1(y,v) \quad (4)$$

$$= \sum_{x=0}^{N-1} g_1(x,u)\left[\sum_{y=0}^{N-1} f(x,y)g_1(y,v)\right] \quad (5)$$

hence $\boldsymbol{T} = \boldsymbol{G}^T[\boldsymbol{F}\boldsymbol{G}] = \boldsymbol{G}^T\boldsymbol{F}\boldsymbol{G}$. (Invertible iff $\boldsymbol{G}$is non-singular)

Bi-dimensional Fourier transform

The Fourier transform uses the following kernel

$$g^F(x,y,u,v) = \frac{1}{N}\exp\left(\frac{-j2\pi(xu+yv)}{N}\right). \tag{6}$$

This kernel is separable, since $g^F(x,y,u,v) = g_1^F(x,u)g_1^F(y,v)$, with $g_1^F(x,u) = \frac{1}{\sqrt{N}}\exp\left(\frac{-j2\pi(xu)}{N}\right)$.

Since $ux = xu$ we have also $g_1^F(i,j) = g_1^F(j,i)$.

$\Rightarrow$ complex and symmetric matrix $\boldsymbol{G}$ ($\boldsymbol{G}^T = \boldsymbol{G}$).

In addition $\boldsymbol{G}$ orthogonal (unitary) :

$$\boldsymbol{G}^{-1} = \boldsymbol{G}^* = \overline{\boldsymbol{G}}. \tag{7}$$

The kernel of the inverse Fourier transform is hence

$$h^F(x,y,u,v) = \frac{1}{N}\exp\left(\frac{+j2\pi(xu+yv)}{N}\right). \tag{8}$$

Comments

Technique may be extended to more than 2 dimensions.

FFT algorithm may be used ($\Rightarrow \mathcal{O}(N \times N \log N)$ operations) : quasi-linear.

Sampling theorem : applies also. (choice of sampling intervals as a function of the frequency spectrum of the image)

Applications : signal processing...

Other transforms : Walsh and Hadamard

"Discrete Versions" of the Fourier transform.

Applicable if $N = 2^n$.

Values of the basis functions : $\pm\frac{1}{\sqrt{N}}$.

$\Rightarrow$ calculations simpler, physical interpretation similar.

Kernel of the Walsh (left) and Hadamard (right) transforms $(N = 8)$

Walsh (u across the top):

x	0	1	2	3	4	5	6	7
0	+	+	+	+	+	+	+	+
1	+	+	+	+	-	-	-	-
2	+	+	-	-	+	+	-	-
3	+	+	-	-	-	-	+	+
4	+	-	+	-	+	-	+	-
5	+	-	+	-	-	+	-	+
6	+	-	-	+	+	-	-	+
7	+	-	-	+	-	+	+	-

Hadamard (u across the top):

x	0	1	2	3	4	5	6	7
0	+	+	+	+	+	+	+	+
1	+	-	+	-	+	-	+	-
2	+	+	-	-	+	+	-	-
3	+	-	-	+	+	-	-	+
4	+	+	+	+	-	-	-	-
5	+	-	+	-	-	+	-	+
6	+	+	-	-	-	-	+	+
7	+	-	-	+	-	+	+	-

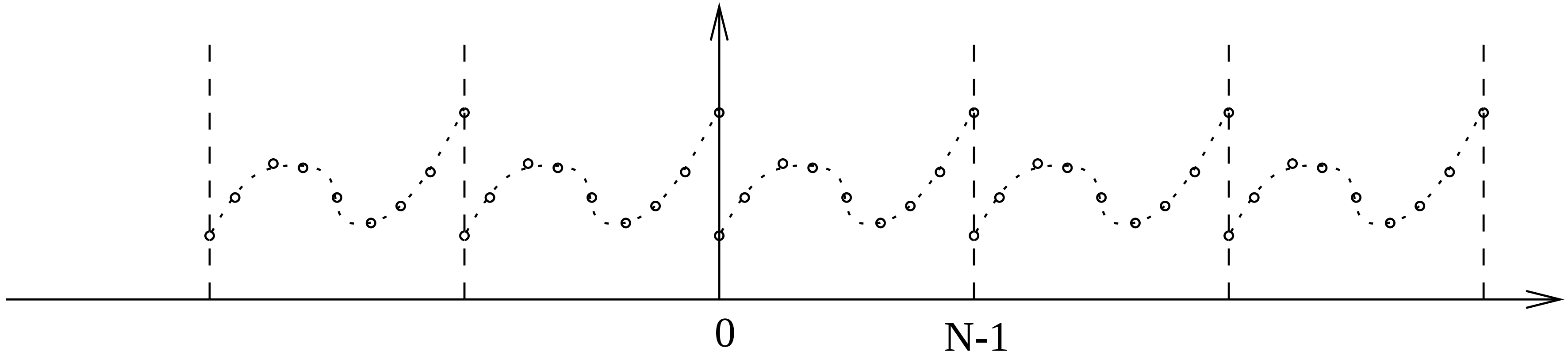
NB. Identical up to a permutation of lines and columns. Real-valued, orthogonal and symmetric $\Rightarrow$ inverse transform = direct transform.

Hadamard : $\boldsymbol{G} = \frac{1}{\sqrt{N}}\boldsymbol{H}_N$ where $\boldsymbol{H}_N$ can be generated recursively using the following "formula"

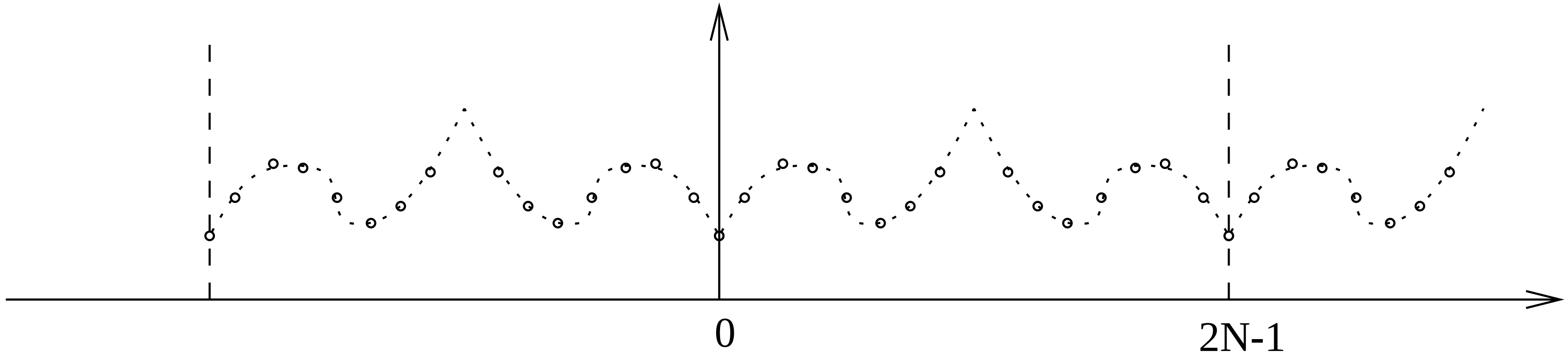
$$\boldsymbol{H}_{2^0} = [1]; \boldsymbol{H}_{2^n} = \begin{bmatrix} \boldsymbol{H}_{2^{n-1}} & \boldsymbol{H}_{2^{n-1}} \\ \boldsymbol{H}_{2^{n-1}} & -\boldsymbol{H}_{2^{n-1}} \end{bmatrix}. \tag{9}$$

Cosinus transform (used in JPEG format)

Problems with Fourier : complex values and border effects.
Fourier transform = series expansion of periodic extension of original signal



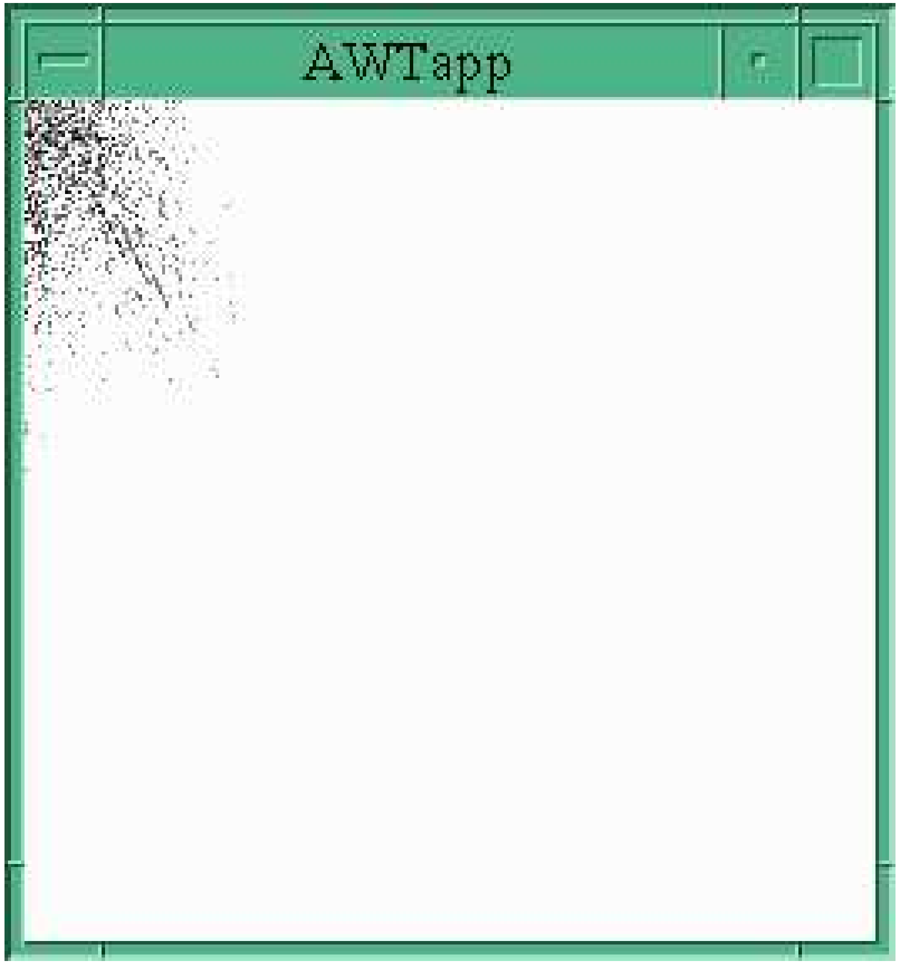
Cosinus transform : series expansion of following extension



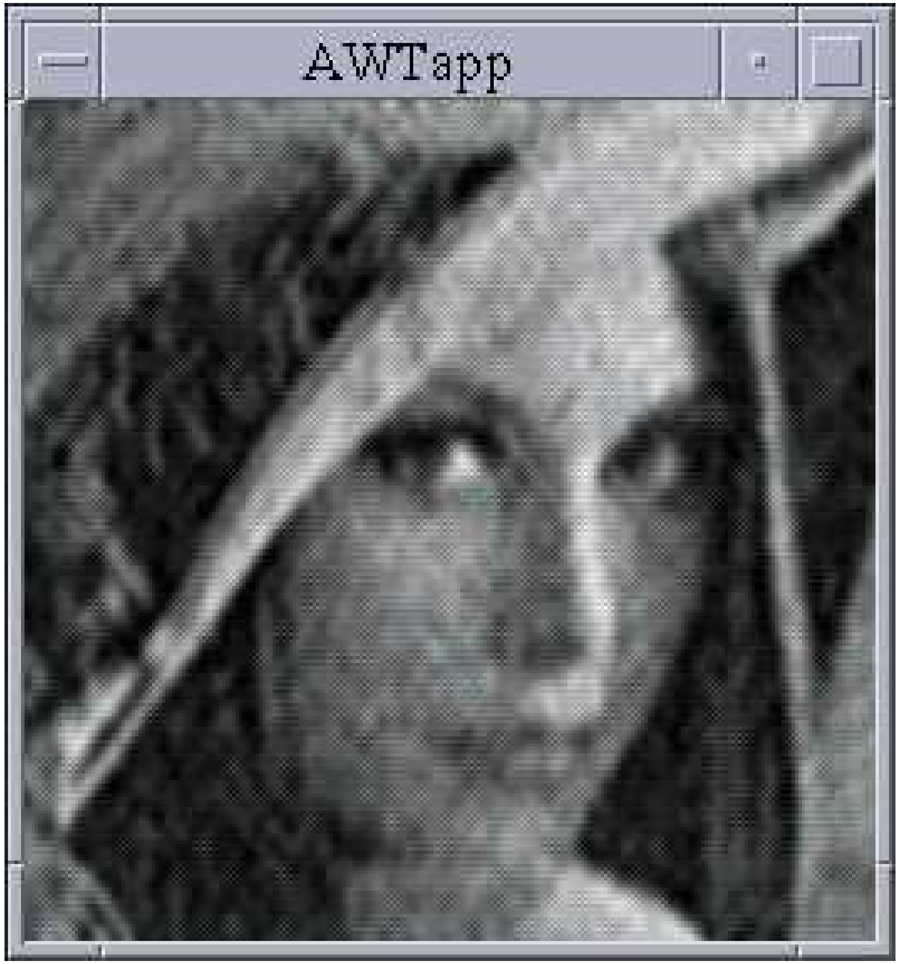
continuous and even...

$$g_1^C(x,u) = h_1^C(x,u) = \frac{1}{\sqrt{N}}\alpha(u)\cos\left(\frac{(2x+1)u\pi}{2N}\right) \quad (\alpha(0) = 1, \alpha(i) = \sqrt{2})$$

Cosinus transform of Lena

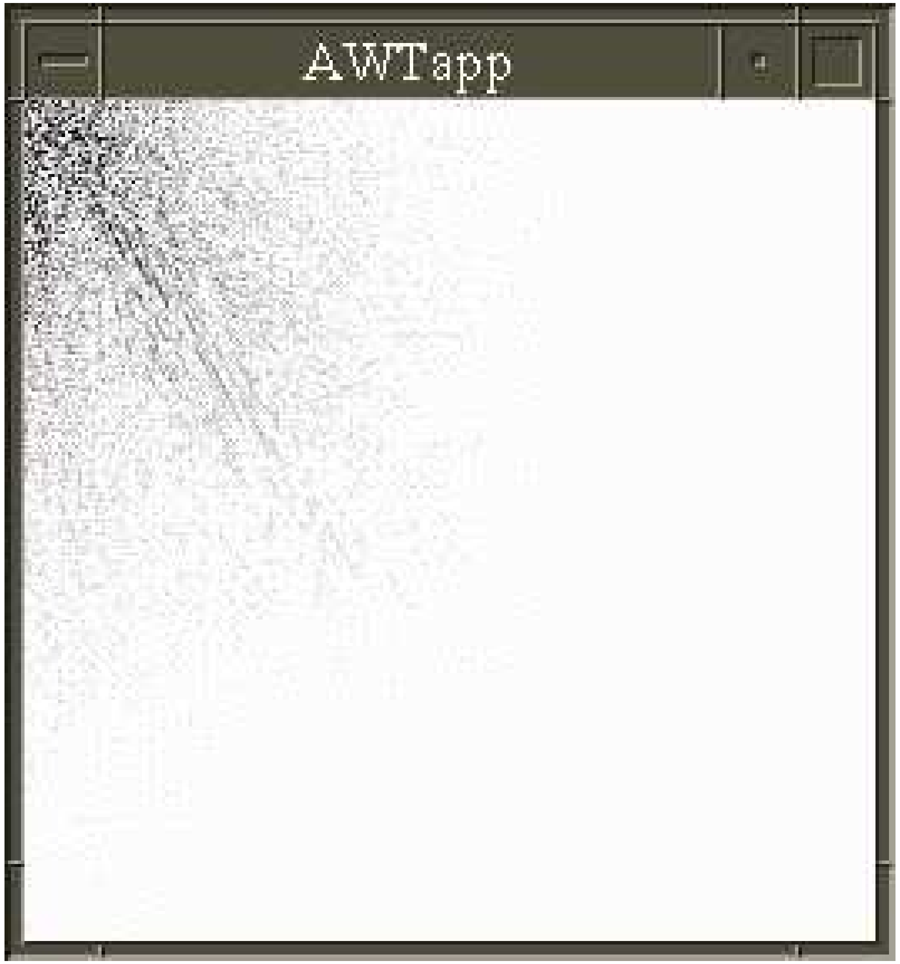


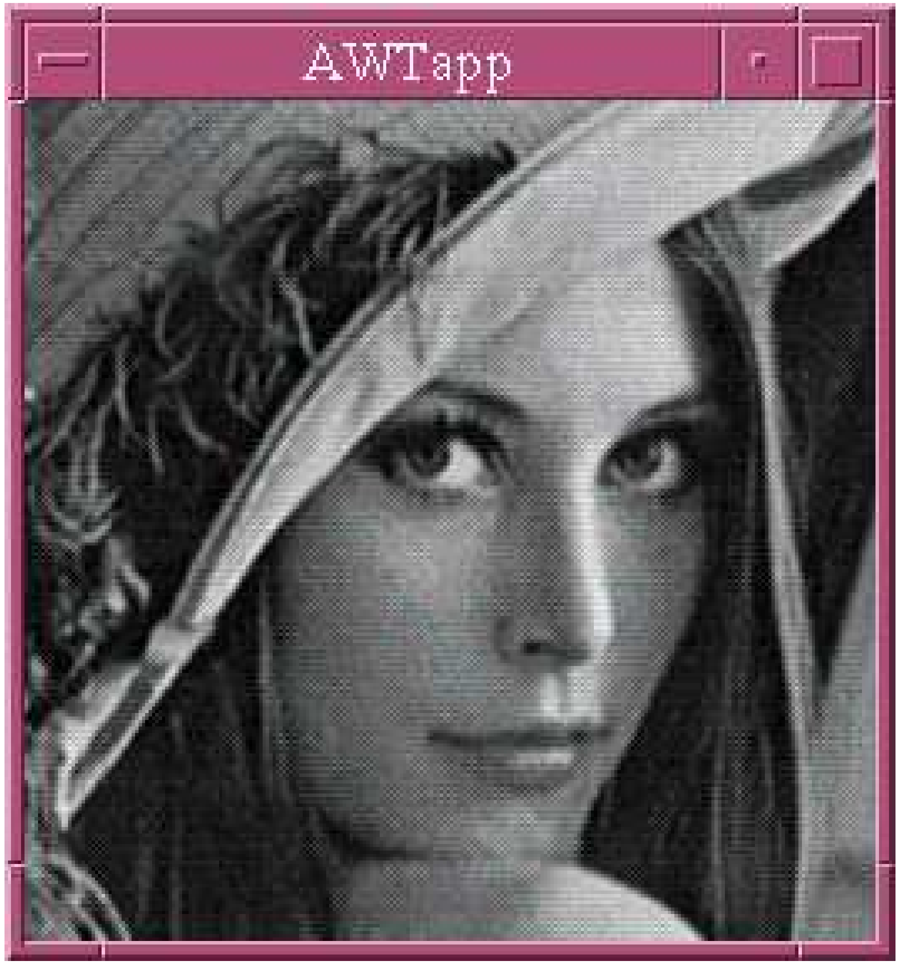
Cosinus transform quantized at level 100.0
Entropy = 0.2493 (zero order, per pixel)
Compression ratio : 32 (w.r.t. original image)



Decoded version of image

Cosinus transform of Lena





Cosinus transform quantized at level 20.0
Entropy = 1.4689 (zero order, per pixel)
Compression ratio : 5.4 (w.r.t. original image)

Decoded version of image

Operations which are easy on the transformed images

Filtering : e.g. HF noise vs LF signal.

Zooming, smoothing.

From the viewpoint of information theory

Concentration of entropy in a reduced number of pixels
$\Rightarrow$ image compression.

Data transmission in an appropriate order :

$\Rightarrow$ first send main information, then details

Sources of redundancy

(Remark : terminology used in image processing literature.)

1. Coding redundancy. Factor $2 - 3$

Some grey-levels are more frequent then others (cf. histogram)

2. Inter-pixel redundancy. Factor > 10

Nearby pixels are similar (continuity of the bi-dimensional signal)

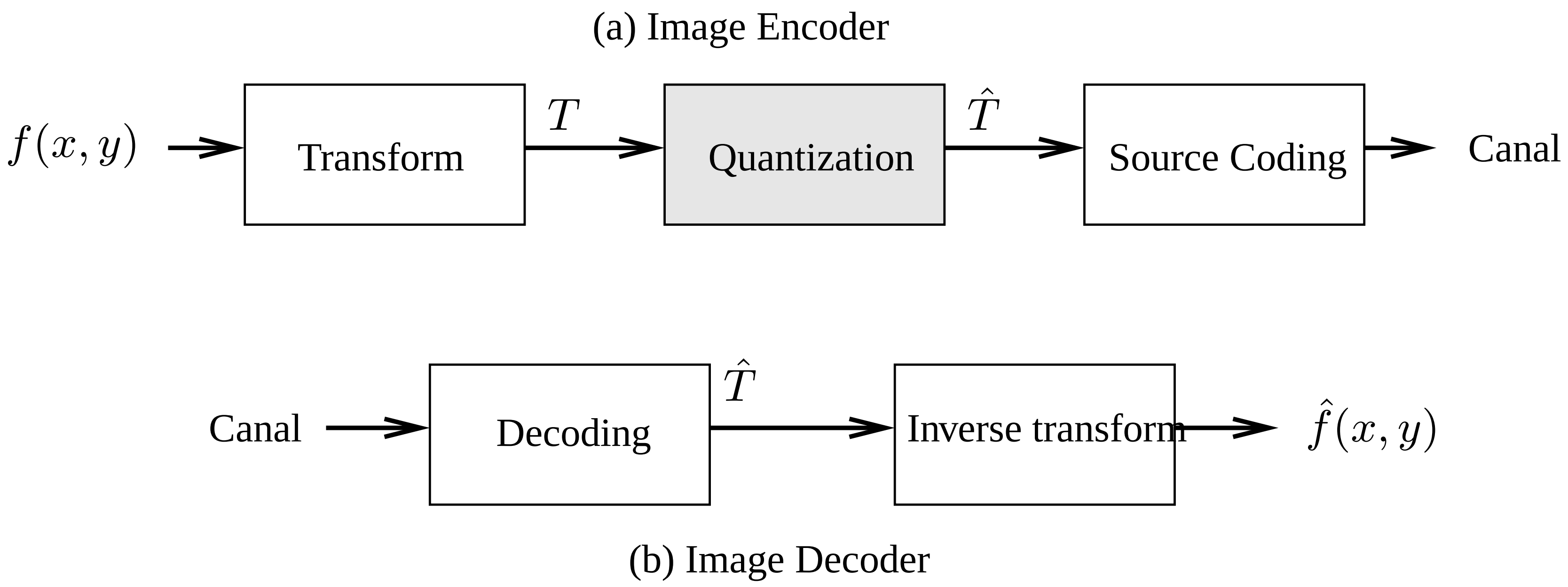
$\Rightarrow$ HF components are normally of low intensity.

3. Psycho-visual redundancy. Factor > 100

Our biological vision system is unable to detect all the details and is (hence) “robust” with respect to certain types of approximations.

$\Rightarrow$ allows to use irreversible compression techniques without impact on perception.

Image compression systems



NB: the central part of the encoder is not necessarily present.

First block: change representation to reduce inter-pixel redundancy and facilitate quantization (take advantage of phsycho-visual redundancy).

Last block: see data compression techniques.

Some approaches

A. Reversible

“Zero order”

In the binary case : coding of black and white areas (cf. FAX)

Differential coding : one transforms the image and codes the differences.

Bit planes.

Predictive coding :

One uses a predictive model to estimate the value of f_n given already seen pixels and one encodes only the prediction errors of this model.

NB: Differential coding = “naive” version of predictive coding.

One can use highly sophisticated prediction models (neural networks...) :
$\Rightarrow$ compromize between model complexity vs entropy of prediction errors
$\Rightarrow$ General principle in automatic learning (Minimum Description Length).

B. Irreversible

Predictive coding

We don't encode prediction errors (or very roughly)

Use of image transforms

Often applied locally.

One doesn't encode HF content (or very roughly).

C. Standards

Binary images : "run-length" encoding for FAX.

Monochrome images : JPEG (cosinus transform 8×8 plus Huffman.)

Sequences of color images : MPEG (cosinus transform, plus predictive models along time axis).

A short introduction to wavelets

Problem : basis functions of most classical transforms are not very good to represent images compactly.

Reasons : “non-stationary” aspect $\Rightarrow$ frequency content depends on spatial coordinates.

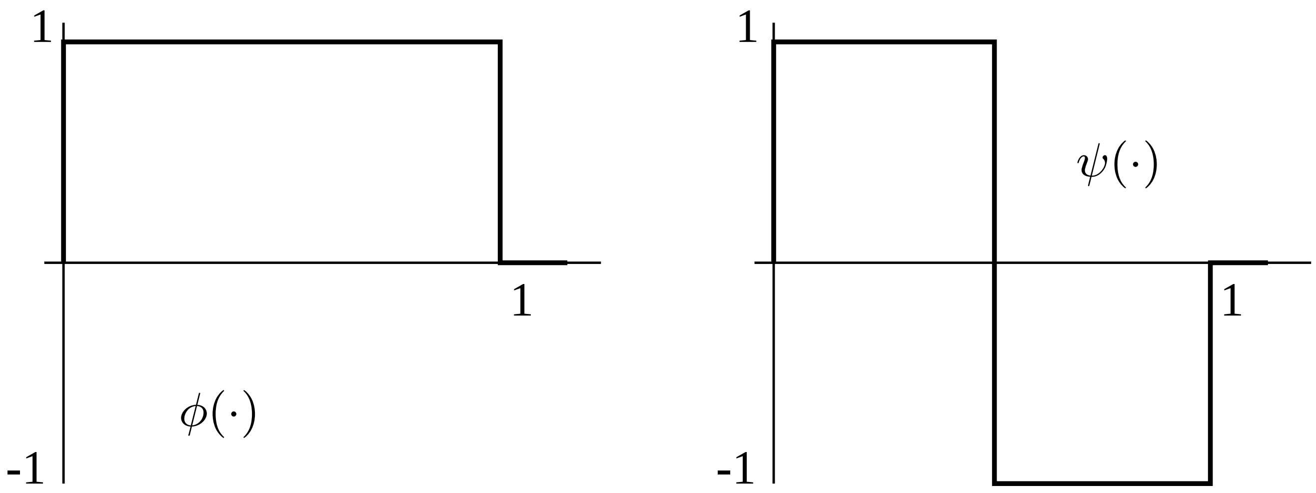
$\Rightarrow$ requires the use of image transforms on small windows of the original image (cf. JPEG).

Wavelets: constructive approach to build a catalog (dictionary) of well suited signals.

Main idea : extract frequency components *localized in space (or time)*

The higher the frequency, the more local the information extracted.

Example : Haar wavelets (local version of Walsh-Hadamard)



The function $\phi(\cdot)$ is called mother wavelet.

It is used to build all the other wavelets by translation/scaling.

E.g. : $\psi_0^1(x) = \psi(2x)$ et $\psi_1^1(x) = \psi(2x-1)$.

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}. \tag{10}$$

$\mathbb{R}^{2^{k+1}}$: $\psi_j^i = \psi(2^i x - j)$, avec $j = 0, 1, \ldots, 2^i - 1$

Compression technique (irreversible) :

(i) Compute Haar transform; (ii) set to 0 all pixels $\leq \epsilon$; (iii) code remaining prixels reversibly.

1

10

30

61