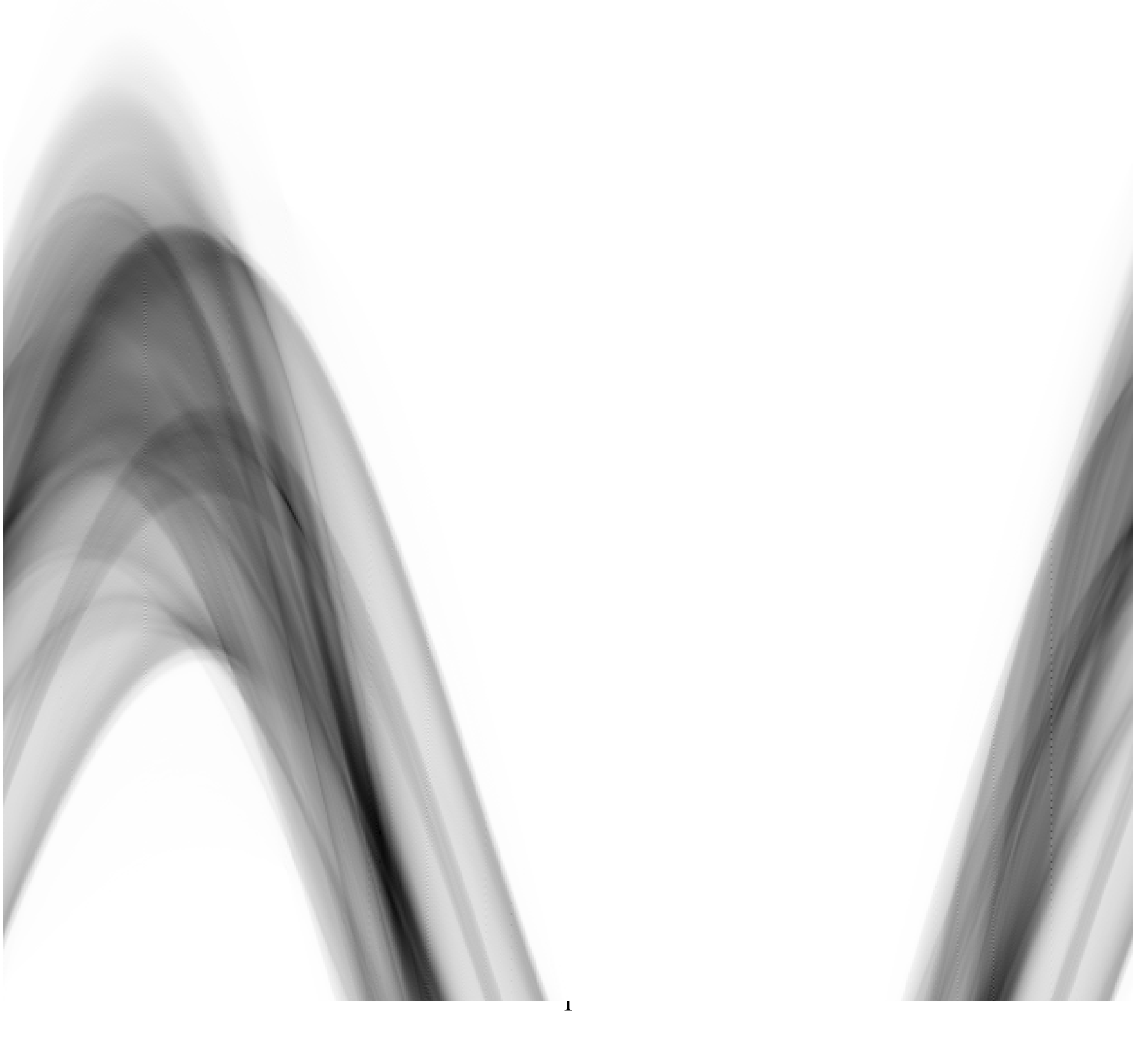
3A SRI

Vision and pattern recognition in images

philippe.joly@irit.fr



I. Metrology - Characterisation

A. Characterisation of regions

1. Colour

a) Histograms

See course 2A SRI

b) Colour spaces

See course 2A SRI

2. Texture

a) Fourier Transform - Gabor Filters

➔ **Direct and inverse Discrete Fourier Transform**

In the discrete domain, the calculation of the Discrete Fourier transform is usually expressed as:

$$\hat{F}_{\omega} = \sum_{t=0}^{N-1} F_t e^{-2i\pi\frac{\omega t}{N}}$$

The calculation of the inverse transform is then given by:

$$F_t = \frac{1}{N}\sum_{\omega=0}^{N-1} \hat{F}_{\omega} e^{2i\pi\frac{\omega t}{N}}$$

Example:

X	1	3	8	4	5	1	2	6
DFT(x)	30	-1.1716 - 6i	-4 + 6i	-6.8284 + 6i	2	-6.8284 – 6i	-4 - 6i	-1.1716 + i
|TFD(x)|	30	6.1133	7.2111	9.09	2	9.09	7.2111	6.1133

Note that:

- the first coefficient of the DFT is real. It is equal to the sum of the samples (in practice it is N.average(x)). This coefficient is generally referred to in frequency transforms as the "DC" or "continuous component" coefficient.

Exercise: Fill in the table below:

X	1	5	1	5	1	5	1	5
DFT(x)								
|TFD(x)|								

➔ 2D Fourier Transform

The 2D Fourier transform of an image is obtained by applying the DFT to each row independently of each other. Then the DFT is applied once again to each column of the result.

The resulting matrix is called the Fourier's spectrum.

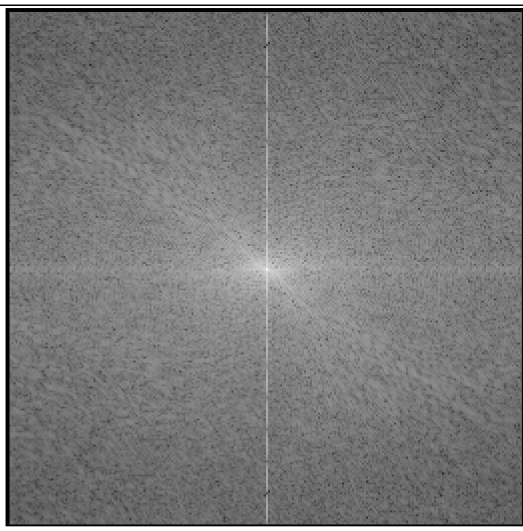
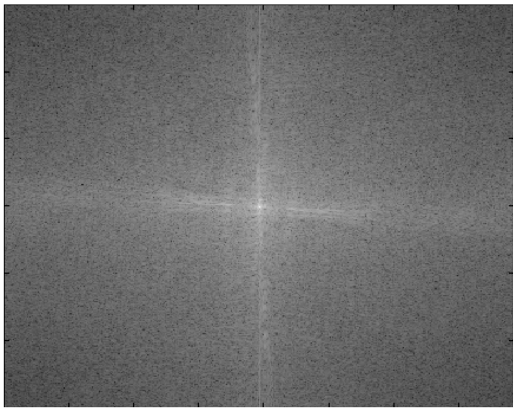
➔ Notions of Systems and Filters

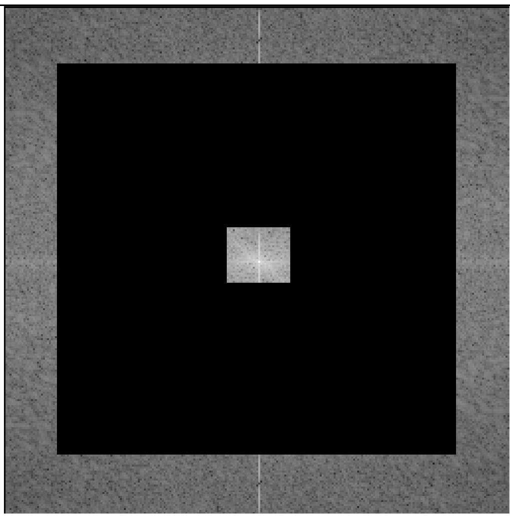
Once the spectrum has been calculated, it is possible to produce a frequency filter by cancelling or attenuating the amplitudes of some of the frequencies. A distinction is made between:

- high-pass/low-cut: retains only high frequencies
- low-pass / high-cut
- band-pass / band-cut: which retains (resp. cuts) only the information of a given frequency interval.

➔ Frequency image representation

It is often customary in image processing to swap the quadrants to bring the low frequencies (LF) to the centre in order to simplify filtering calculations.

 Starting image (Lena)	 DFT 2D (log of the standard module - LF in the middle)
	



Band-cut filter

Inverse DFT (filtered image)

➔ Gabor filters

The filter parameters are μ_θ, σ_θ, μ_ω, σ_ω

A filter is expressed as : $G(\theta,\omega) = e^{\frac{(\theta-\mu_\theta)^2}{2.\sigma_\theta^2}} . e^{\frac{(\omega-\mu_\omega)^2}{2.\sigma_\omega^2}}$

Example of possible values for the filter parameters:

for i ranging from 0 to 4 (= 5 possible "grains")

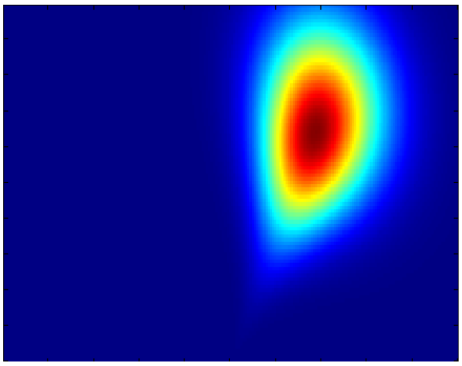
for j ranging from 0 to 5 (= 6 possible orientations)

μ_ω(i) = 3/4*(max(F) - min(F)).2^{-i}

μ_θ(j) = j* π /6

σ_ω(i) = (max(F) - min(F)).$2^{-(i+1)}$/sqrt(8.ln(2))

σ_θ(j) = (π/12)/sqrt(2.ln2)



b) Wavelet Transform - Haar Filters

➔ Continuous wavelet transform

$W(a,b) = < f, \psi_{a,b} > = \int_{t=-\infty}^{+\infty} f(t).\overline{\psi}_{a,b}(t)\partial t$ with $\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right)$.

Where :

- $\psi(t)$ is a **mother wavelet** to be defined.

- The parameter "a" expresses the "dilation" (or scale) of the mother wavelet. The smaller the (temporal) scale, the higher the frequency: a small wavelet is used, which therefore oscillates more. Conversely, a large scale will allow low frequencies to be analyzed (the wave oscillates more slowly).
- "b" expresses its translation, i.e., the position on the signal where the wavelet decomposition is calculated.

For a function to act as a mother wavelet, it must be continuous, complex-valued (or real), and it must satisfy the following conditions:

- $\int_{t=-\infty}^{+\infty} \psi(t)\partial t = 0$ (i.e., in the discrete domain, if it is real: $\sum \text{coeff} > 0 = -\sum \text{coeff} < 0$)
- $\int_{t=-\infty}^{+\infty} |\psi(t)|^2 \partial t < \infty$

The reverse transformation is obtained with :

$$f(t) = \frac{1}{c_\psi^2} \int_a \int_b W(a,b) \frac{1}{a^2} \psi\left(\frac{t-b}{a}\right) \partial b . \partial a \quad \text{with} \quad c_\psi = \sqrt{2\pi \int_{\omega=-\infty}^{+\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} \partial \omega}$$

This reconstruction is only possible if $c_\psi < +\infty$. This condition is called the **admissibility condition**.

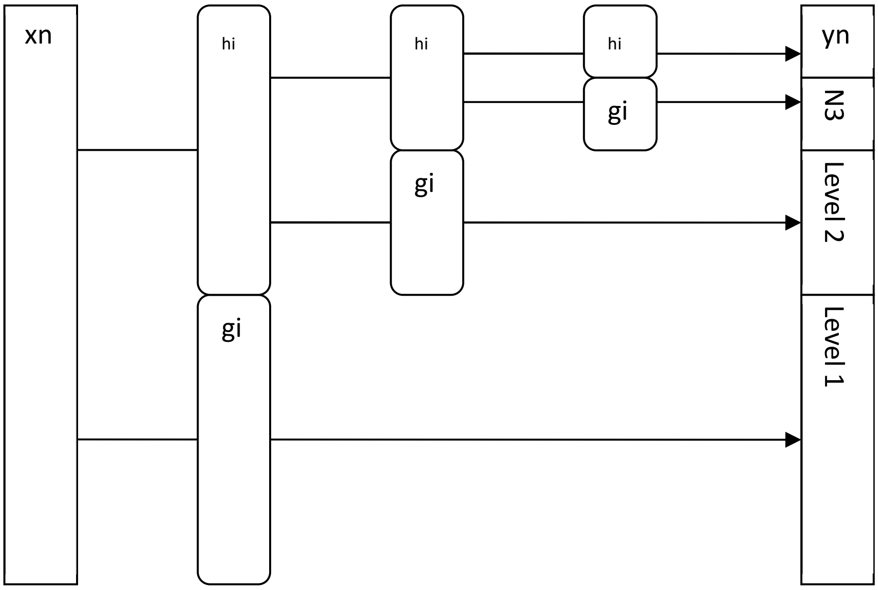
➔ Discrete Wavelet Transform - Mallat's Algorithm

The first problem is to discretize the parameter space of a and b. In order to guarantee the reconstruction of the signal by inverse transform, a "logarithmic discretization" of the scale factor is used - the base of the logarithm being left to the user's choice. Generally, a binary logarithm is used, which leads to the values 1, 2, 4, 8, 16, ... for a. This is known as a **dyadic decomposition of** the signal.

We pose $a = a_0^j$ where a0 depends on the chosen logarithmic base (ie. 2) and where j expresses the level of the decomposition. $b = k.a_0^j.b_0$ Where k expresses the multiple of the window shift, and where b0 is relative to the window's progression step. In general, b0 = 1 is chosen.

We can then express the **scaling function** that represents the wavelet at different scale values and for different translation values: $\psi_{j,k}(t) = a_0^{\frac{-j}{2}} . \psi\left(a_0^{-j} t - k b_0\right)$.

The Discrete Wavelet Transform is expressed as an algorithm that breaks down the signal into different frequency sub-bands. These bands are obtained by convolving this signal with a low-pass linear filter and a high-pass linear filter (corresponding to a discrete wavelet $\psi_{j,k}$) at different scales.



The impulse responses of h and g are dependent on each other (filtering must be complementary). The relationship between h and g can be:

$$g_{L-1-n} = (-1)^n.h_n$$

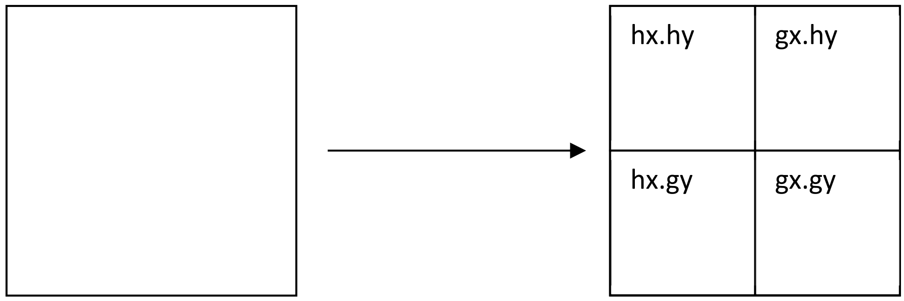
Filters that verify this relationship are called **quadrature mirror filters**.

The number of levels (i.e. the number of iterations of the low-frequency process - or the "order") can be fixed a priori, or it can depend on a threshold on the number of values produced after subsampling.

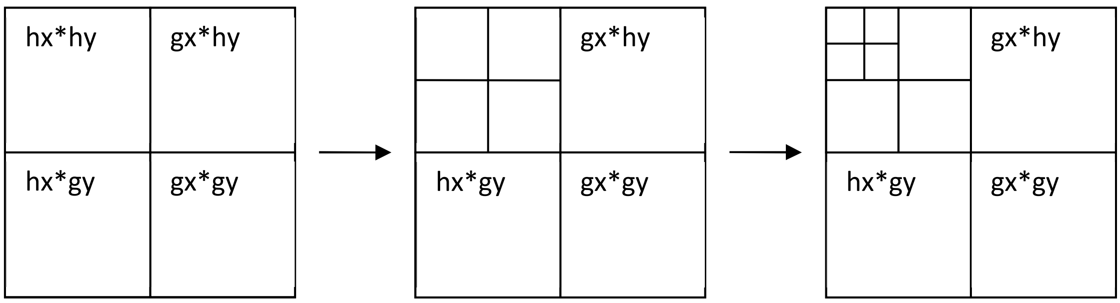
Each level of the decomposition corresponds to a resolution of the signal (low frequency) which is sub-sampled. This is called a **multi-resolution analysis**.

➔ 2D wavelet transform

The image is convolved row by row with the gx and hx filters, and then, on the results of this first convolution, a second convolution is performed column by column with the gy and hy filters. The results are arranged in a matrix with the same dimensions as the image in the following way:



The process is repeated on the low frequency image obtained in part (hx.hy)



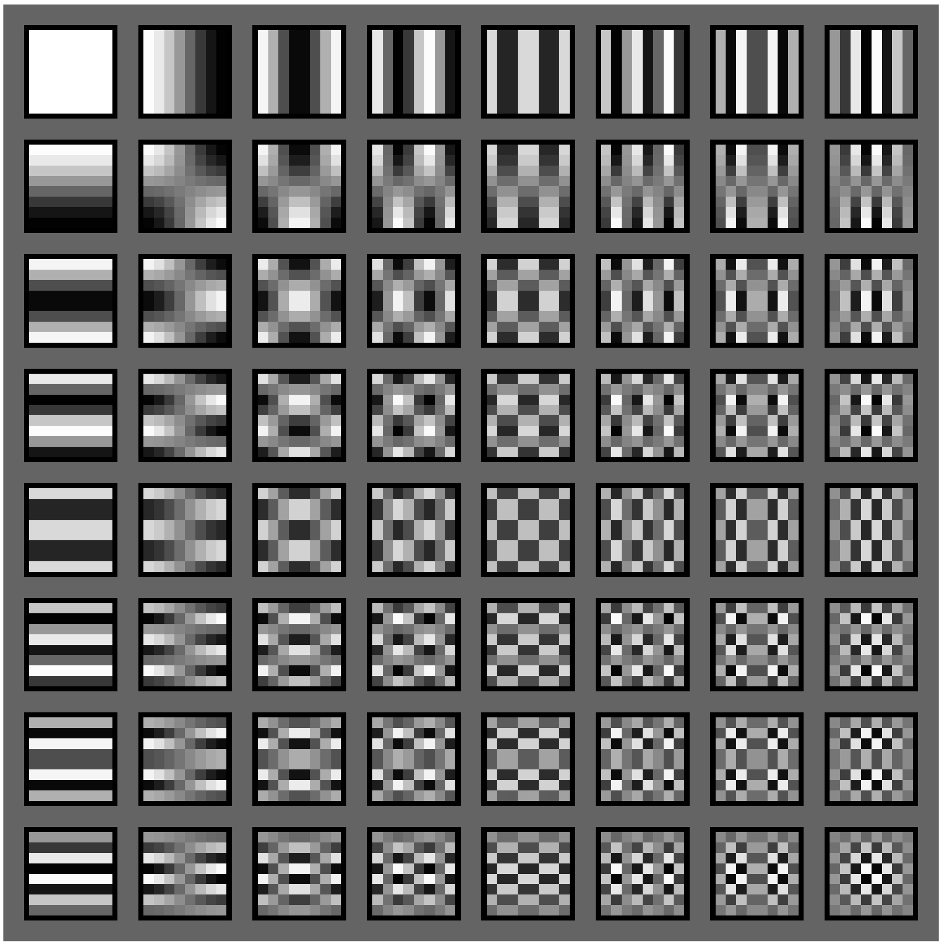
c) Discrete Cosine Transform

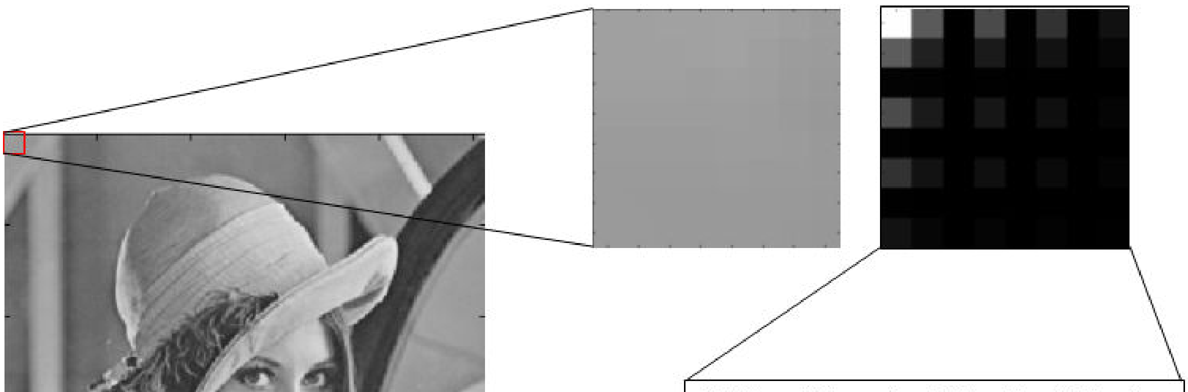
$$DCT(u,v)=\frac{2}{N}c(u).c(v)\sum_{y=0}^{N-1}\sum_{x=0}^{N-1}\cos\left(\frac{\pi}{N}u\left(x+\frac{1}{2}\right)\right)\cos\left(\frac{\pi}{N}v\left(y+\frac{1}{2}\right)\right)M(x,y)$$

with $c(a)=\begin{cases}\frac{1}{\sqrt{2}}\ \text{si a}=0\\ 1\ \text{sinon}\end{cases}$

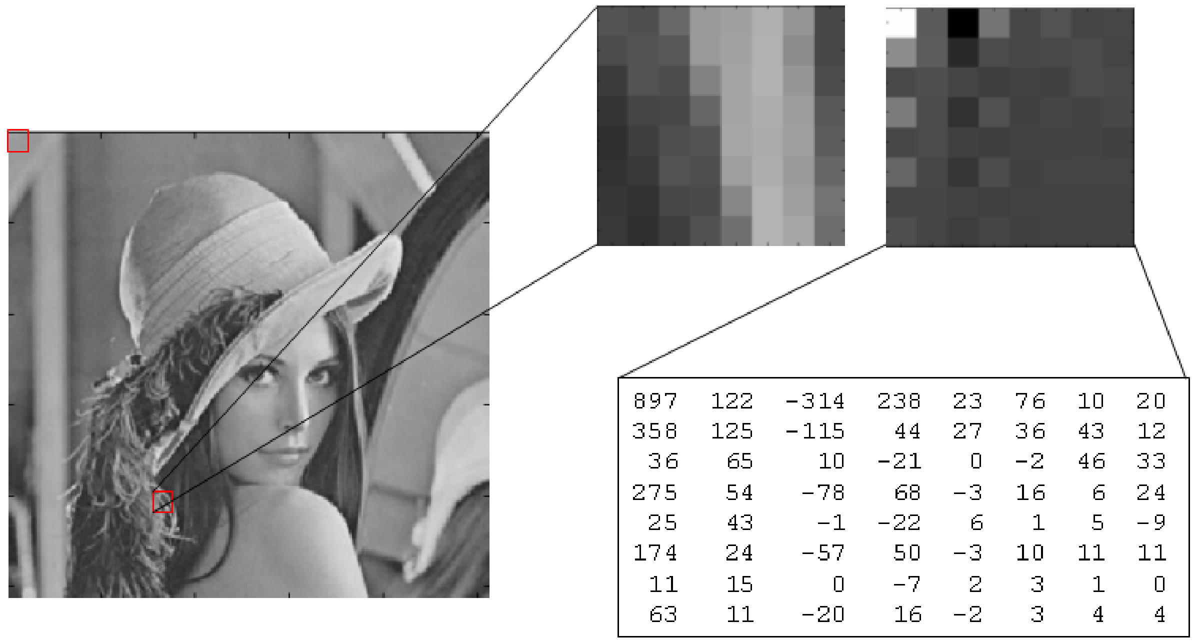
The inverse transform is obtained with :

$$M(x,y) = \frac{2}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} c(u).c(v).DCT(u,v)\cos\left(\frac{\pi}{N}u\left(x+\frac{1}{2}\right)\right)\cos\left(\frac{\pi}{N}v\left(y+\frac{1}{2}\right)\right)$$





1258	444	0	367	5	245	2
454	162	-1	130	3	87	0
13	7	0	2	1	2	0
372	130	0	108	0	72	0
8	0	-2	1	0	1	0
248	87	0	72	1	48	0
3	1	0	0	0	0	0
86	30	0	25	0	17	0



897	122	-314	238	23	76	10	20
358	125	-115	44	27	36	43	12
36	65	10	-21	0	-2	46	33
275	54	-78	68	-3	16	6	24
25	43	-1	-22	6	1	5	-9
174	24	-57	50	-3	10	11	11
11	15	0	-7	2	3	1	0
63	11	-20	16	-2	3	4	4

d) Co-occurrence matrices

The co-occurrence matrix NxN (N = nb of grey levels) is defined by :

$$m_R(i,j) = \frac{\#\{(x,y)(x',y') \mid (x,y)R(x',y'),\ I(x,y)=i, I(x',y')=j\}}{\#\{(x,y)(x',y') \mid (x,y)R(x',y')\}}$$

R: spatial relationship between 2 pixels (distance and orientation)

I(x,y): grey level at position (x,y)

#: number of elements

Haralick parameters are often used to characterise the distribution of coefficients in the matrix. They are calculated using the following 14 functions:

notations

$m_x(i)$: somme de la ligne i $\qquad m_{x+y}(k)$: somme de la k ième diagonale secondaire

$m_y(i)$: somme de la colonne i $\qquad m_{x-y}(k)$: somme de la k ième diagonale principale

moment angulaire d'ordre 2 $\quad f_1 = \sum_i \sum_j m(i,j)^2$

contraste $\quad f_2 = \sum_{n=0}^{N} n^2 \cdot m_{x-y}(n)$

corrélation $\quad f_3 = \dfrac{\sum_i \sum_i \left(i \cdot j \cdot m(i,j) - \mu_x \mu_y\right)}{\sigma_x \sigma_y}$

variance $\quad f_4 = \sum_i \sum_i (i-j)^2 \cdot m(i,j)$

moment des différences inverses $\quad f_5 = \sum_i \sum_i \dfrac{1}{1+(i-j)^2} \cdot m(i,j)$

moyenne des sommes $\quad f_6 = \sum_{k=2}^{2N-1} k \cdot m_{x+y}(k)$

variance des sommes $\quad f_7 = \sum_{k=2}^{2N-1} (k - f_6)^2 \cdot m_{x+y}(k)$

entropie de la somme $\quad f_8 = -\sum_{k=2}^{2N-1} m_{x+y}(k) \cdot \log\left(m_{x+y}(k)\right)$

entropie $\quad f_9 = -\sum_i \sum_j m(i,j) \cdot \log\left(m(i,j)\right)$

variance des différences $\quad f_{10} = \sum_{k=0}^{N} \left(k - \mu_{x-y}\right)^2 \cdot m_{x-y}(k)$

avec $\quad \mu_{x-y} = \dfrac{1}{N} \sum_{k=0}^{N} m_{x-y}(k)$

entropie des différences : $$f_{11} = -\sum_{i=0}^{N} m_{x-y}(i)\log\left(m_{x-y}(i)\right)$$

corrélation de l'information : $$f_{12} = \frac{H_{XY} - H1_{XY}}{\max\left(H_X, H_Y\right)}$$

$$f_{13} = \left[1 - \exp\left(-2.0\left(H2_{XY} - H_{XY}\right)\right)\right]^{\frac{1}{2}}$$

avec $$H_{XY} = f_9 \qquad H_X = -\sum_{k=0}^{N} m_x(k)\log\left(m_x(k)\right) \qquad H_Y = -\sum_{k=0}^{N} m_y(k)\log\left(m_y(k)\right)$$

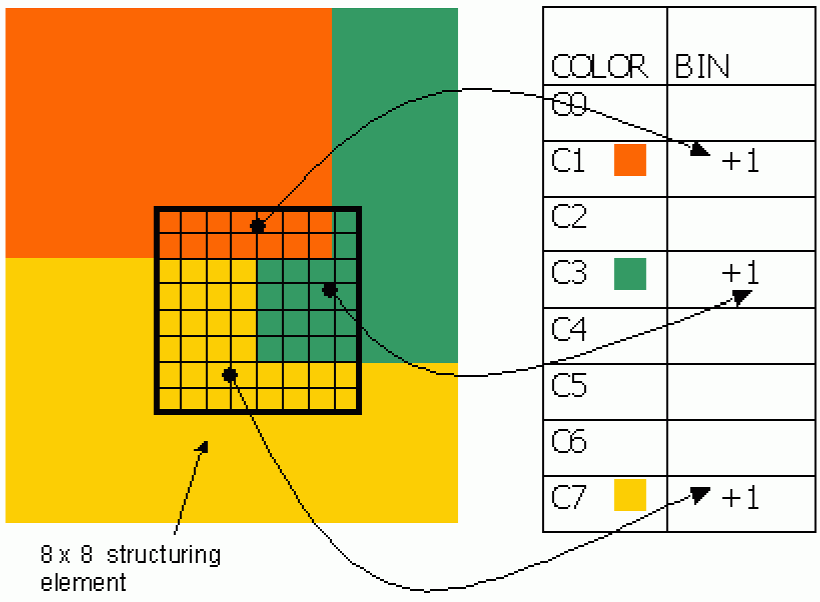
$$H1_{XY} = -\sum_i \sum_j m(i,j)\log\left(m_x(i)m_y(j)\right) \qquad H2_{XY} = -\sum_i \sum_j m_x(i)m_y(j)\log\left(m_x(i)m_y(j)\right)$$

corrélation maximum : $$f_{14} = \left(2\grave{e}me \text{ des plus grandes valeurs propres de } Q\right)^{\frac{1}{2}}$$

$$Q = \left\{Q(i,j)\right\}\begin{cases} i = 0 \cdots N \\ j = 0 \cdots N \end{cases} \qquad Q(i,j) = \sum_{k=0}^{N} \frac{m(i,k)m(k,j)}{m_x(i)m_y(j)}$$

e) Colour structure

An array is created with as many entries as there are colours in the LUT.
A structuring element (here an 8x8 square) scans the region.
For each different colour C in the region, T[C] <- T[C] + 1



When a colour is "compact", it is associated with a small value in the array relative to the number of pixels involved. When a colour is distributed, it has a large value in the array relative to the number of pixels involved.