

Année 2020-2021



Traitement numérique des images

Etienne BELIN
etienne.belin@univ-angers.fr
Université d'Angers

Chapitre 0 : la vision artificielle (computer vision)

- **Qu'est ce que c'est ?**

Connaissances et technologies qui permettent de concevoir des machines qui peuvent « voir »

- **Premier niveau : vision**

Acquiert une image grâce à une chaîne intégrant des éléments optiques et un capteur

- **Second niveau : traitement de l'image**

Modifie le contenu de l'image afin de mettre en évidence des éléments d'intérêt (objets, contours)

- **Troisième niveau : reconnaissance**

Utilise des techniques « d'intelligence artificielle » pour identifier des formes connues dans l'image

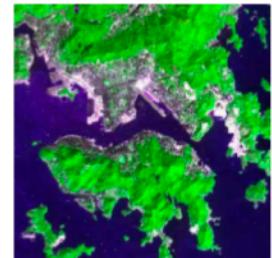
Extension de la perception humaine ?

- **Capteurs meilleurs que l'oeil humain**

Voient ce que nous ne voyons pas directement, du fait de la limitation de notre système visuel

- **Autres propriétés optiques**

- problème d'échelle, de résolution, de point de vue (ex : images satellitaires)



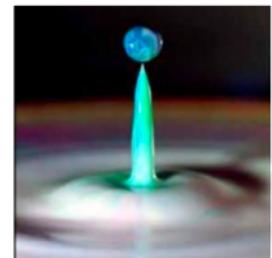
- **Autres gammes de longueurs d'onde**

- caméras multispectrales, infrarouge proche, thermiques (infrarouge lointain)



- **Autres cadences**

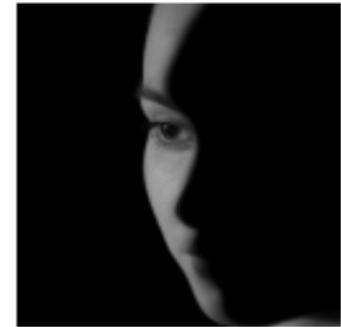
- caméras à haute cadence, plusieurs milliers d'images par seconde.



Mieux que la perception humaine ?

- **Moins « intelligent » que l'humain**

Même si les images initiales sont parfois très riches, le résultat de leur traitement par un système de VA reste souvent basique



- **Puissance de calcul**

Les systèmes de VA ont une capacité de calcul infiniment plus limitée que celle du cerveau humain, ou de celui des animaux évolués



- **Connaissance et reconnaissance**

Nous exploitons nos connaissances pour interpréter le contenu de l'image : on reconnaît parfois des objets alors qu'ils ne sont pas visibles



Exemple d'applications : imagerie médicale

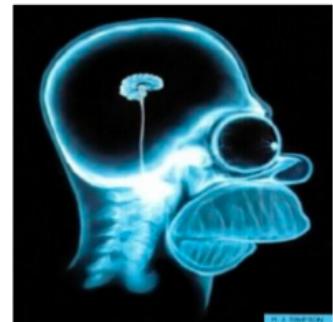
- **Objectifs**

aider le médecin lors du diagnostic, le chirurgien lors de la réalisation d'un geste opératoire



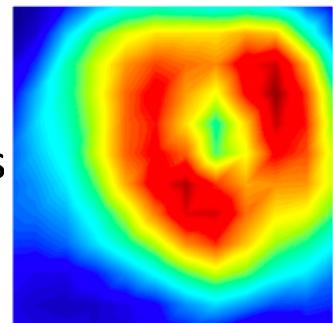
- **Amélioration des images**

Rehaussement du contraste, élimination du bruit, mise en évidence des détails



- **Détection et localisation**

Positionnement des organes, détection des tumeurs, mesure de dimensions et de volumes



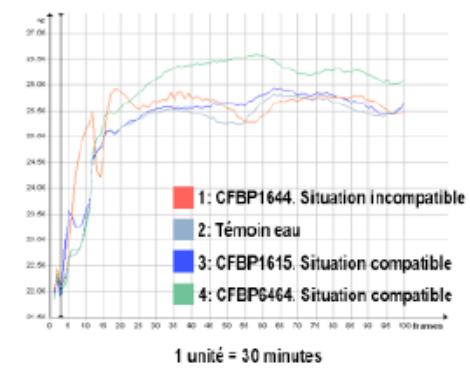
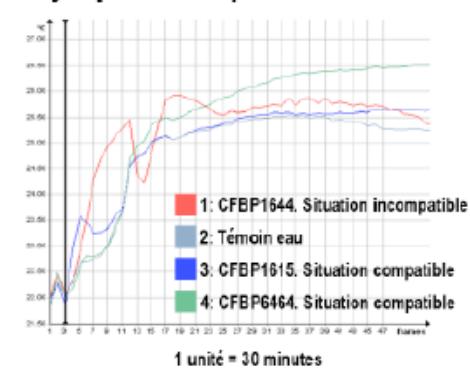
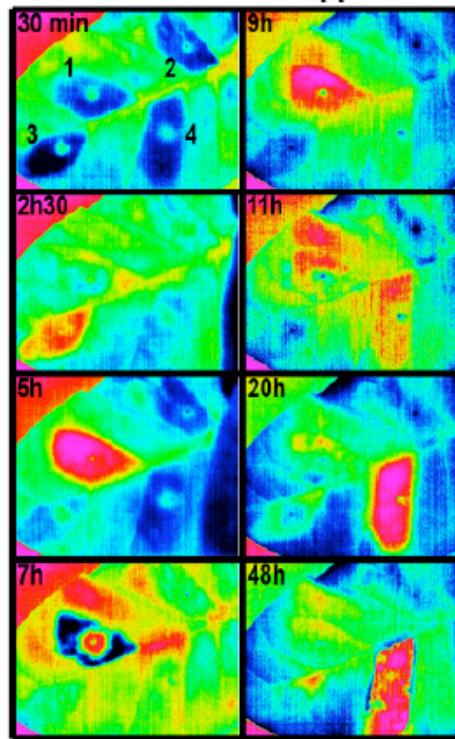
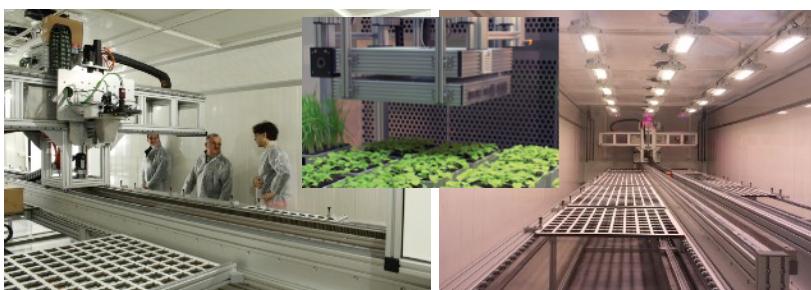
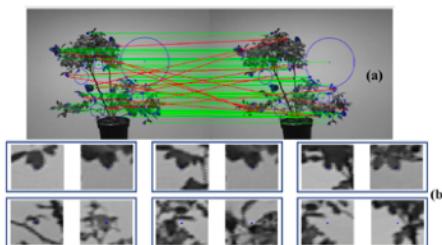
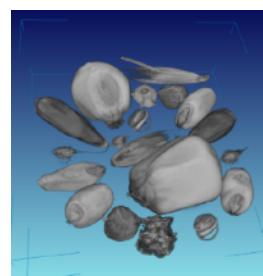
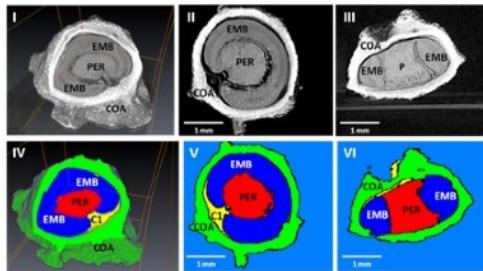
- **Imagerie interventionnelle**

Assistance en ligne au praticien : opérations réalisées sur les images en temps réel

Exemple d'applications : imagerie pour la caractérisation du végétal

• Objectifs

collecter, sur un grand nombre d'individus, un ensemble de caractéristiques notamment morphologiques dont la variabilité est analysée en fonction du patrimoine génétique et des conditions environnementales de production.



Exemple d'applications : industriel pour contrôle de la qualité, de l'aspect, ...

- **Objectifs**

Eviter le contrôle visuel par un opérateur (tâche répétitive peu valorisante)



- **Contrôle dimensionnel**

Détermine la dimension, la forme, la position de l'objet qu'il observe



- **Contrôle d'aspect**

Détermine la couleur, la texture des objets observés



- **Contrôle de la qualité**

À partir des données précédentes, le système détermine la qualité d'un produit

Qu'est ce qu'une image ?

Une image est avant tout un **signal 2D (x,y)**. Souvent, cette image **représente une réalité 3D (x,y,z)**

- D'un point de vue mathématique :

Une image est un matrice de nombres représentant un signal

Plusieurs outils permettent de manipuler ce signal



- D'un point de vue humain :

Une image contient plusieurs informations sémantiques

Il faut interpréter le contenu au-delà de la valeur des nombres

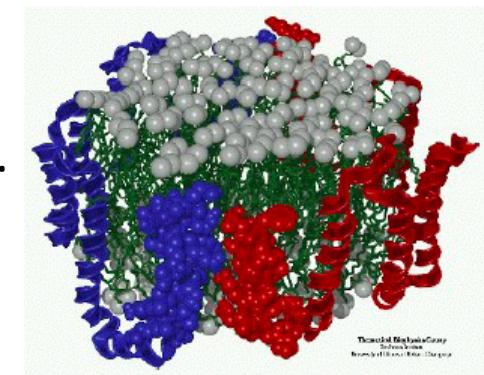
- Image naturelle – Plusieurs moyens d'acquisition

caméra, microscope, tomographie, infra-rouge, satellite, ...

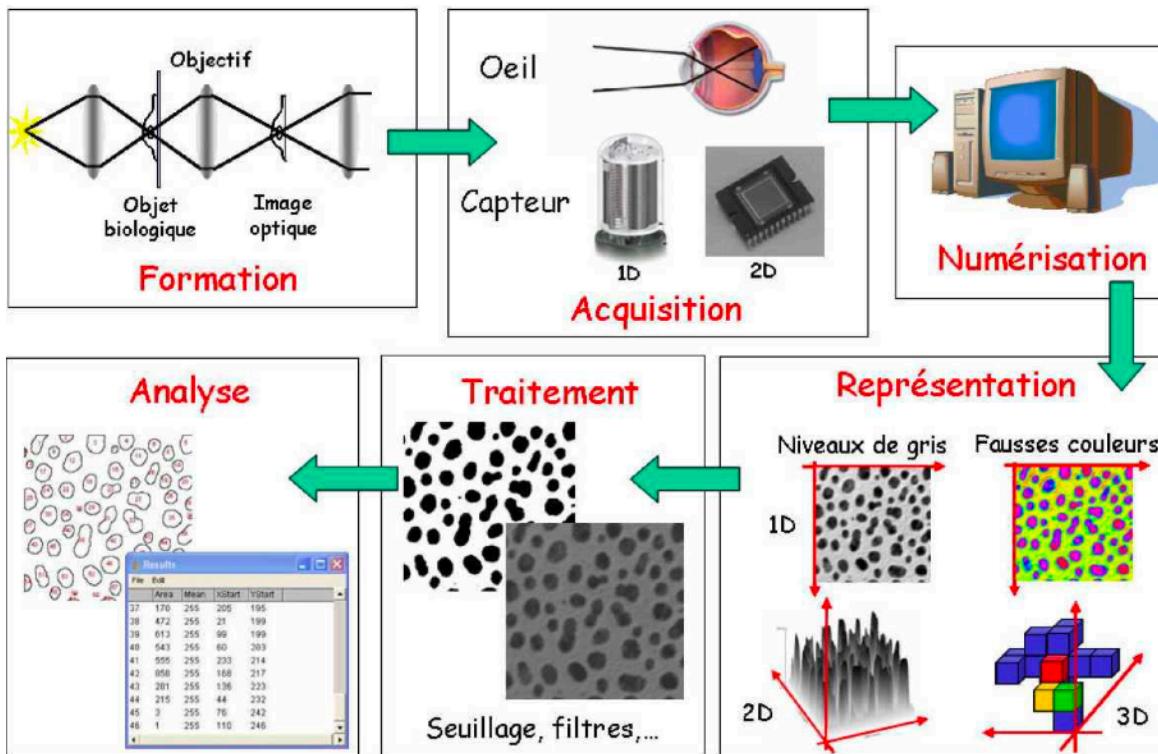
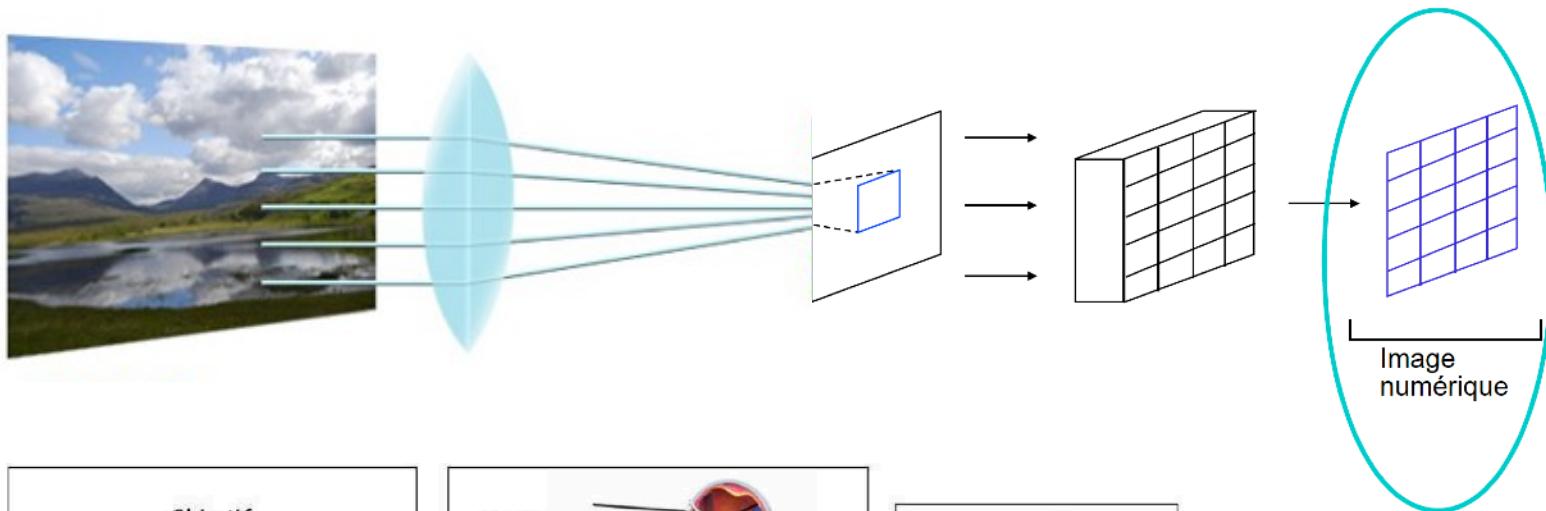


- Image artificielle – Plusieurs outils de représentation

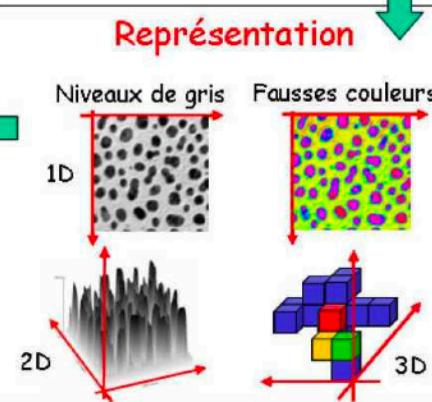
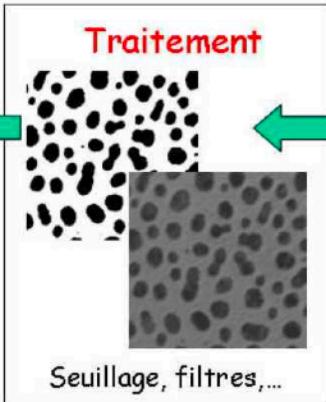
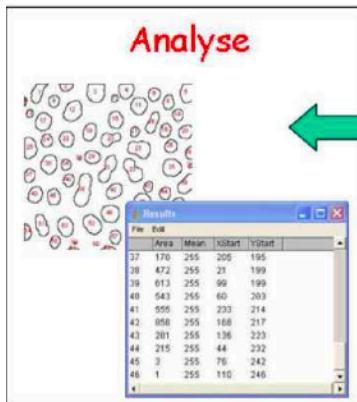
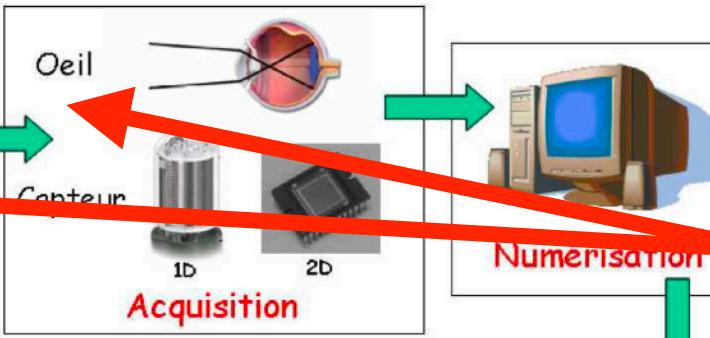
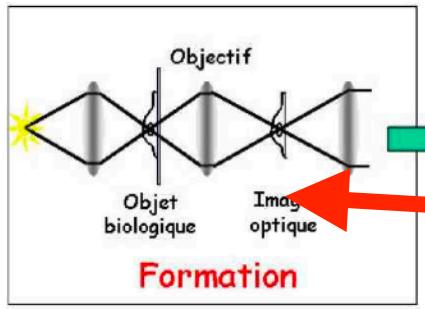
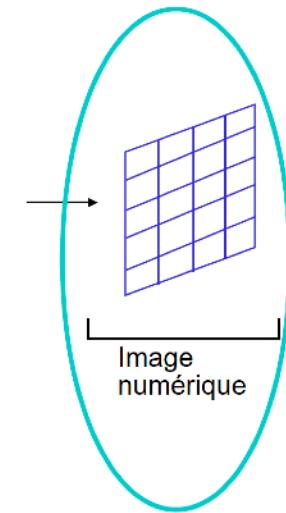
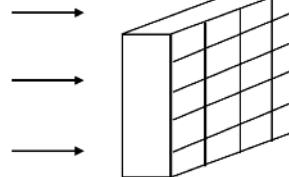
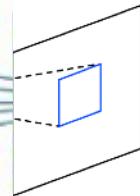
synthèse d'images, réalité virtuelle, visualisation scientifique, ...



Comment se forme une image (naturelle) sur un capteur ?



Comment se forme une image (naturelle) sur un capteur ?



Faire des images “exploitables” commence en contrôlant les 2 premières étapes.

Controlling illumination & sensor

Objective: extract as easy as possible a useful information



Information : contrast between an object (useful information) and the background (noise)



Contrast = a good contrast comes from illumination

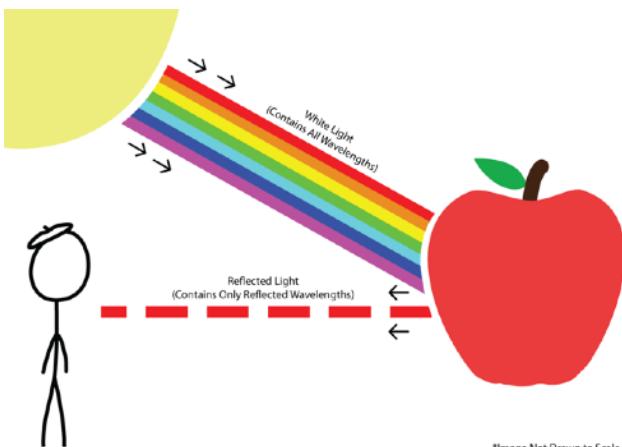
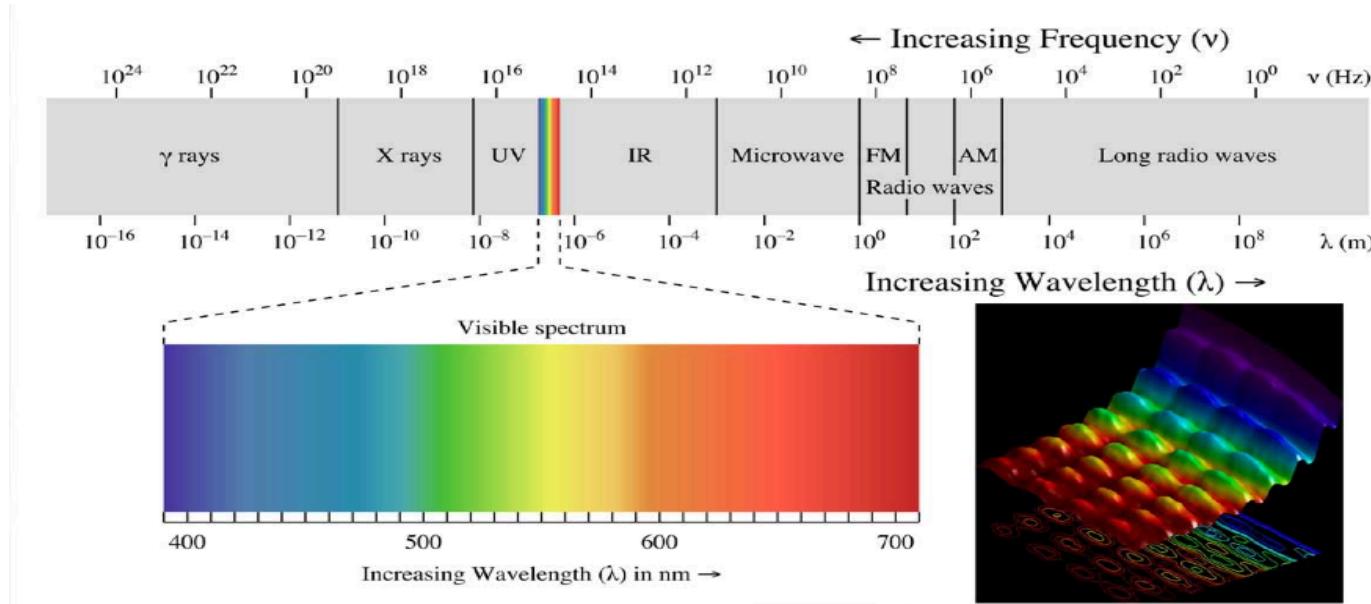


Illumination= light

We use light properties and interactions between light and matter
(directivity, wavelength, intensity, temporal coherence, polarization, absorbance of objects, ...)

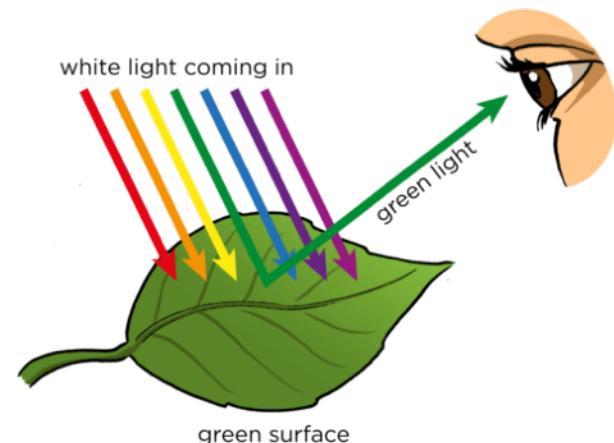
If the scene has an absorbance contrast

- From white light (visible spectrum) to reflected color by absorbance



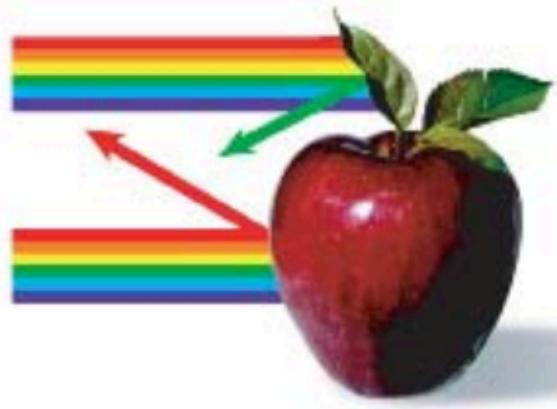
Color object defined
by the wavelength
(color) reflected

*Image Not Drawn to Scale (Obviously)

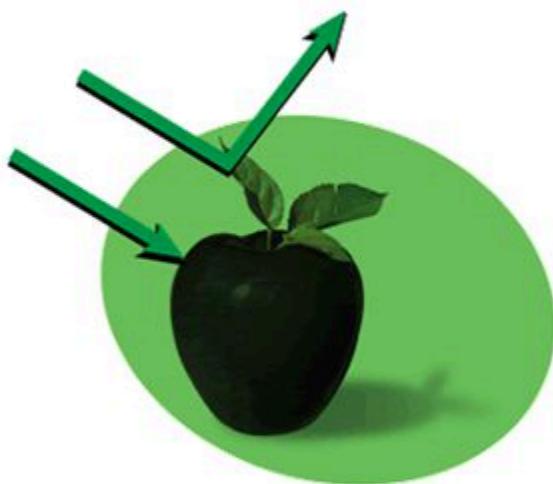


If the scene has an absorbance contrast

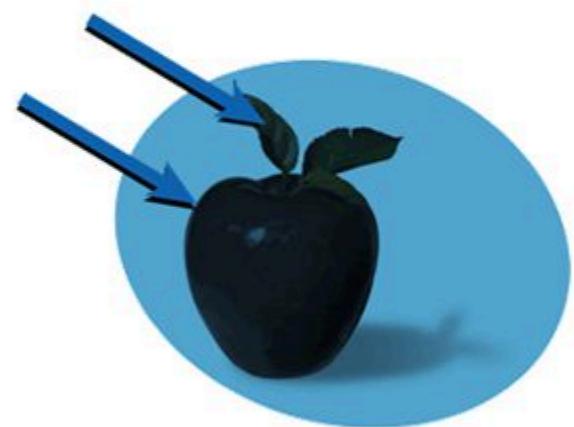
- From specific light



In red light, the apple appears red because it reflects the red light. But the leaves look black.



In green light, the apple appears black because no red light strikes it. But the leaves look green.



In blue light, both the apple and the leaves appear black.

If the scene has an absorbance contrast

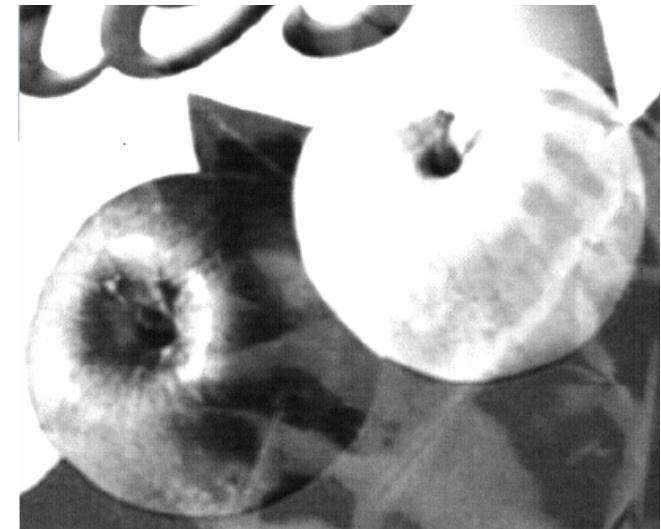


White light

Red light



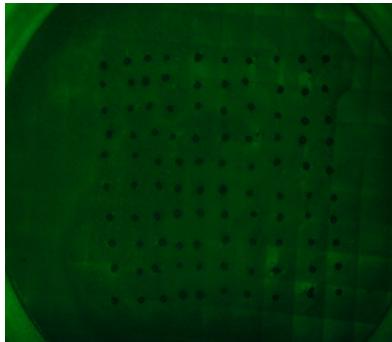
Weak contrast



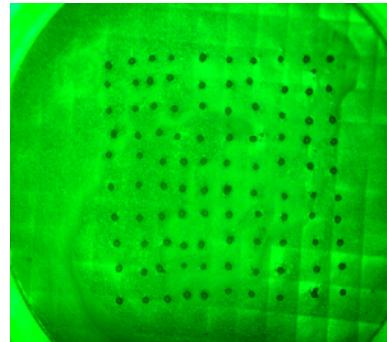
Strong contrast

If the scene has an intensity contrast

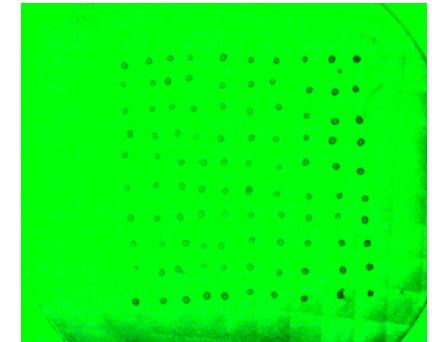
Weak intensity



Correct intensity



Strong intensity

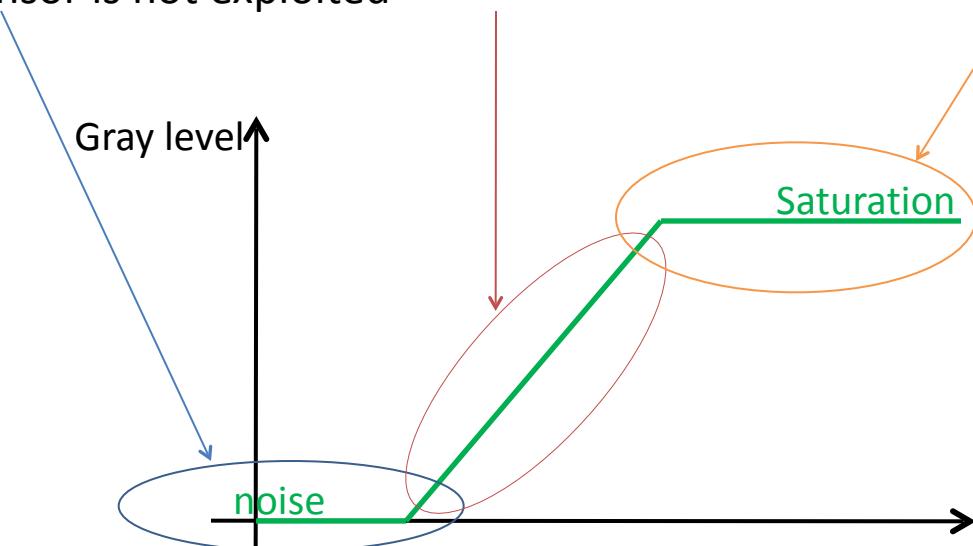


Bad contrast

Dynamic of the sensor is not exploited

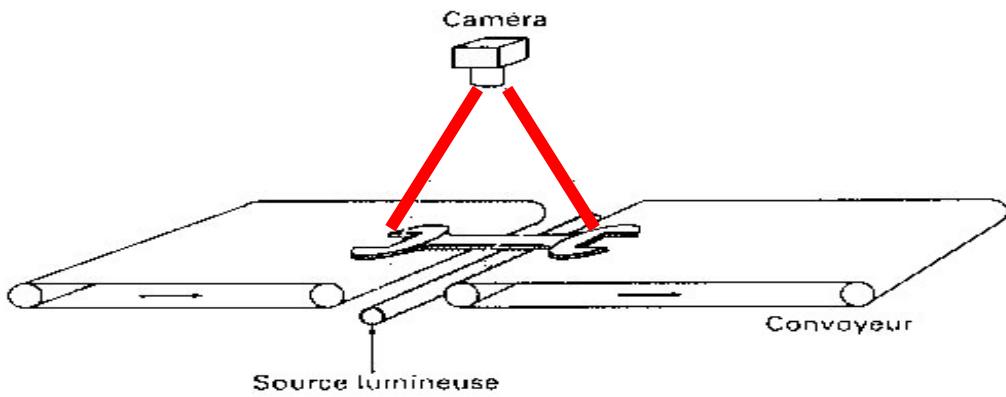
Good contraste

Very good contrast
even if dynamic is badly
exploited

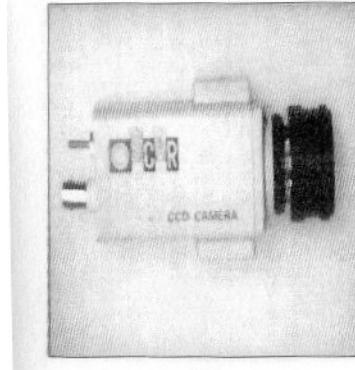


If contrast between object/background

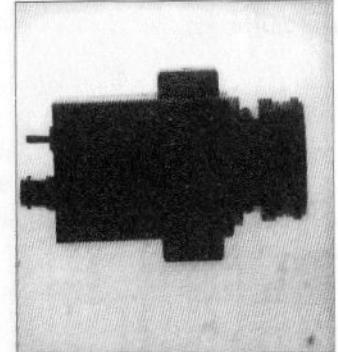
Opaque objects = edge can be obtained with **backlighting**



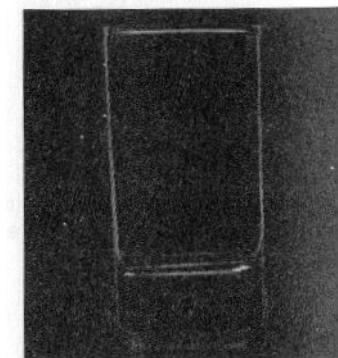
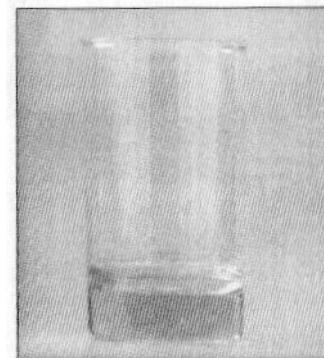
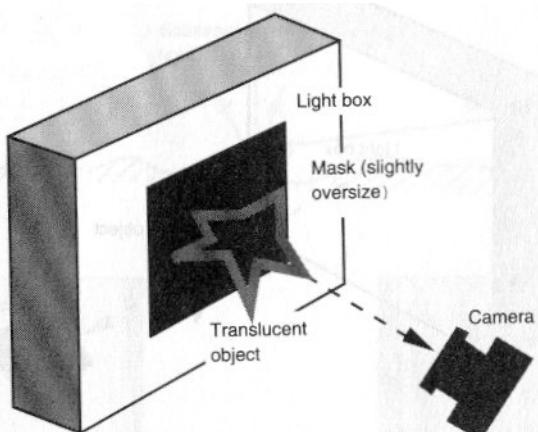
Scatter lighting from front



backlight

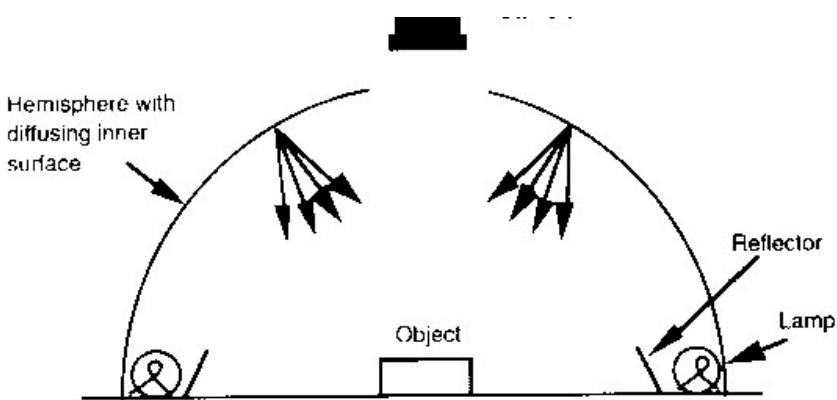


Transparent objects



If contrast due to directivity

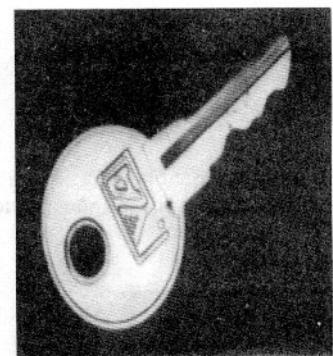
- Scatter (diffuse)



Scatter light



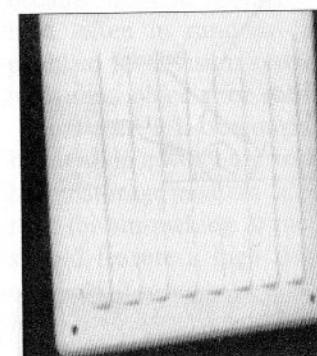
Specular light



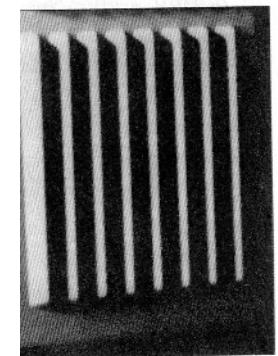
- Specular (direct) : necessity to control the orientation of observed object

- Low : require a powerful light source

Scatter light



Low light



What's your opinion ?

To see craters



To see clear and dark spots

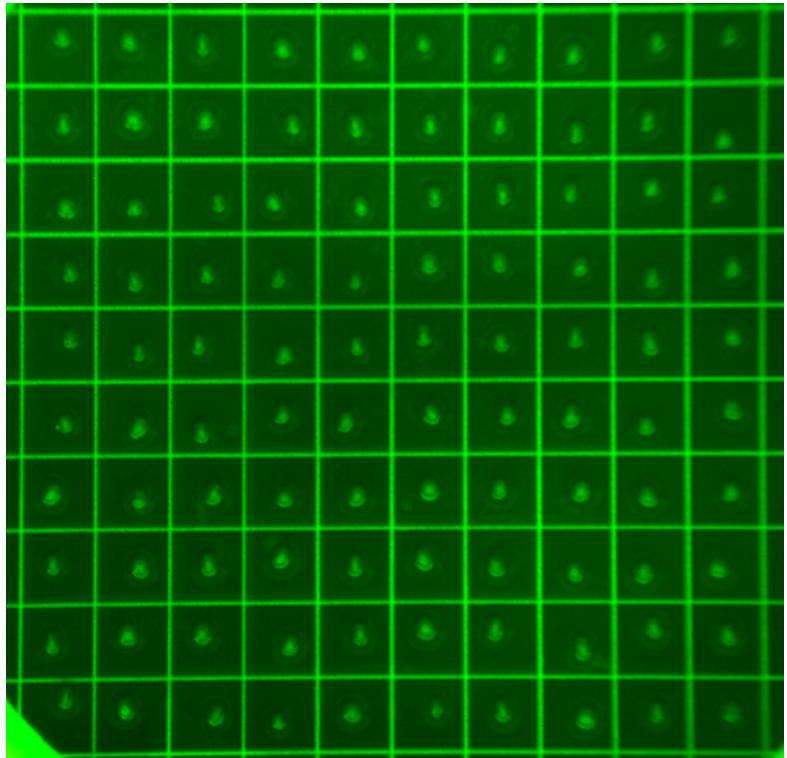
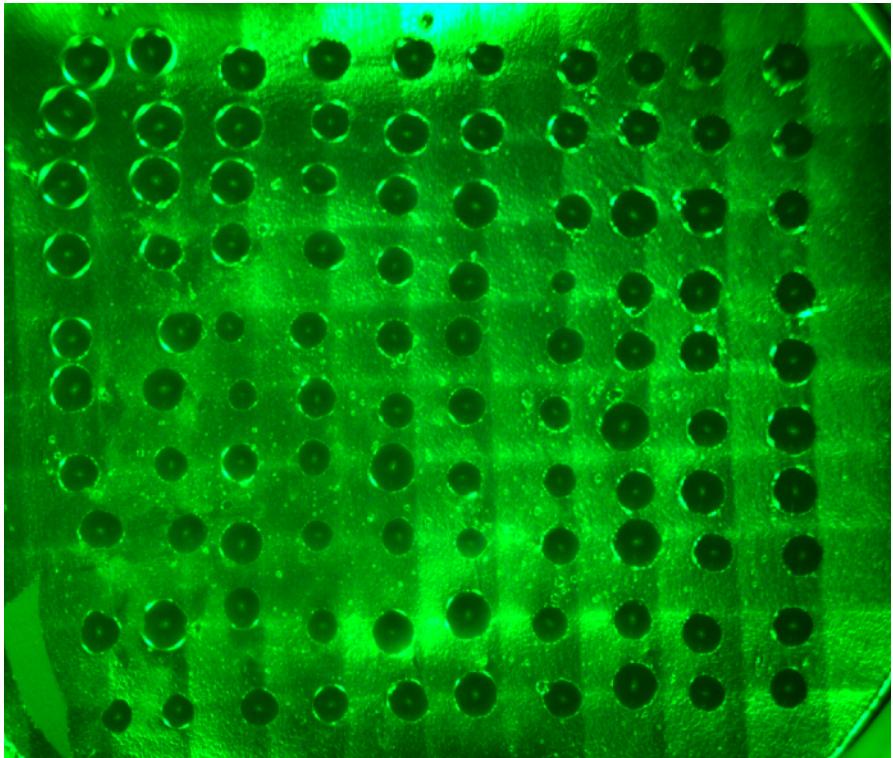


To see necrosis at the cabbage leaf surface



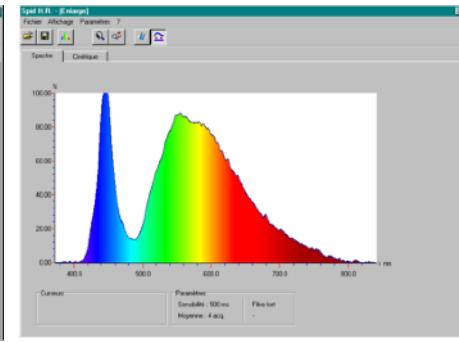
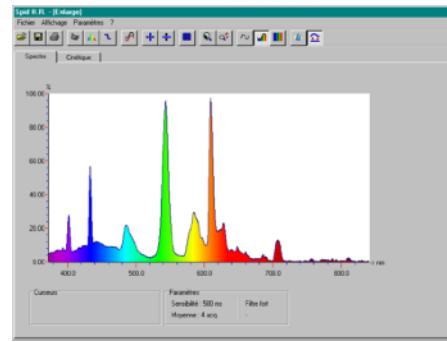
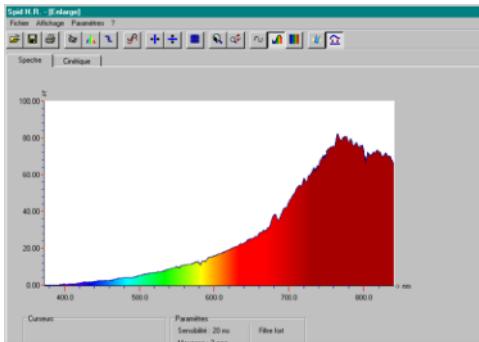
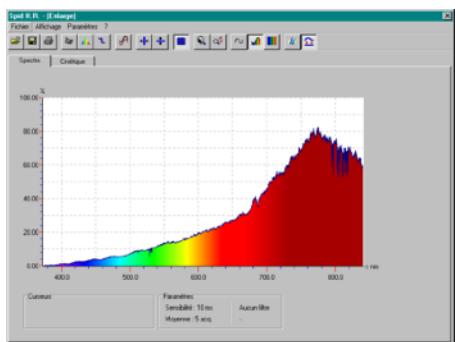
To see alveoli

What's your opinion ?



Light mode ?

If a good contrast exists in the visible spectrum

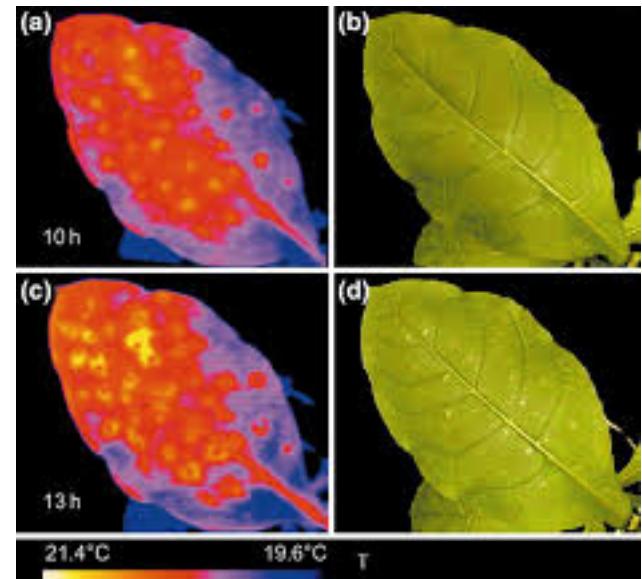
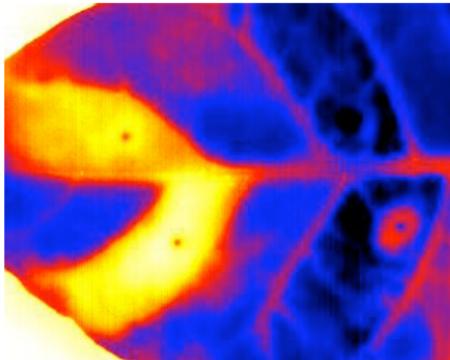


Choice criteria

- Modulation, lifetime, efficiency, size, price
- currently: fluorescent tubes and LED equally used

If thermal contrasts

- Thermal differential



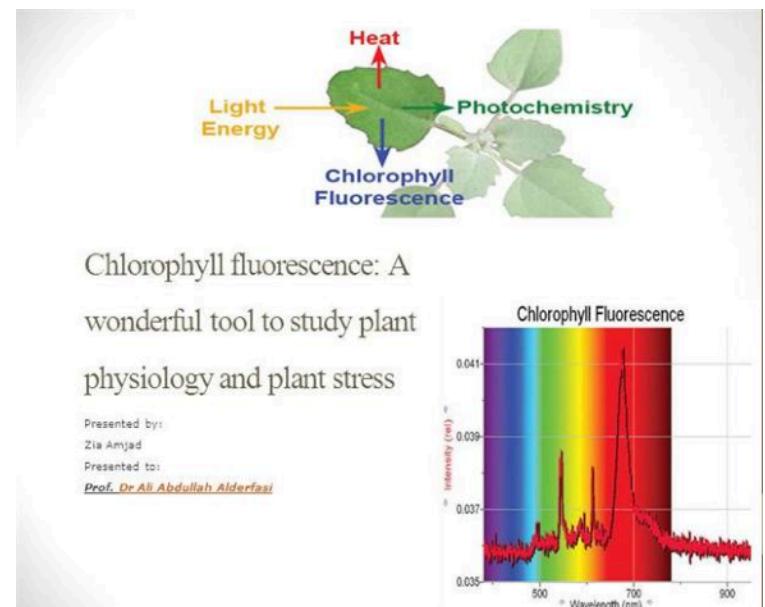
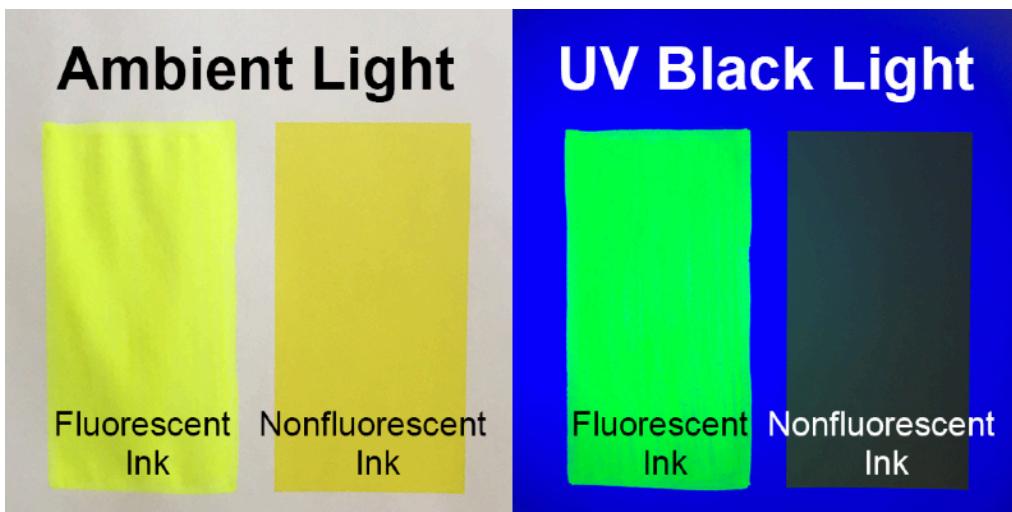
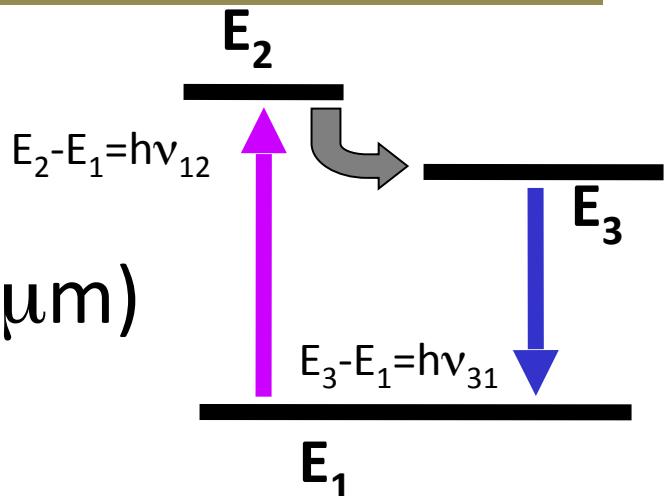
- Difference in surface condition (ex. : same color in visible spectrum but different surface condition)

→ Infrared spectral domain($\lambda > 0,8 \mu\text{m}$)

If contrasts in fluorescent

λ_{12} in UV $\rightarrow \lambda_{31}$ in visible

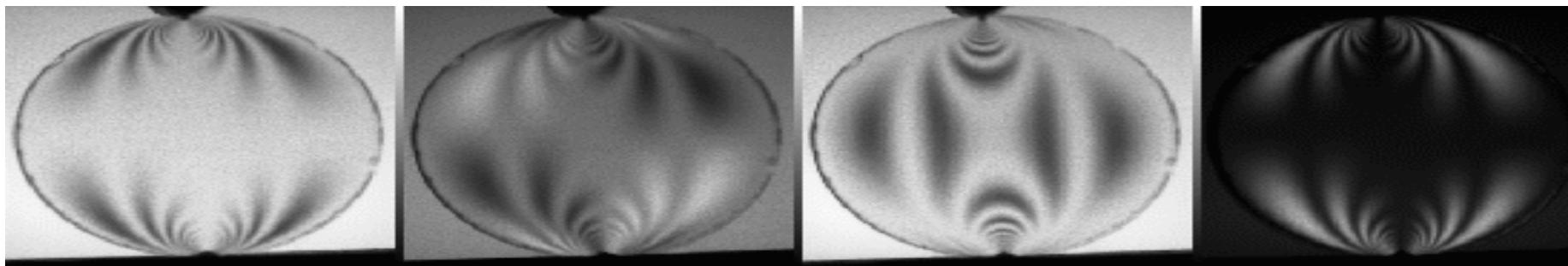
\rightarrow UV spectral domain ($\lambda < 0,4 \mu\text{m}$)



If contrasts due to optical index

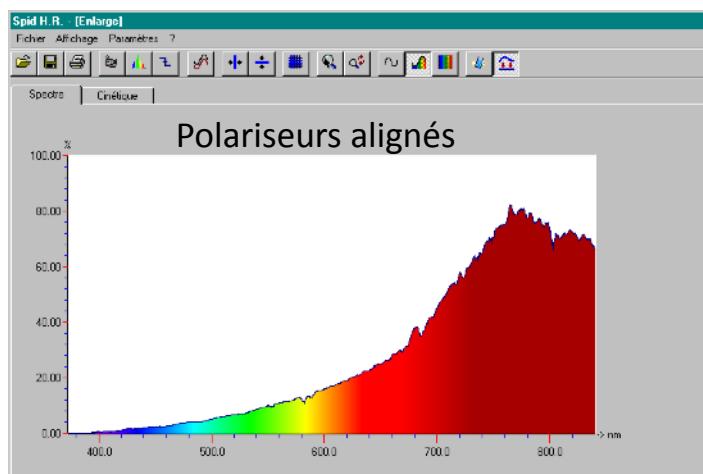
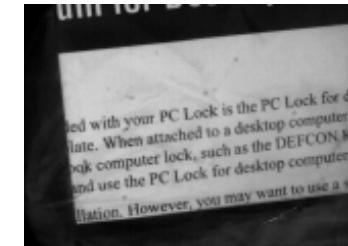
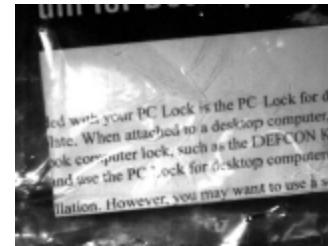
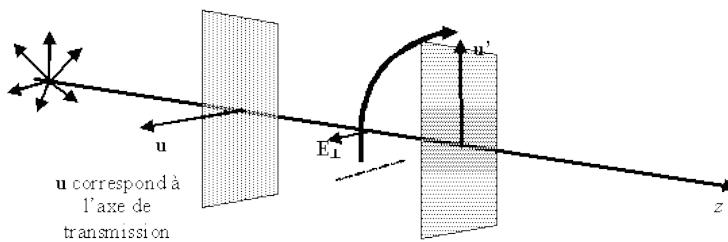
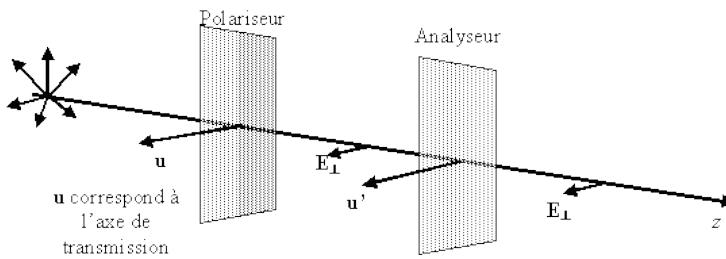
Index of medium may vary due to

- natural (polaroïd, liquid crystals)
- or caused birefringence
 - electrically by pockels or kerr effects,
 - magnetically by Faraday effect,
 - mechanically by photoelasticimetry)

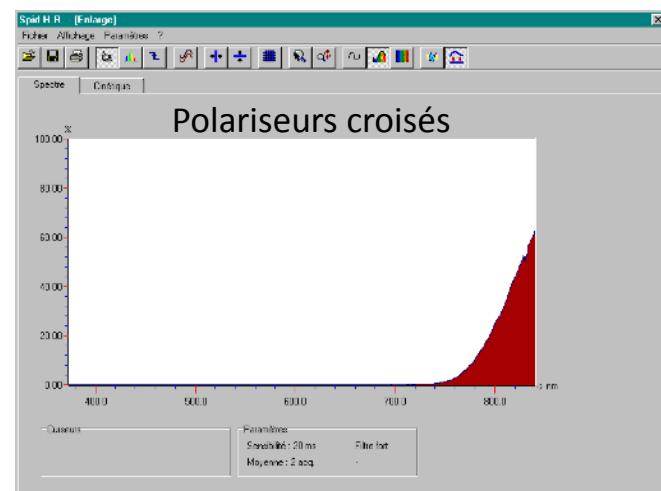


→Polarized light

If contrasts due to optical index

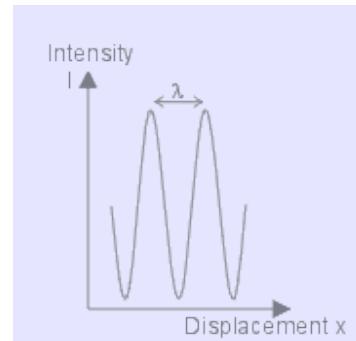
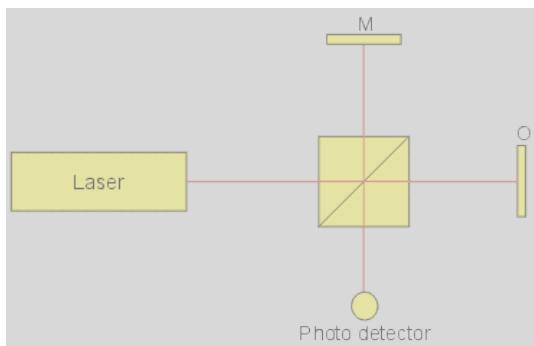


Spectre d'une lampe à incandescence



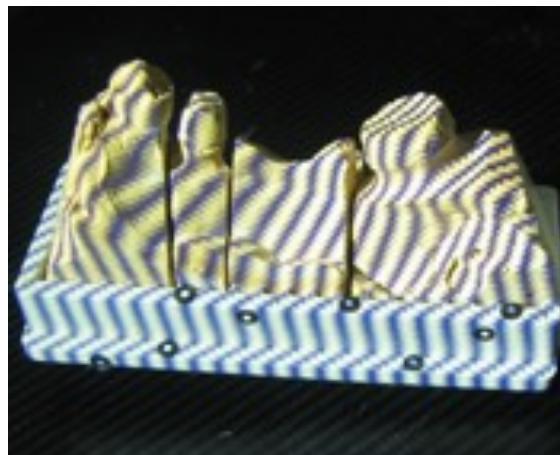
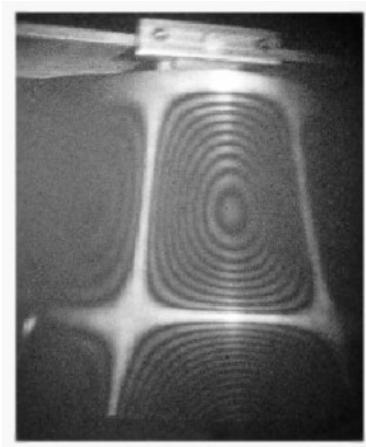
If contrasts are hidden in small details

Size < microns (human vision is not working anymore)



Interferometry

Imaging systems : static or dynamic holography, fringe projectors for topography, vibrometer



→ Temporal coherence of laser light

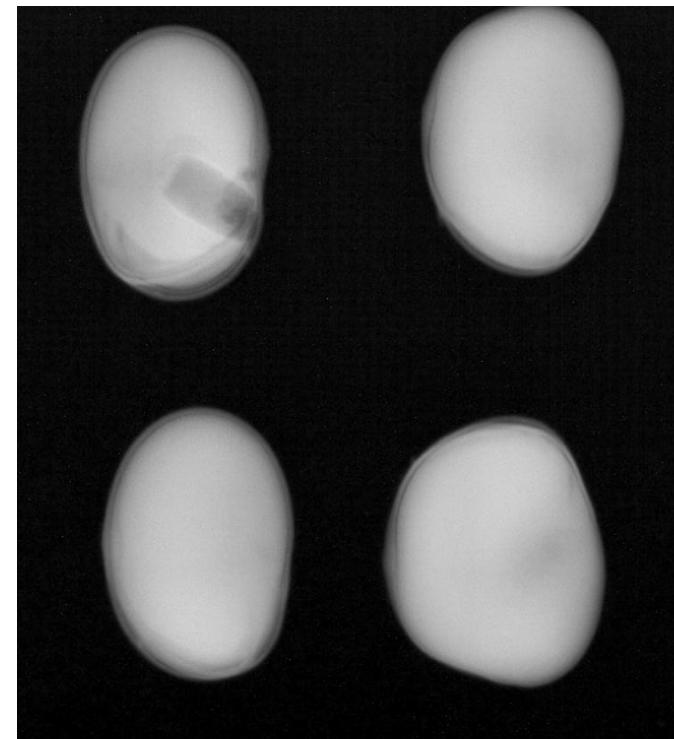
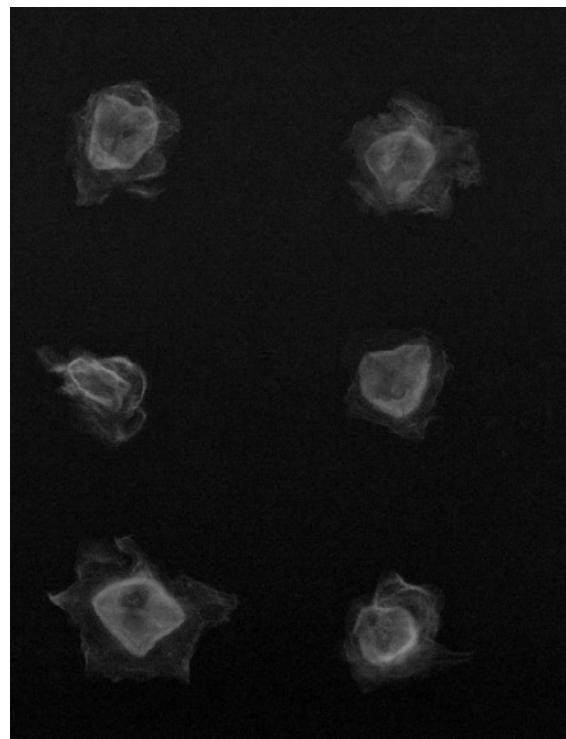
If contrasts are hidden in small details

→ MICROSCOPY

SIZE	METHOD	MAX. MAGNIFICATION	OBSERVATION
1 μm	optical microscope	1500	samples are easy to handle
1 nm	transmission electron microscope	1 000 000	samples may be difficult to be prepared
1 atome	tunneling or atomic force microscope	100 000 000	Very hard to well prepared the samples

If contrasts are inside the samples

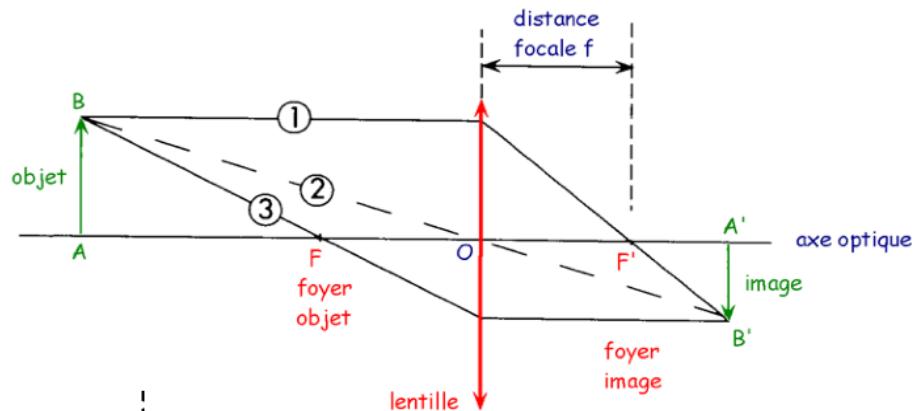
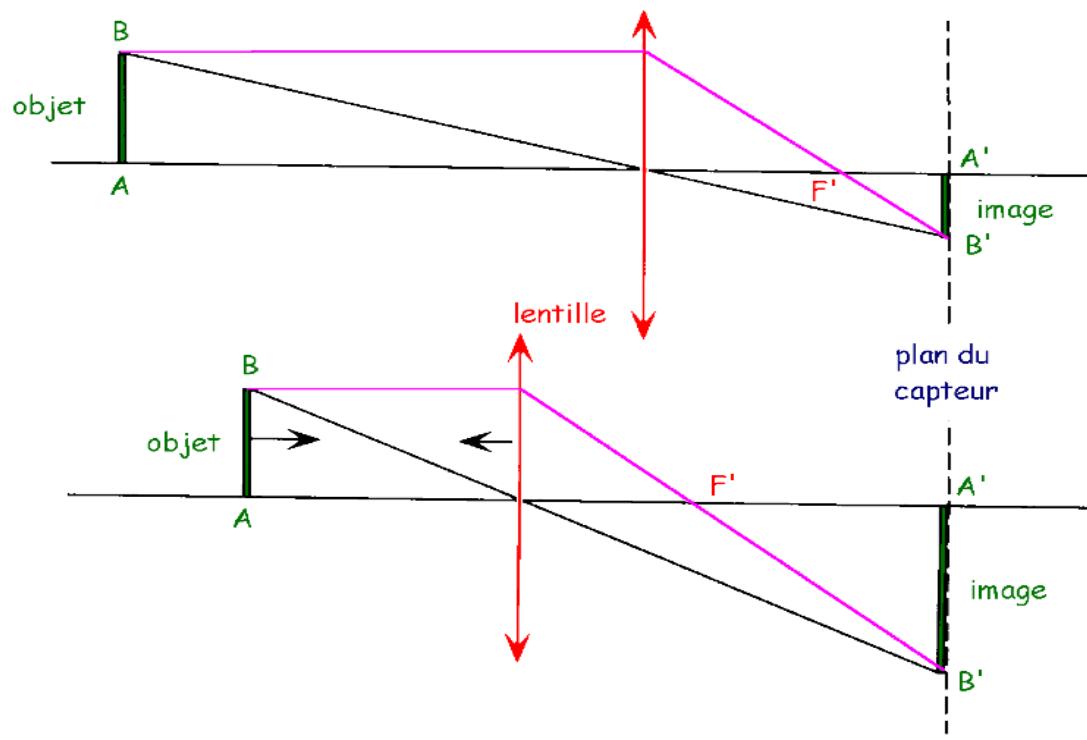
Let's use the invasion power of X-ray



How is an image formed ?

Based on principles of geometrical optics

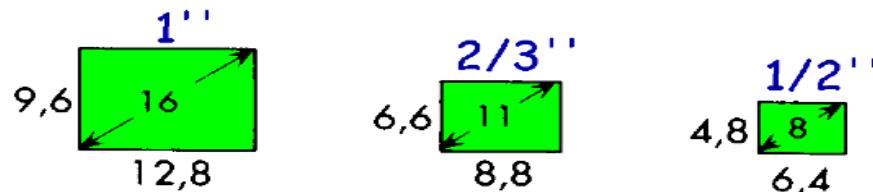
Focus: the lens is moving so that the image of the object (far or close positioned) is located on the sensor



Focal length and size of image

The image of the object is projected at the surface of the sensor. The sensor is characterized by its size (in inch)

CCD or CMOS sensor



cercle image
formé par l'objectif

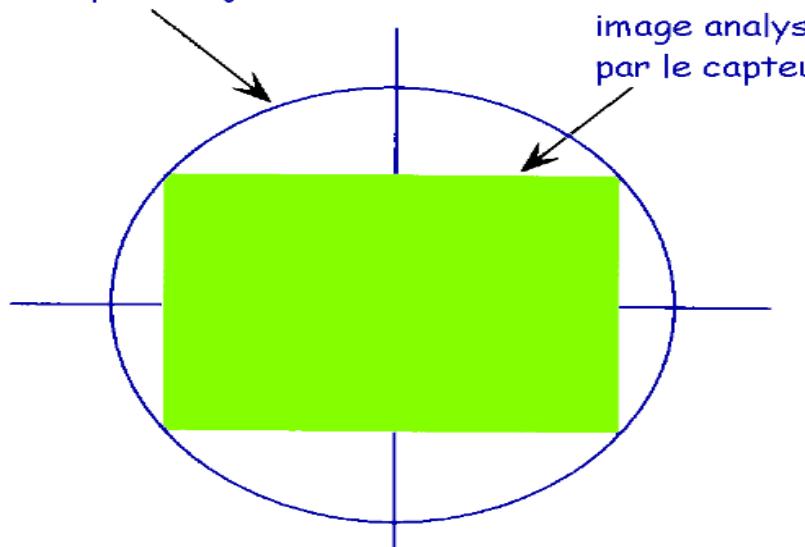
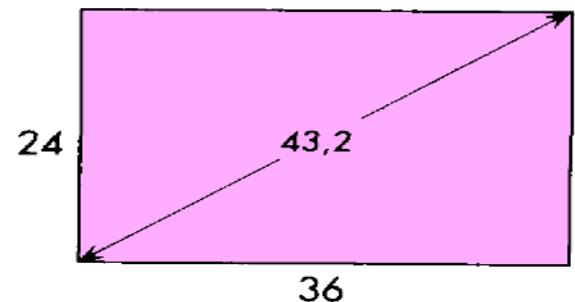
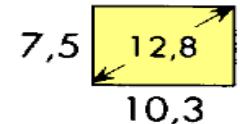
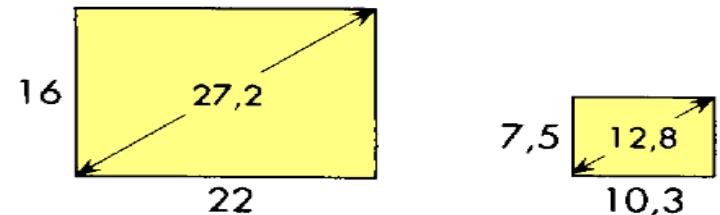


Photo film



Cinema film

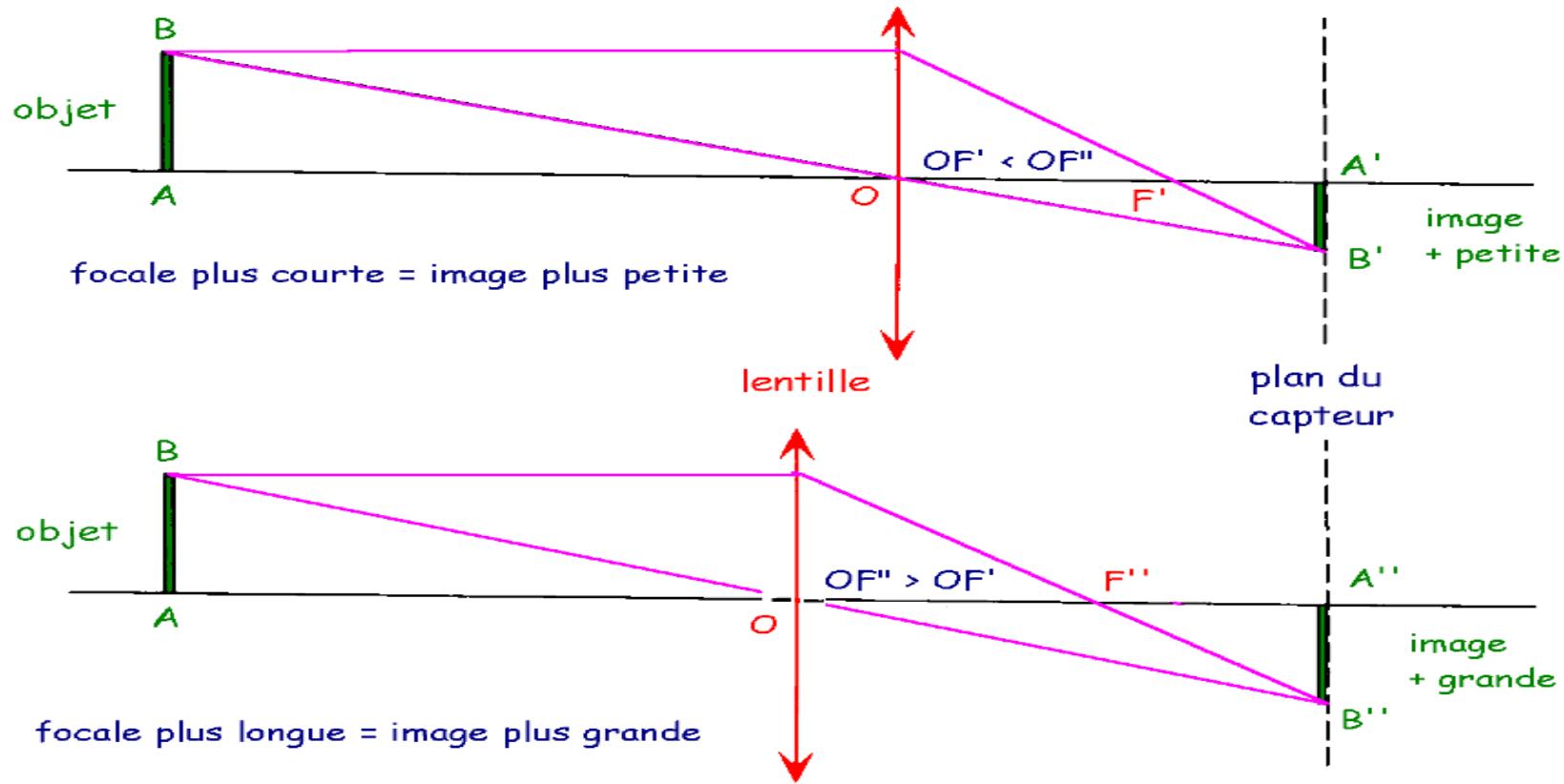


Adjust the focal distance and size of image

To vary the size of the image projected on the sensor:

- move the lens
- change the focal length.

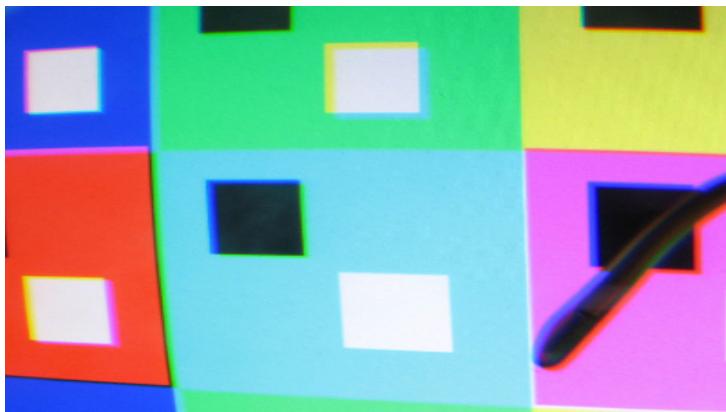
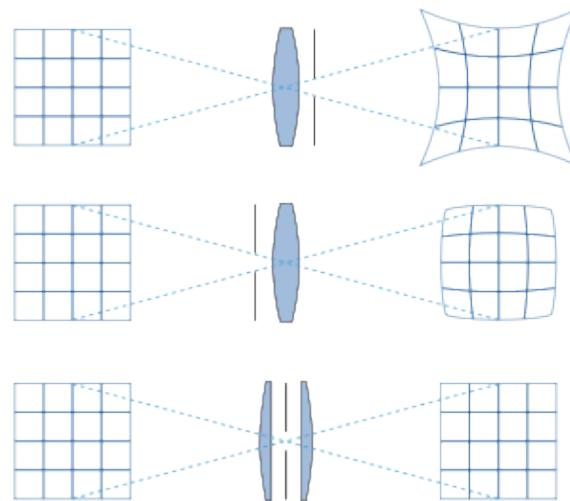
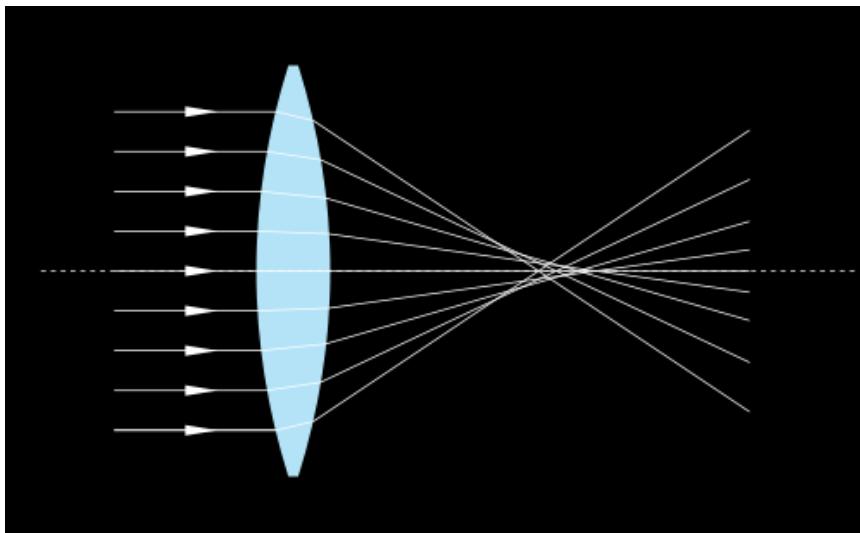
Optimized use of the sensor size: zoom.



Geometric & chromatic aberrations

- Spheric aberrations
- Astigmatism
- Distortion

Coming from geometrical parameters of the system



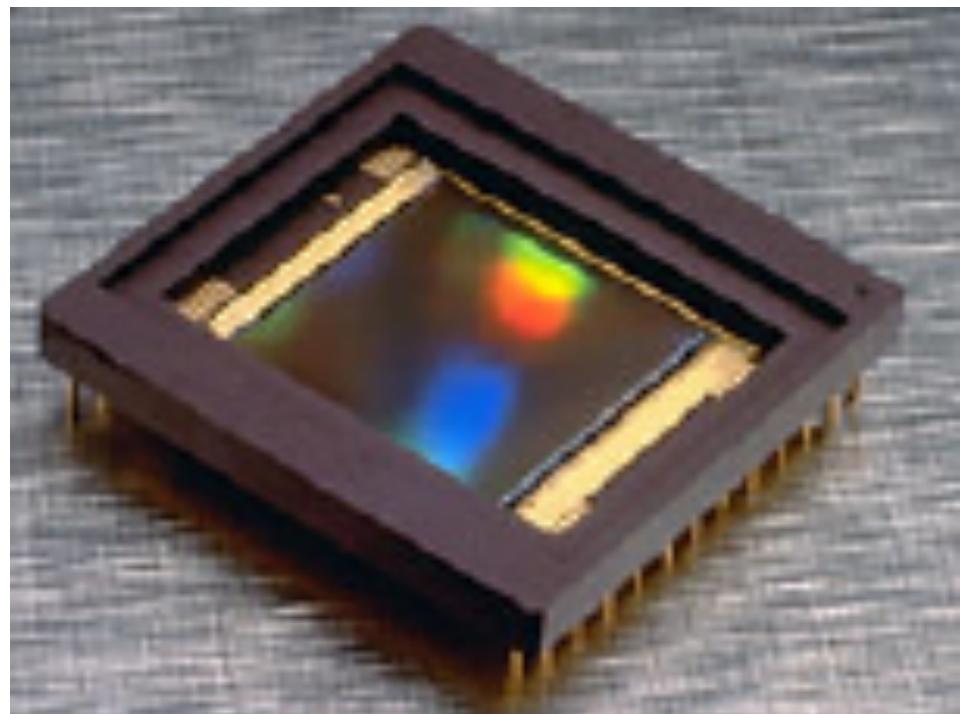
Coming from the decomposition of white light into a color band light

CCD (Charged Coupled Device) sensor

- An integrated circuit with on the upper face an analysis zone composed of multiple photosensitive sites (related to the resolution of the image)
- Built on a silicon substrate
- Each photosite: semiconductor element

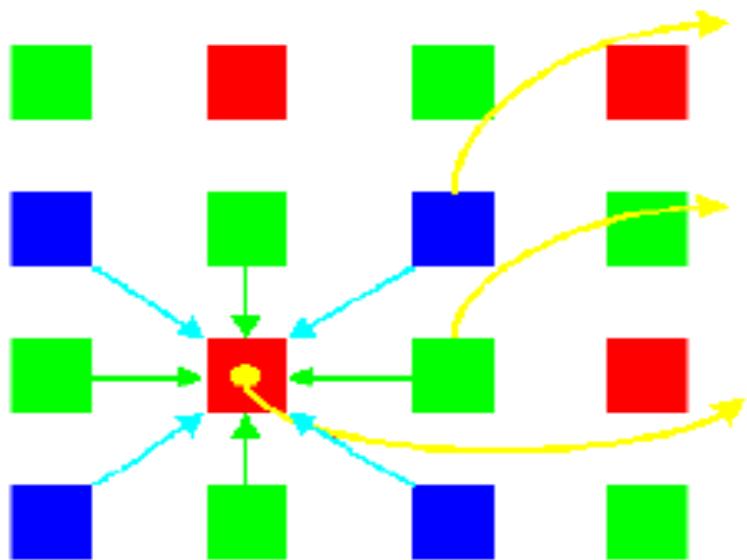
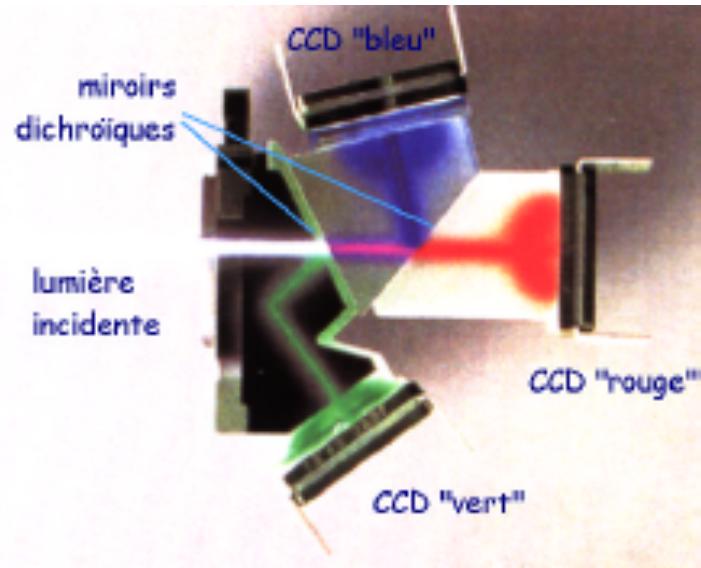
Principle

- Arrival of a photon: photo-electron creation if energy sufficient to cross the threshold of sensitivity (gap)
- The photo-electrons are stored in the potential well (appropriately polarized pixel).
- The reading of these photo-electrons is controlled by polarization via field effect transistors. The reading happens directly with a shutter hiding the source, or by frame transfer.



Monochrome or color images

- Monochrome : only 1 CCD needed
- Color as Red, Green, Blue (primary colors)
 - 3 CCD (one for Red, one for Green and one for Blue)
 - 1 CCD + Bayer mosaic filter



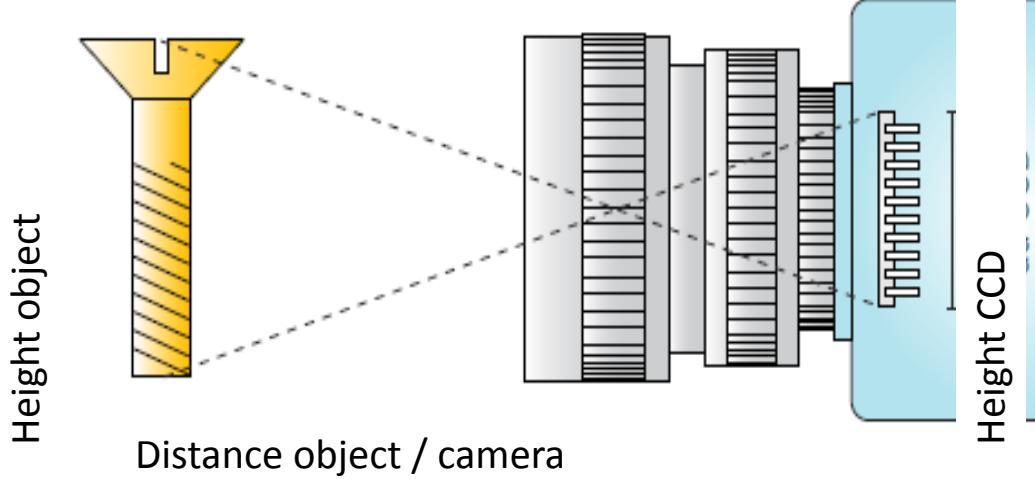
- le photosite mesure l'intensité du bleu
 - le vert est calculé par moyenne sur les 4 verts voisins
 - le rouge est calculé par moyenne sur les 4 rouges voisins
- le photosite mesure l'intensité du vert
 - le bleu est calculé par moyenne sur les 2 bleus voisins
 - le rouge est calculé par moyenne sur les 2 rouges voisins
- le photosite mesure l'intensité du rouge
 - le bleu est calculé par moyenne sur les 4 bleus voisins
 - le vert est calculé par moyenne sur les 4 verts voisins

Let's finish with a small exercise about optic

Exercise : how to choose the right lens ?

The focal length is the main parameter of a lens . To represent an entire object on the CCD, we calculate the focal length of the height and the width of the object. The smallest value is then the focal length of the lens.

- Focal lenght (width) = distance (object/camera) x width CCD / (width object + width CCD)
- Focal length (height) = distance (object/camera) x height CCD / (height object + height CCD)



Format CCD	height CCD [mm]	width CCD [mm]
1/4"	2,4	3,2
1/3"	3,6	4,8
1/2"	4,8	6,4
2/3"	6,6	8,8
1"	9,6	12,8

- Kameras

[FireWire mono](#)[FireWire Farbe](#)[FireWire Bayer](#)

+ Konverter

+ Grabber

+ Software

+ Optik

+ Zubehör

DFK 21AF04 – Spezifikation[Oberblick](#) | **Spezifikation** | [Dokumentation](#) | [Kompatibilität](#) | [Software](#)**Allgemeines Verhalten:**

Video-Formate	640 x 480 YUV (4:2:2) @ 30, 15, 7.5, 3.75 fps
Bildrate	640 x 480 Mono (8 bit) @ 30, 15, 7.5, 3.75 fps
Empfindlichkeit	0.5 lx bei 1/30s, Verstärkung 20 dB
Dynamikbereich	ADC: 10 bit, Ausgang: 8 bit
SNR	ADC: 9 bei 25 °C, Verstärkung 0 dB

Schnittstelle (optisch):

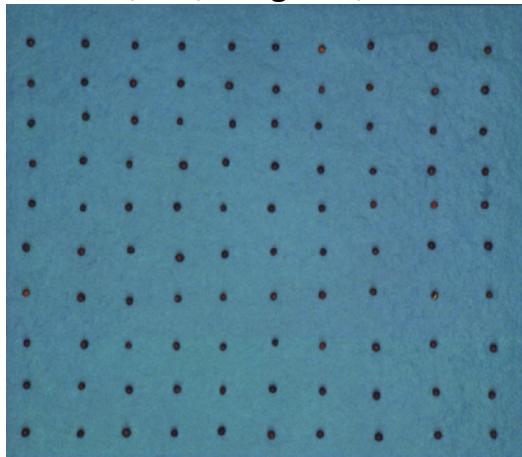
Sensor-Spezifikation	ICX098BQ [671.06 KB, PDF]
Typ	Progressive Scan
Format	1/4 "
Auflösung	H: 640, V: 480
Pixel-Große	H: 5.6 µm, V: 5.6 µm
Objektivanschluss	C/CS-Mount

Schnittstelle (elektrisch):

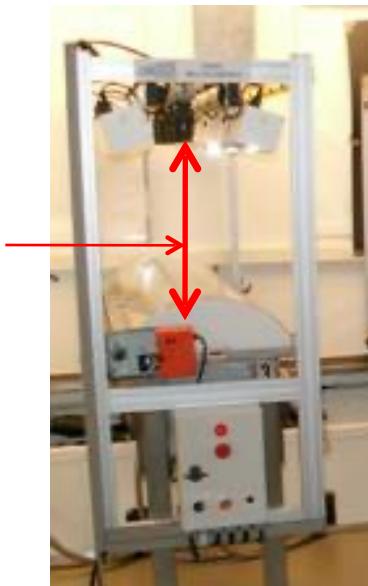
Versorgungs-Spannung	8 bis 30 VDC
Stromaufnahme	ca. 200 mA bei 12 VDC

Object

Width: 0,2m / Height: 0,175m.



Distance: 1m



Questions :

- Which lens do we need to choose ?
- What happens if the distance object/camera is reduced 10 times?
- What happens if the size of my objects is multiplied by 3 ?
- Considering a use of the same chosen tools (camera + lens),
at what minimum distance should I position my system to
be able to correctly image a bush of 1m height ?

Référence	Format	Dist. focale [mm]	MOD [m]
H0514-MP	1/2"	5	0,1
M0814-MP	2/3"	8	0,1
H1214-M(KP)	1/2"	12	0,25
C1614-M(KP)	2/3"	16	0,25
C2514-M(KP)	2/3"	25	0,25
C3516-M(KP)	2/3"	35	0,35
C5028-M(KP)	2/3"	50	0,90

Chapitre 1 : Généralités sur les images numériques

Une image peut être continue selon les coordonnées (X,Y) et aussi en amplitude.

Sous forme numérique :

- digitalisation des coordonnées : **échantillonnage**
- digitalisation de l'intensité : **quantification**
- image matricielle et chaque élément de la matrice : **PIXEL** (Picture Element)
- Résolution spatiale



- Résolution tonale = codage

L'information est codée en binaire : une mémoire élémentaire ou bit (0 ou 1)

- A l'aide de 1 bit on a donc 2 possibilités 0/1 (noir/blanc)
- A l'aide de 2 bits on a $2 \times 2 = 4$ possibilités : 00/01/10/11
- A l'aide de 3 bits on a $2 \times 2 \times 2 = 8$ possibilités :
000/001/010/011/100/101/110/111
- A l'aide de 8 bits on a $2 \times 2 = (2)^8 = 256$ possibilités

Cet ensemble de 8 bits est appelé "octet". En général les informations sont regroupées par groupe de 8, 16, 24, 32 ou 64 bits c'est à dire 1, 2, 3 ou 4 octets.

- Avec 16 bits ou 2 octets on a $2 \times 2 = (2)^{16} = 256 \times 256 = 65536$ possibilités
- Avec 24 bits ou 3 octets on a $256 \times 256 \times 256 = 16777216$ possibilités (plus de 16 millions).

Image NB codée sur 1 bits

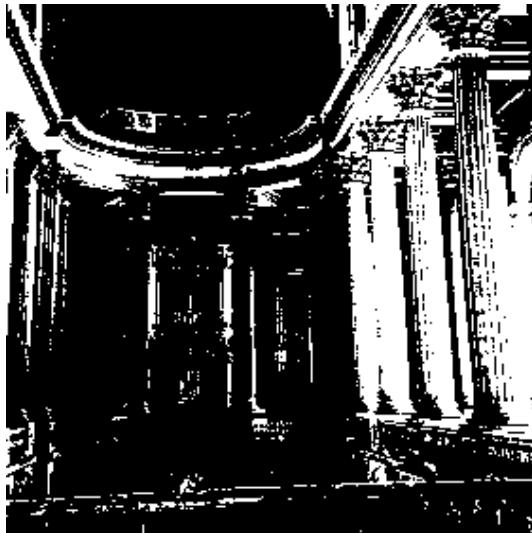


Image en niveau de gris (NG) 8 bits



Image couleur codée sur 3x8 bits



Image couleur codée sur 8 bits



- Effets de la résolution spatiale et tonale



256x256



128x128



64x64



32x32

- Pixellisation
- Loss of clearness
- Less visible details



6 bits



4 bits



3 bits



2 bits



1 bit

- Appearance of fake edges
- Quantification noise

Source : Antoine Mazanera. *Introduction et filtrage*. ENSTA (France).

Various formats for digital images

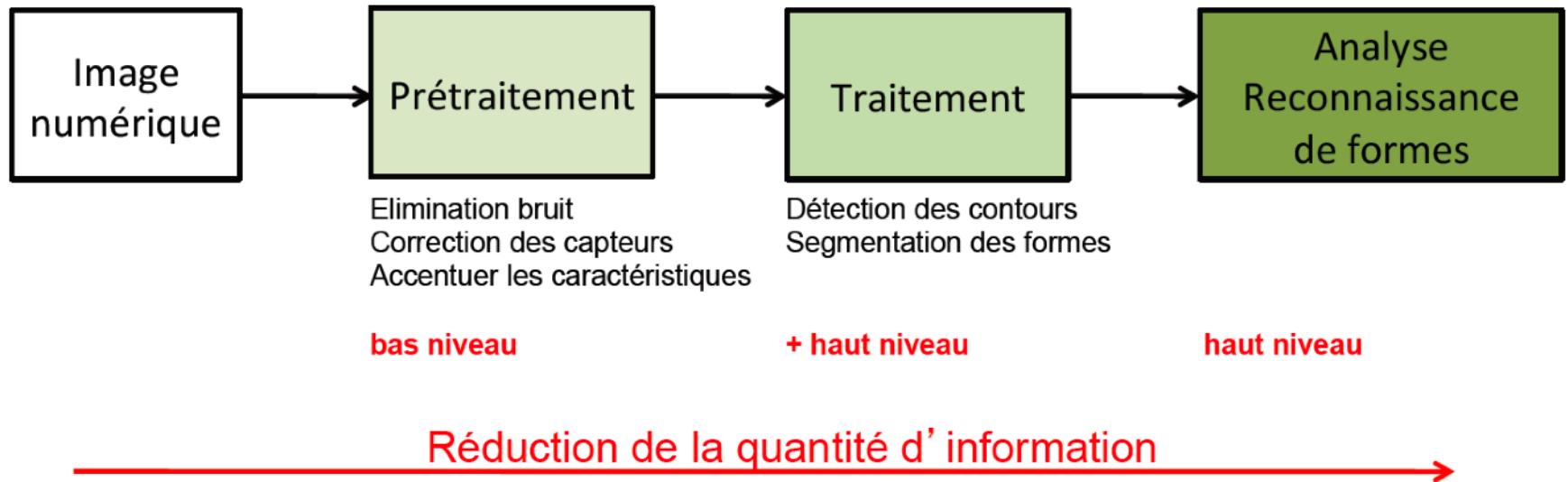
JPEG, JPEG2000, PNG, GIF, TIFF, BMP, XMP, Targa , FITS, PM, What characterizes them?

Main characteristics intrinsic to a format:

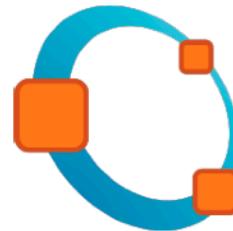
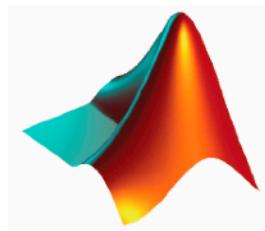
- status in relation to patents: free formats and some " Owners".
- number of colors supported
- data compression: lossless or lossy (JPEG)
- ...
- transparency: One of the colors of the palette can be ignored when display
- interlacing: displaying a low resolution version refined when loading
- animation movies, 3D visualization
- uses: scientific computing , archiving, Internet

Format	Compression des données	Nb de couleurs	Affichage progressif	Format propriétaire	Usage
BMP	Non compressé	de 2 à 16 millions	Non	Non	Image non dégradée mais très lourde. Stockage
JPEG	Réglable, avec perte de qualité. Plus la compression est importante, plus l'image est dégradée. ⚠️ format destructeur	16 millions	Oui	Non, libre de droits	Tous usages, selon compression. Images "naturelles".
GIF	Oui, sans perte de qualité	de 2 à 256 avec palette.	Oui	Brevet Unisys	Logos et Internet. Supporte les animations et la transparence.
TIFF	Réglable, au choix sans perte ou avec perte de qualité	16 millions	Non	Brevet Aldus corporation	Tous sauf Internet
PNG	Oui, sans perte de qualité	de 2 à 256 ou 16 millions	Oui	Non, libre de droits	Tous, recommandé Internet mais incompatible avec les navigateurs anciens. Supporte la transparence.

- Structure d'une chaîne de traitement d'images et logiciels adaptés



Pléthores de solutions pour du traitement d'images avec un minimum de programmation

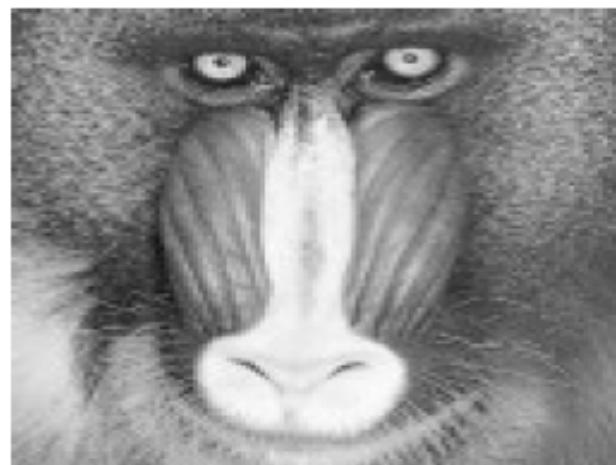
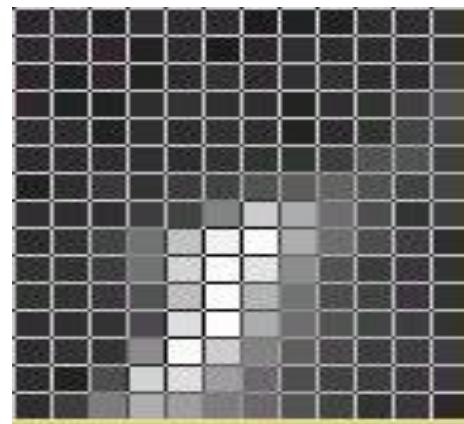


```
17 string >>>
18 int iN;
19 double dblTemp;
20 bool again = true;
21
22 while (again) {
23     iN = -1;
24     again = false;
25     getline(cin, sInput);
26     system("cls");
27     stringstream(sInput) >> dblTemp;
28     iLength = sInput.length();
29     if (iLength < 4) {
30         again = true;
31     }
32     continue;
33     if (Input[iLength - 3] != '.') {
```

rsbweb.nih.gov/ij/

Chapitre 2 : Caractérisations des images numériques

Digital image = Matrix = Tab



Intensity (grey-level)

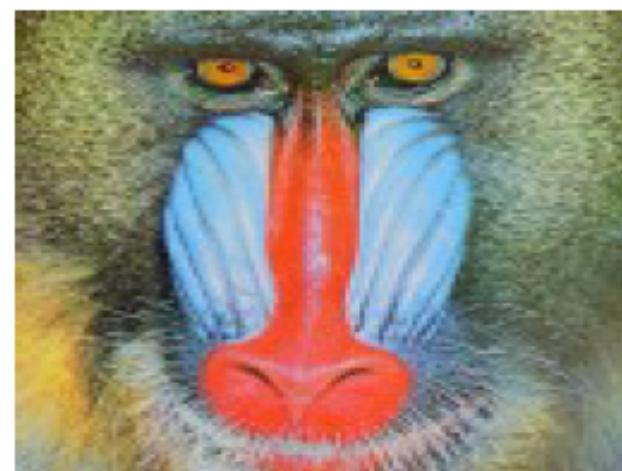
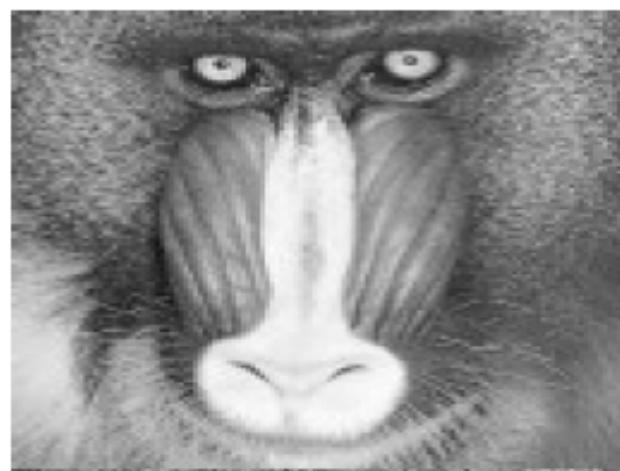


Binary



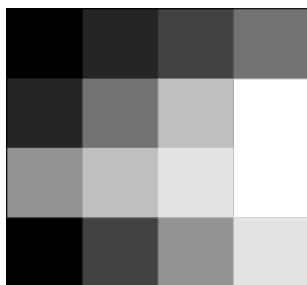
Color (RGB)

Types of digital images



Intensity (grey-level)

$I(x,y)$ sur $[0 \dots 255]$
Mainly coded on 2^8 levels



0	1	2	3
1	3	5	7
4	5	6	7
0	2	4	6

Example: grey-level coded on $2^3 = 8$ levels

Binary

$I(x,y)$ sur $[0, 1]$
Coded on 2^1 levels

000	001	010	011
001	011	101	111
100	101	110	111
000	010	100	110

Color (RGB)

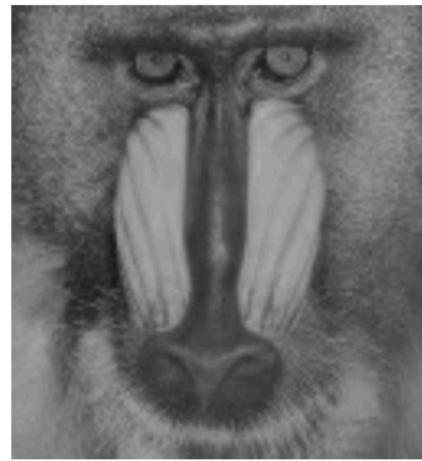
$I_R(x,y)$ $I_G(x,y)$ $I_B(x,y)$
Coded on 3×2^8 levels

0	1	1	1
1	1	1	0
1	0	0	1
1	0	1	1

Talking about RGB image...



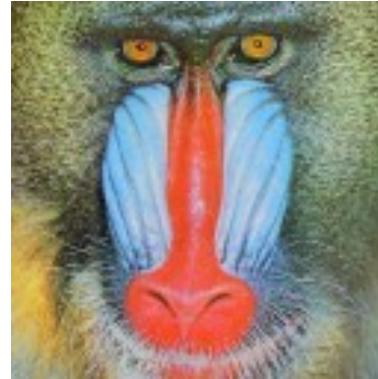
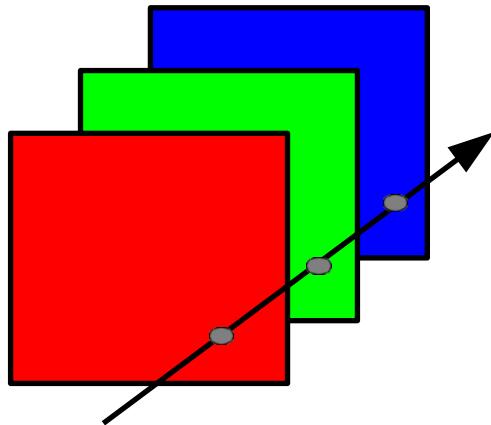
Intensity (grey-level)



Intensity (grey-level)



Intensity (grey-level)



Also existing other color spaces than RGB ...

Still talking about RGB image...



Intensity (grey-level)

Binary

Color (RGB)

Le passage de couleur vers intensité est obtenu par la formule suivante :

$$I = \alpha R + \beta G + \gamma B \text{ où } \alpha = 0.2989, \beta = 0.5870 \text{ et } \gamma = 0.1140.$$

On remarquera qu'il s'agit d'une somme pondérée des trois composantes de couleur. La somme des coefficients de pondération vaut 1.

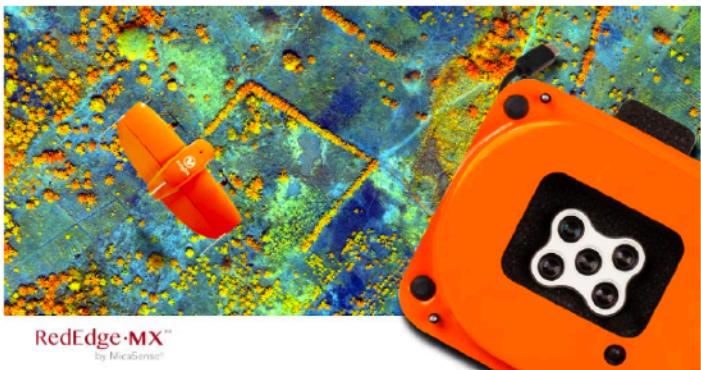
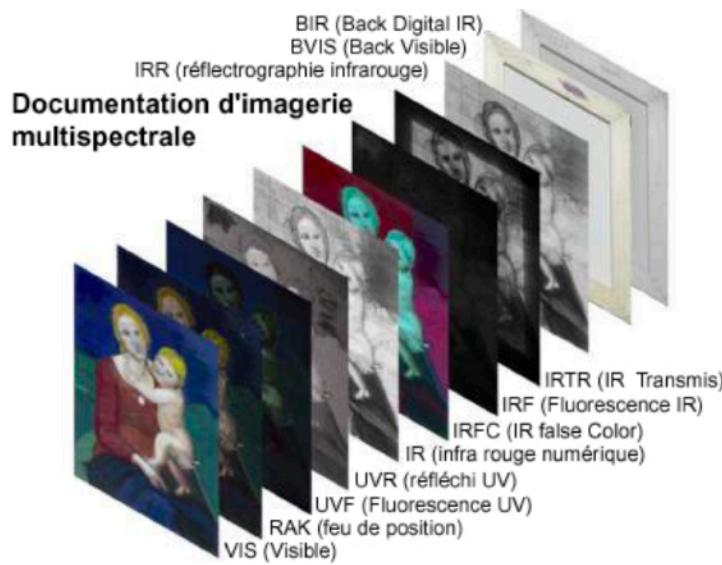
Ces coefficients ont été obtenus par des mesures psychovisuelles. On observe que le coefficient de pondération le plus important est celui correspondant à la composante verte. Pourquoi ?

Converting images

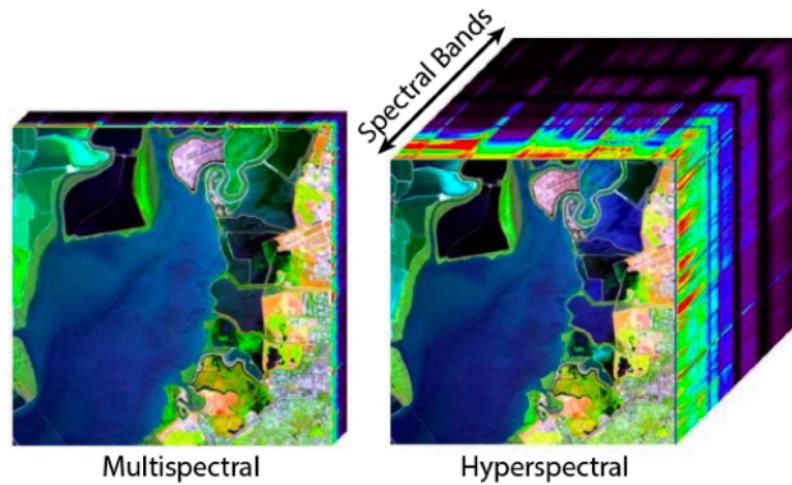
to	from	8-Bits	8-Bits Color	16-Bits	32-Bits	RGB Color
8-Bits			LUT removing	Nothing in appearance. Loss of values and loss of precision.	Nothing in appearance. Loss of values and loss of precision.	Grey-level = average of RGB channels.
8-Bits Color	Nothing			Nothing in appearance. Loss of values and loss of precision.	Nothing in appearance. Loss of values and loss of precision.	Nothing in appearance. LUT creating.
16-Bits	Nothing	LUT preserving			Nothing in appearance. Loss of values and loss of precision.	Loss of colors
32-Bits	Nothing	LUT preserving	Nothing in appearance.			Loss of colors
RGB Color	RGB channels = grey-level	Nothing in appearance. Creating RGB channels	RGB channels = grey-level	RGB channels = grey-level		

Going further...

Multispectral imaging

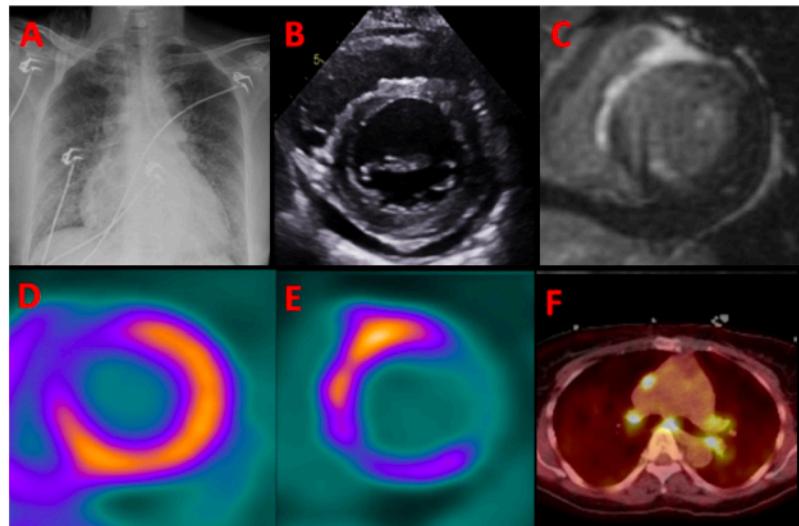
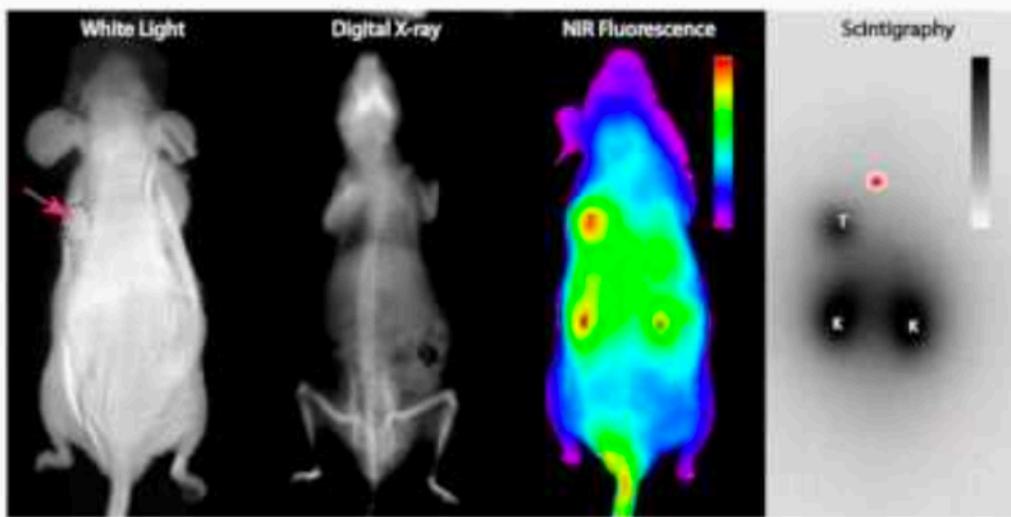


- Développées initialement pour les satellites.
- Capturent la lumière dans un spectre électromagnétique allant au-delà du visible.
- Constituées de multiples composantes
- Pour chaque pixel de l'image : un vecteur qui donne le spectre d'absorption (ou réflexion) dans les différentes longueurs d'ondes.



Going further...

Multimodal imaging



Tumor imaging with separate fluorescent and radionuclear tumor-targeted imaging agents.

3/10/2018

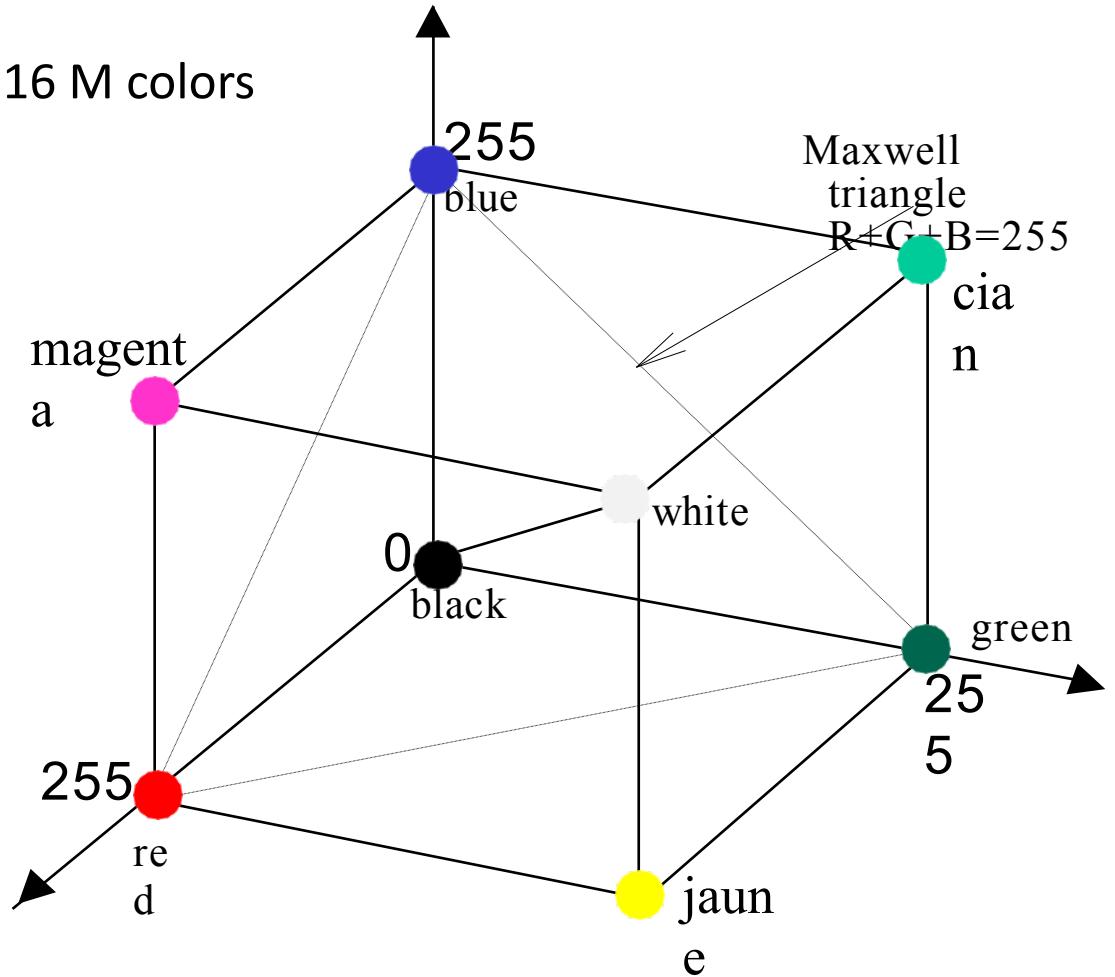
Optical Radiology Laboratory

- Couplage de différents types d'imagerie (RGB, électromagnétique en réflexion ou absorption dans toutes les longueurs d'ondes, fluorescence, thermographie infrarouge, ultrason, IRM, scintigraphie, . . .)
- Visualiser différents paramètres physiques
- Recalage (registration) : problématique actuelle en recherche...

Back to color spaces: RGB & CMY

RGB: Red, Green, Blue & CMY: Cyan, Magenta, Yellow

- Additive synthesis of the color
- Screen, graphical card
- Images 24 bits (3×8 bits) = 16 M colors

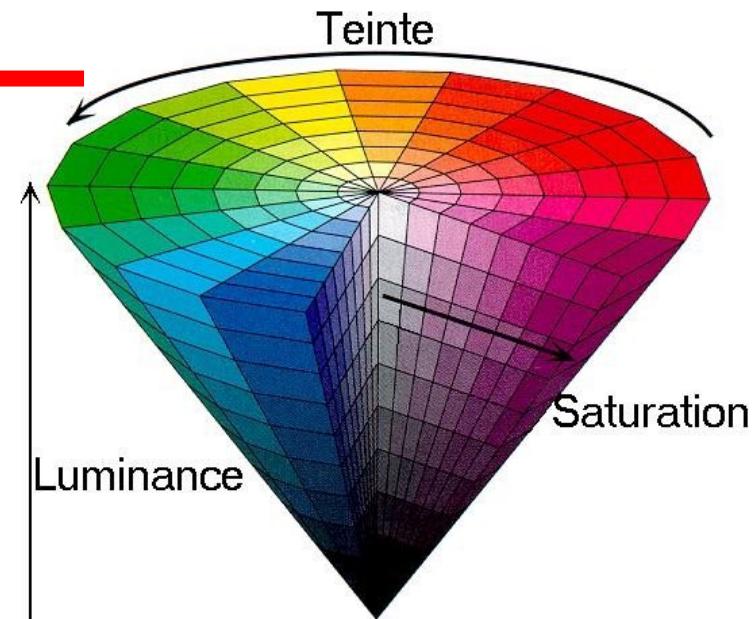


Color space: HSL

HSL: Hue, Saturation, Luminance

- Dominant wavelength (hue)
- Contribution (saturation)
- Intensity by surface unit (luminance)

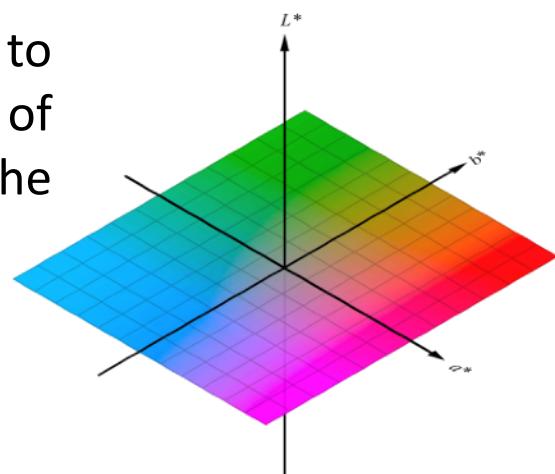
More closely align with the way human vision perceives color-making attributes



Color space: Lab

- L: lightness
- a and b: the green–red and blue–yellow color components.

CIELAB was designed to be perceptually uniform with respect to human color vision, meaning that the same amount of numerical change in these values corresponds to about the same amount of visually perceived change.

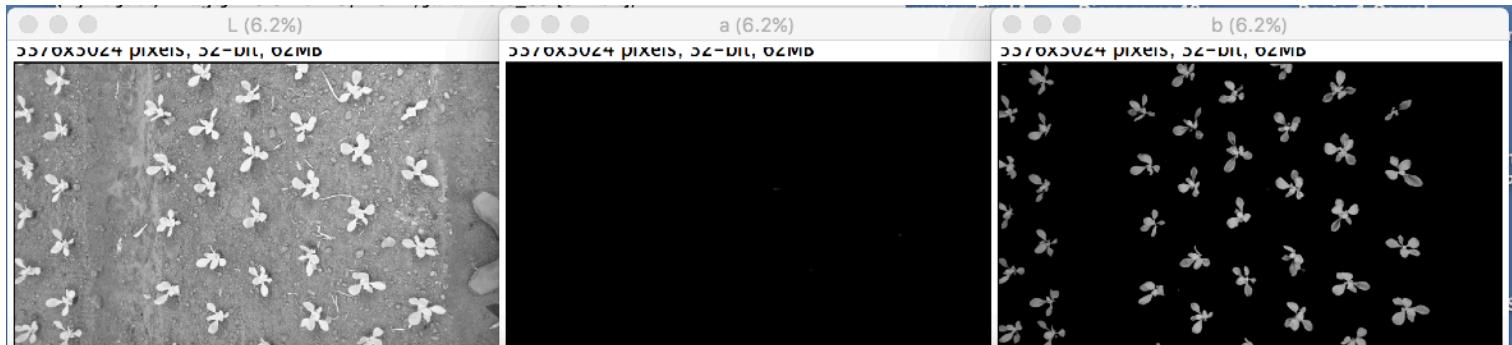


Example: color spaces

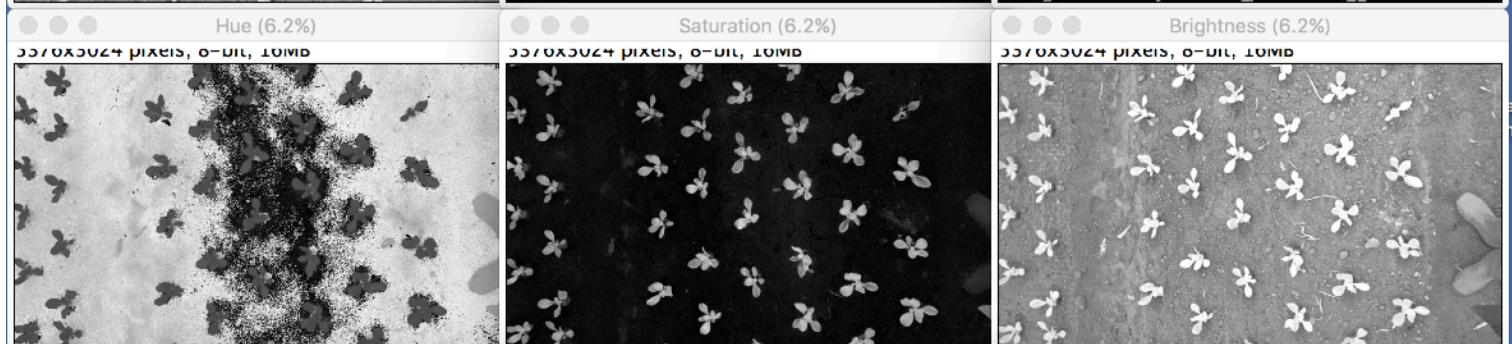


RGB

Lab



HSV

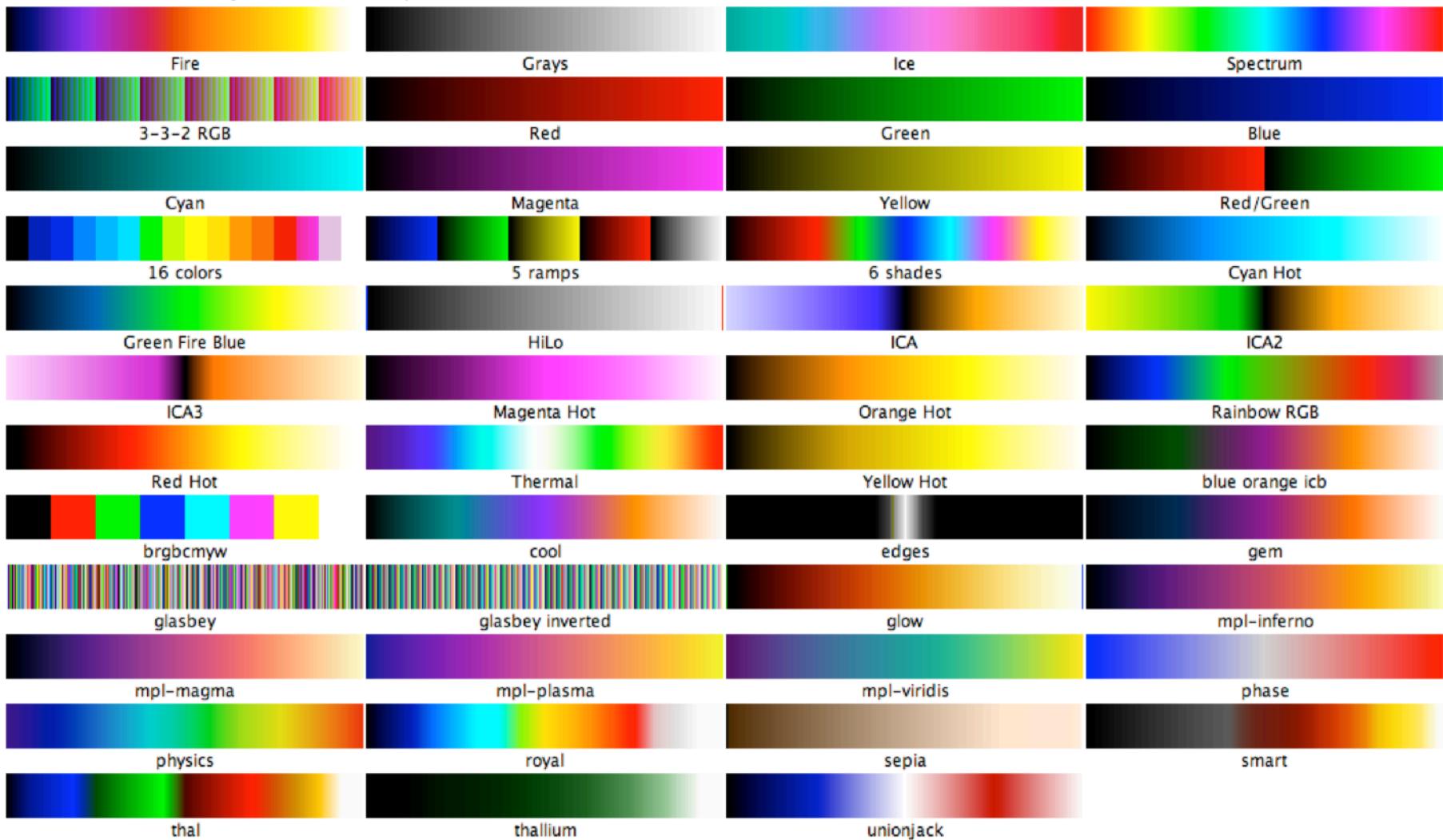


RGB



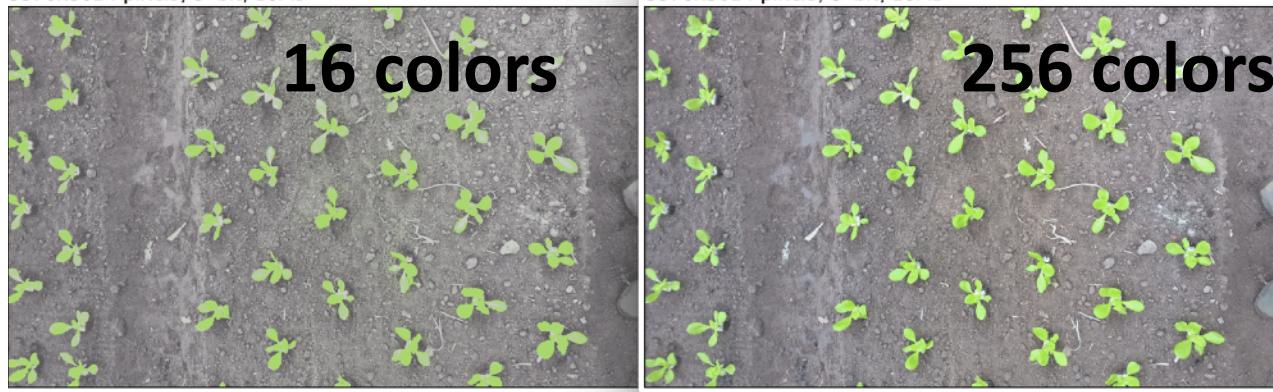
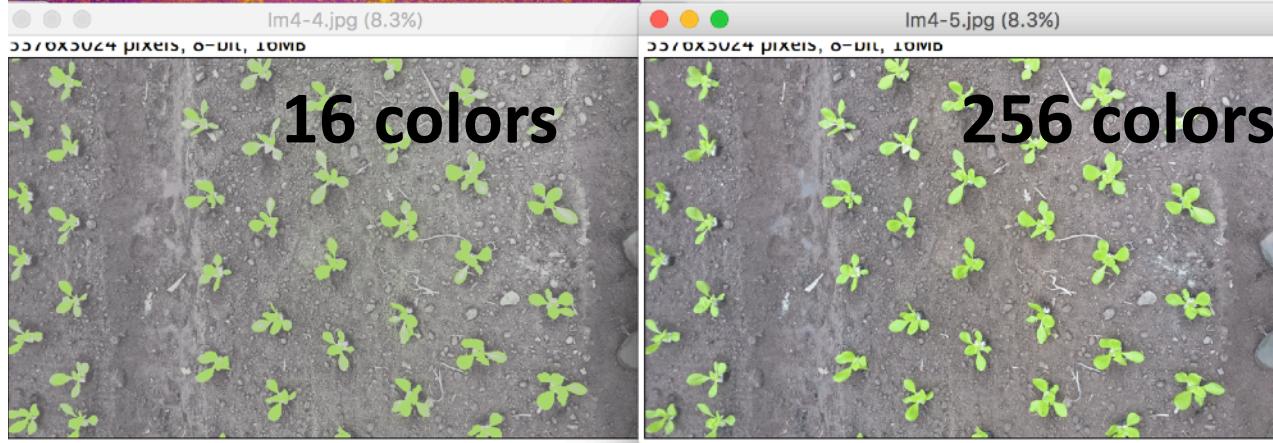
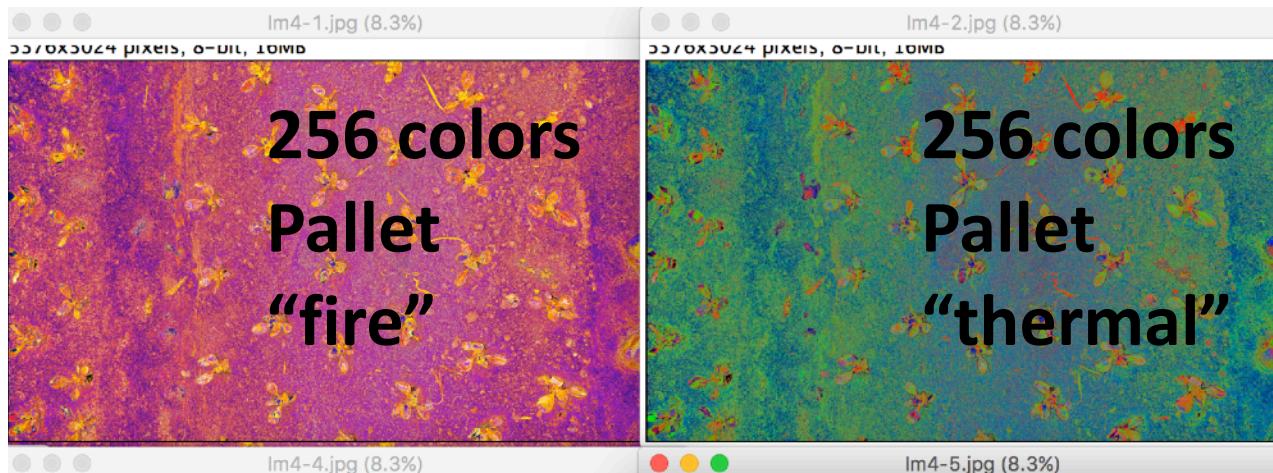
Color pallet / LUT (Look Up Tables)

- 16 M of colors → 256 colors using color pallet/tables
- Indexed image = color pallet + index matrix



Examples

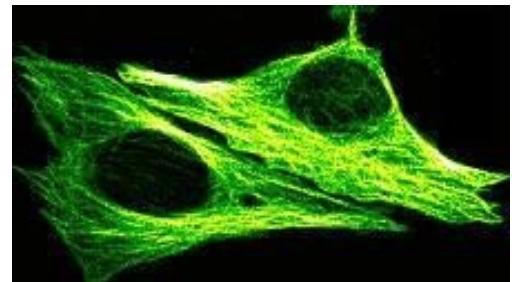
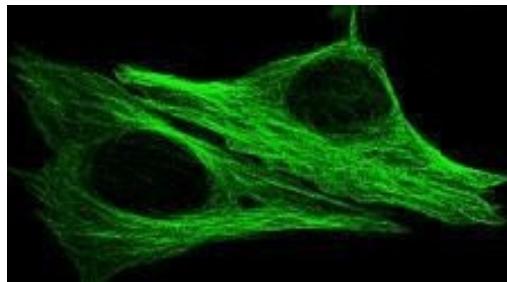
RGB 16 Mcolors



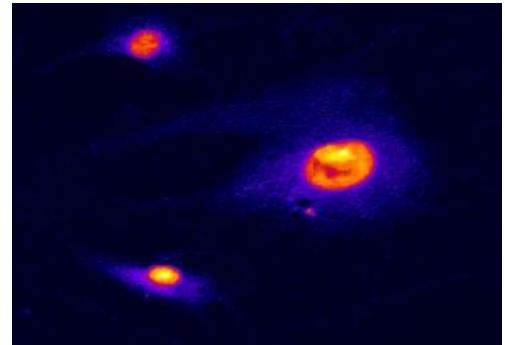
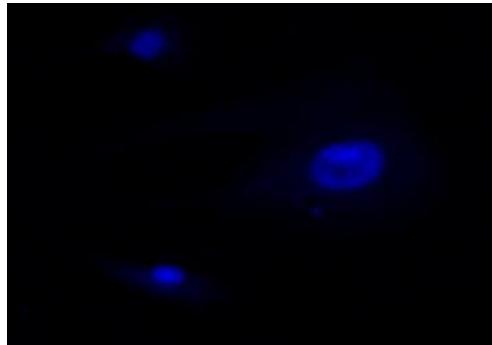
Using LUT

The judicious use of LUT can be very useful to highlight the characteristics of an image. Indeed, the human eye can perceive relatively few shades of colors in the same scene. In addition, it should be noted that the sensitivity of the human eye to light intensities related to wavelengths is uneven. It is easier to perceive shades of yellow, orange or turquoise than shades of dark blues or reds. The pseudo-coloring of the images can therefore make the data more visible without altering the data included in the image.

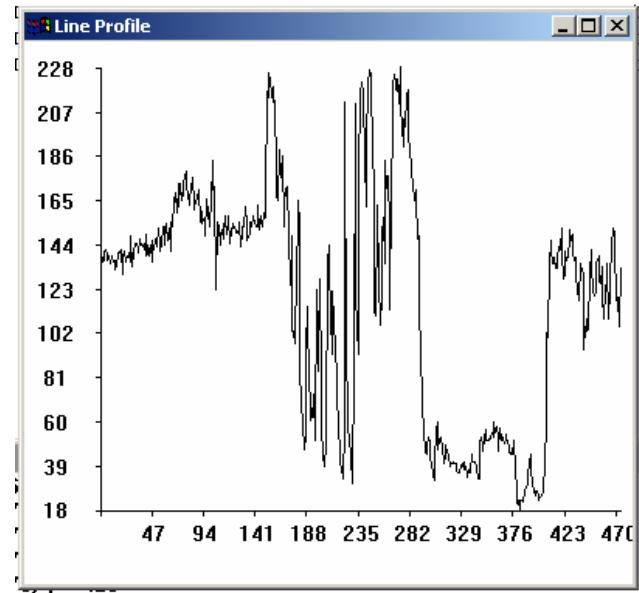
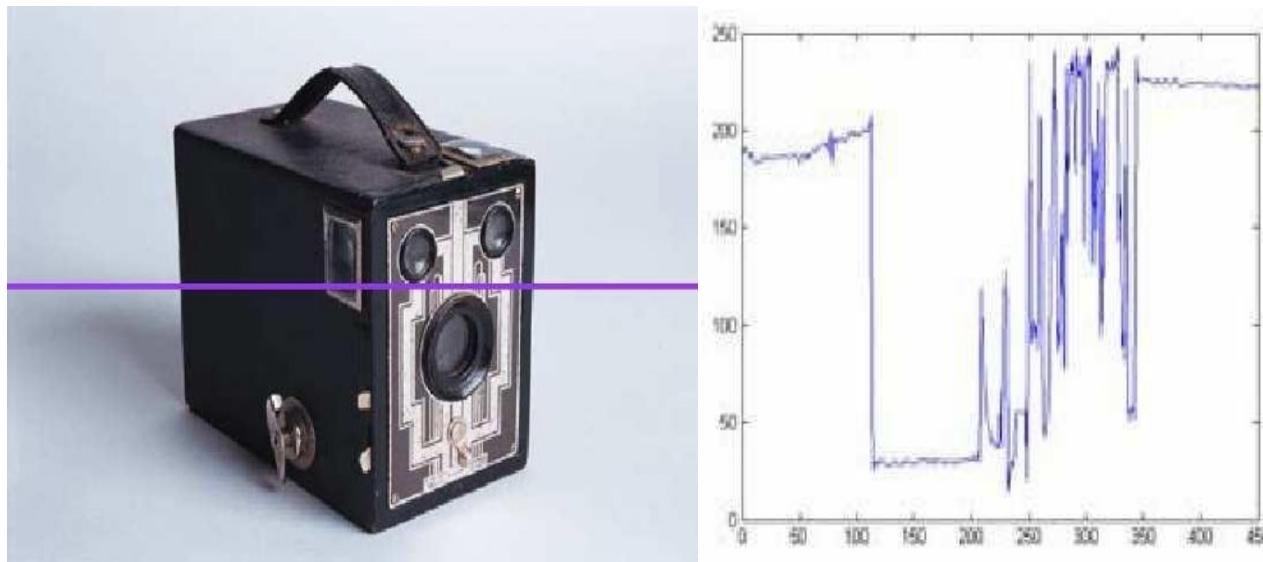
- *LUT classic « Green »*
- *LUT « Hot Green »*



- *LUT classique « Blue»*
- *LUT « Fire »*

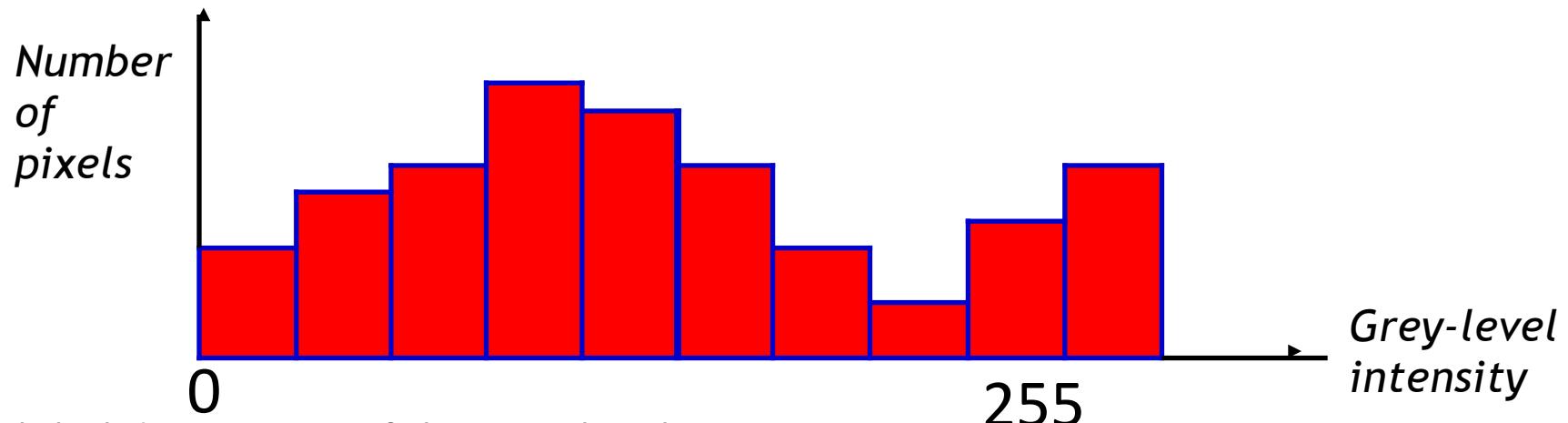


Intensity profile

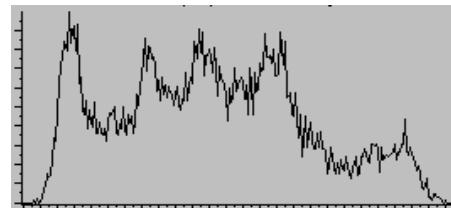


Histogram

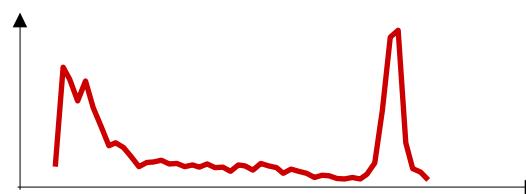
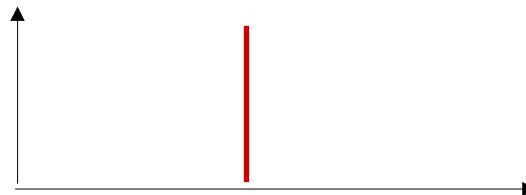
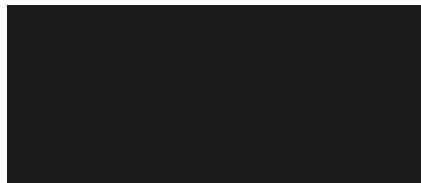
Histogram = displaying number of pixels according to grey-level intensities (0-255)



- Global description of the grey-level int.,
- Can be seen as probability distribution of grey-level int.,
- Useful tool to analyze, to improve or transform image,
- The simplest segmentation techniques come from the analysis of the histogram.



Histogram



Dynamic of an image = [min value ,max value]

Histogram

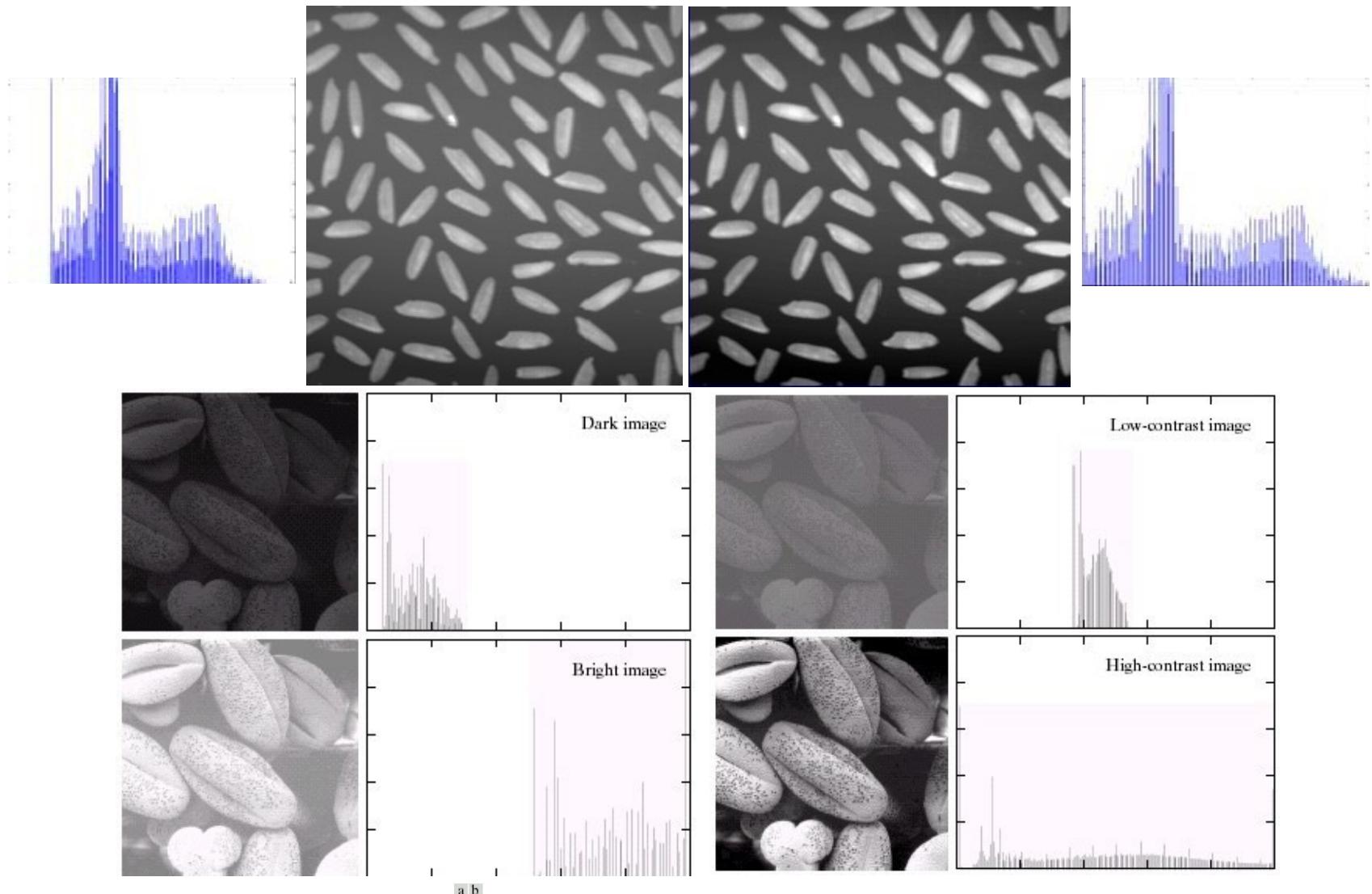


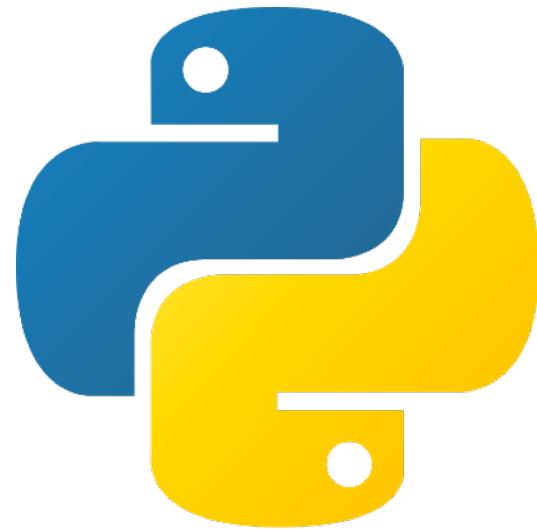
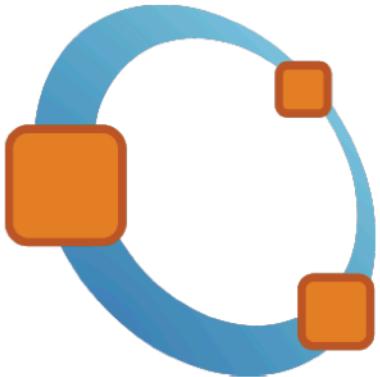
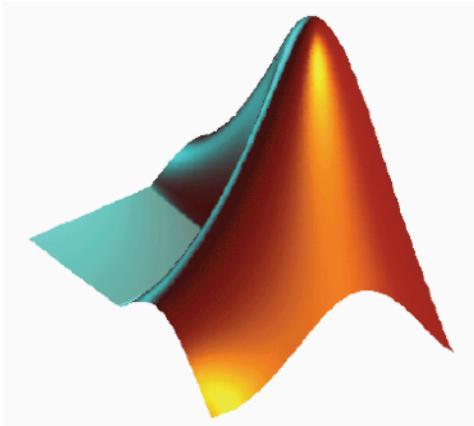
FIGURE 3.15 Four basic image types: dark, light, low contrast, high contrast, and their corresponding histograms. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

HANDS ON n°1

- 1) ouvrir lena.jpg
- 2) afficher cette image
- 2) donner quelques caractéristiques générales de cette image
- 3) quel est le minimum dans l'image ? le maximum ? La moyenne des NG ?
- 4) donner son histogramme
- 5) donner le profil le long d'une ligne
- 6) quelle est la valeur d'intensité du pixel (20,20) ?
- 7) afficher uniquement l'image entre (250 et 300 ; 170 et 390)

- 8) ouvrir FeuilleHaricotVIS.jpg
- 9) donner ses caractéristiques et, si cela s'y prête, donner à voir les différents canaux qui la composent
- 10) la convertir en niveau de gris.

Choose your weapon ?



Let's focus on Fiji (imageJ)...

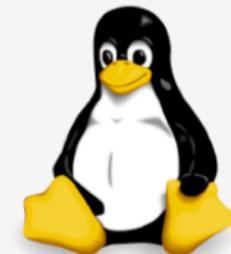


This is a read-only version of imagej.net, available during the transition to a new site.
Please direct any questions or issues to [this Image.sc Forum thread](#).
Thank you for your patience as we improve the website!

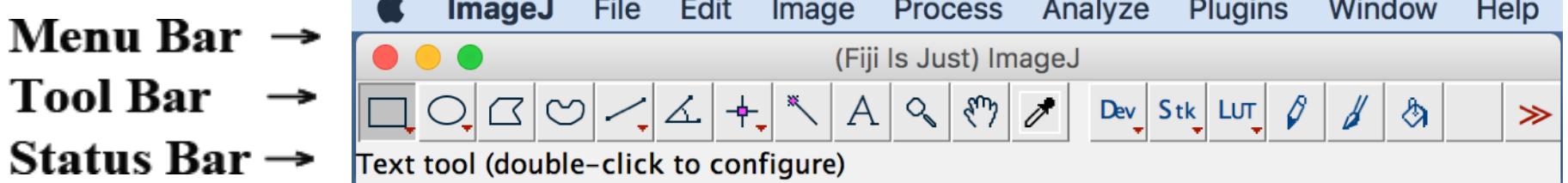
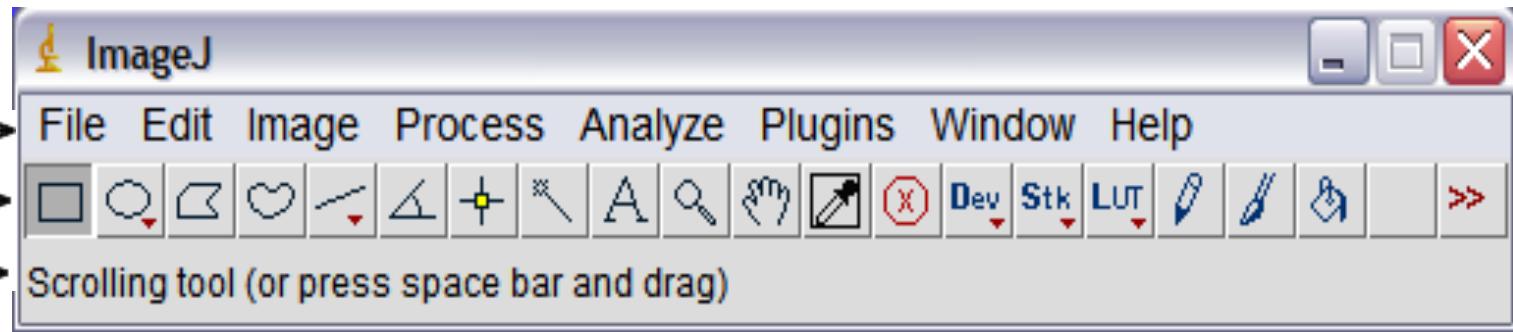
Fiji

[Fiji](#) is an image processing package—a "batteries-included" distribution of [ImageJ](#), bundling a lot of plugins which facilitate scientific image analysis.

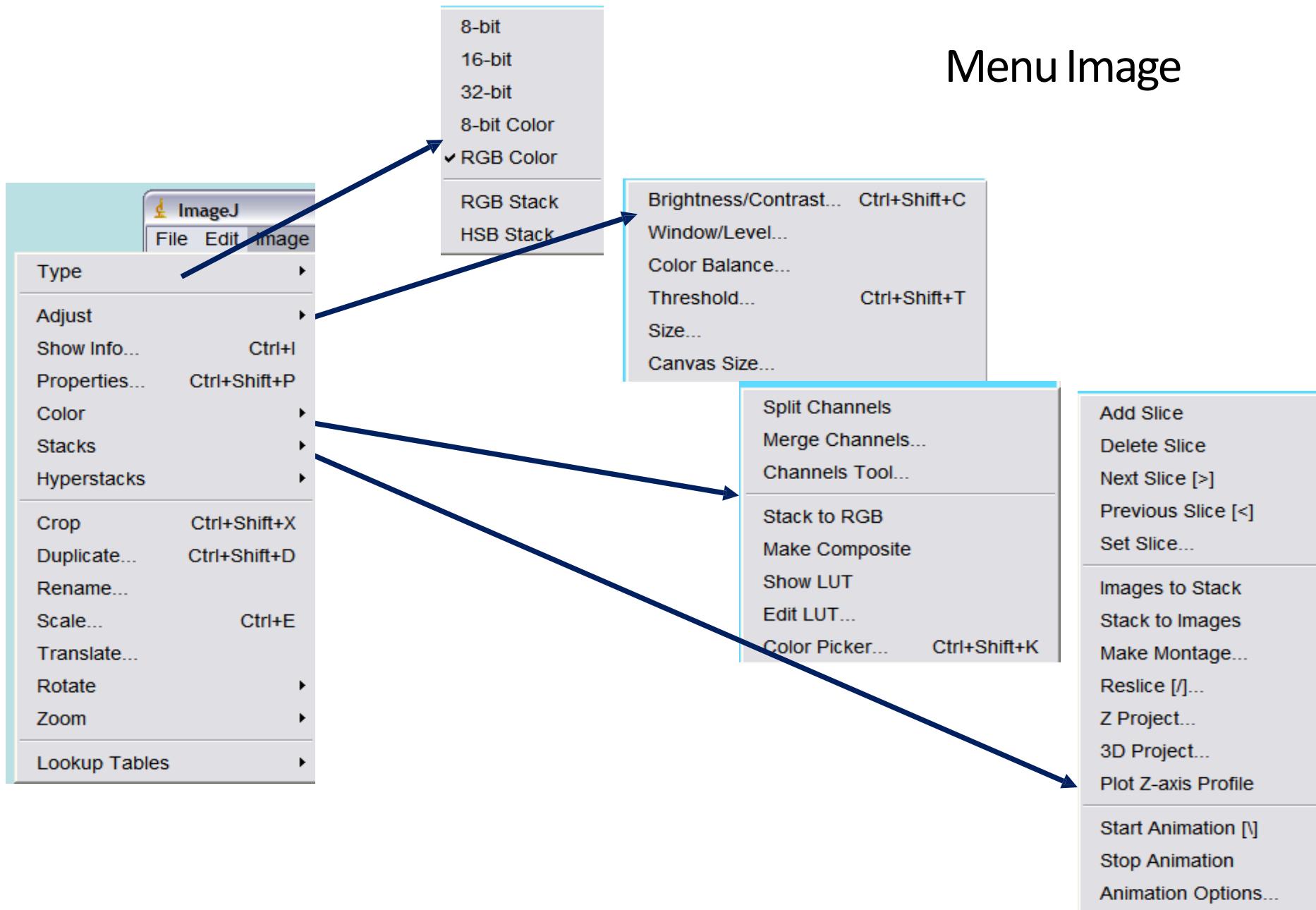
- **For users** - Fiji is [easy to install](#) and has an automatic update function, bundles a [lot of plugins](#) and offers comprehensive documentation.
- **For developers** - Fiji is an open source project hosted in a [Git](#) version control [repository](#), with access to the source code of all internals, libraries and plugins, and eases the [development](#) and [scripting](#) of plugins.

 64-bit	 macOS	 64-bit
Other downloads		
 32-bit	 No JRE	 32-bit

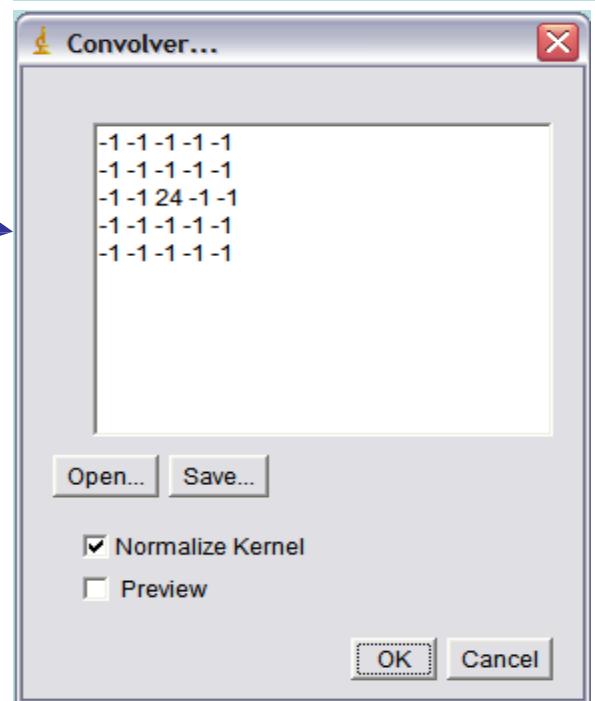
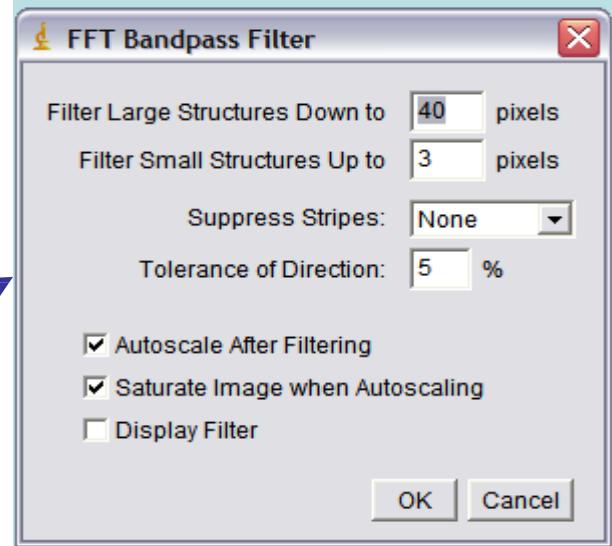
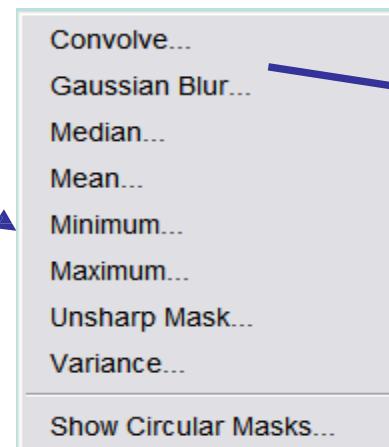
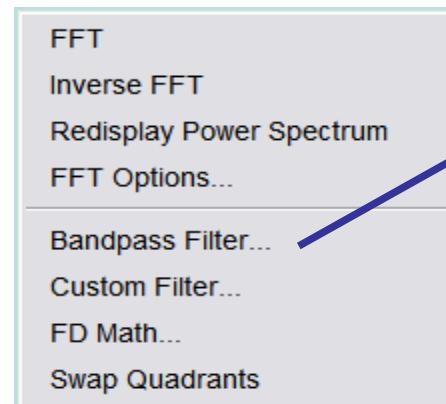
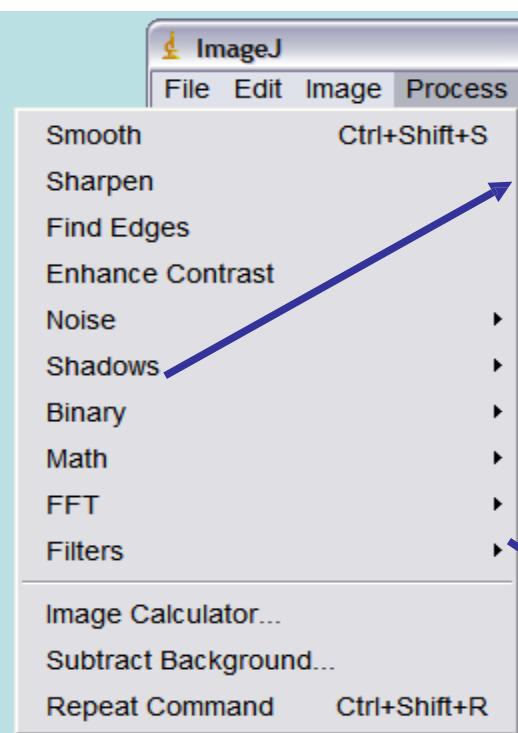
See the [Fiji Downloads](#) page for Life-Line versions, etc.

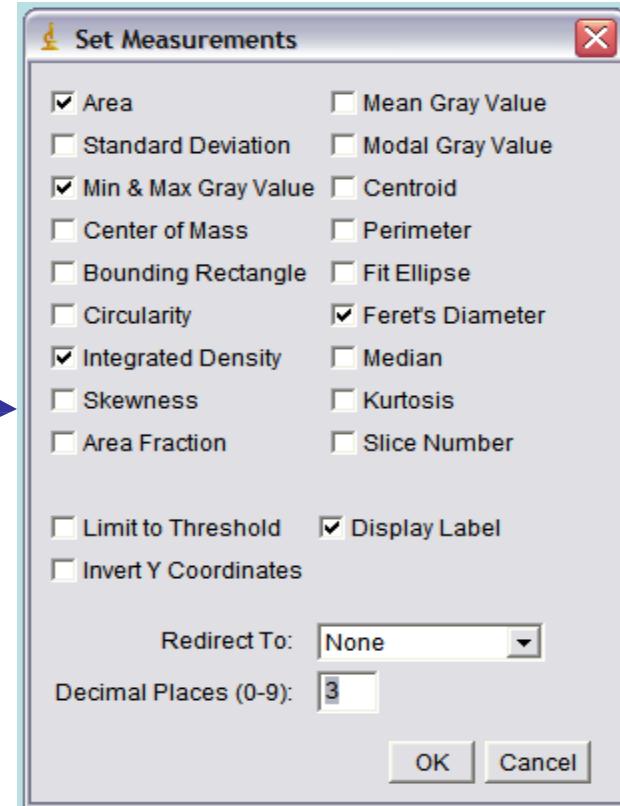
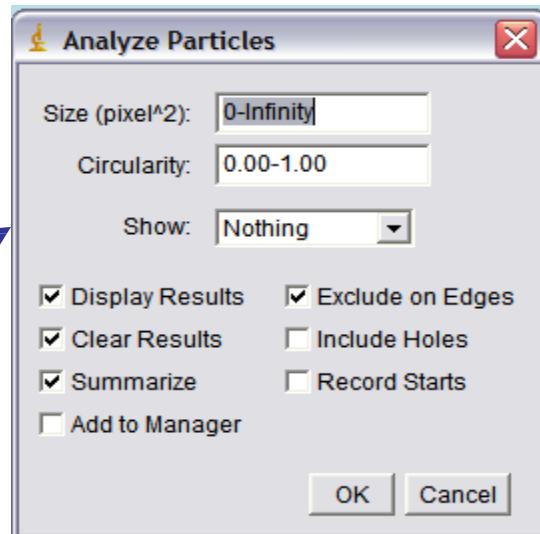
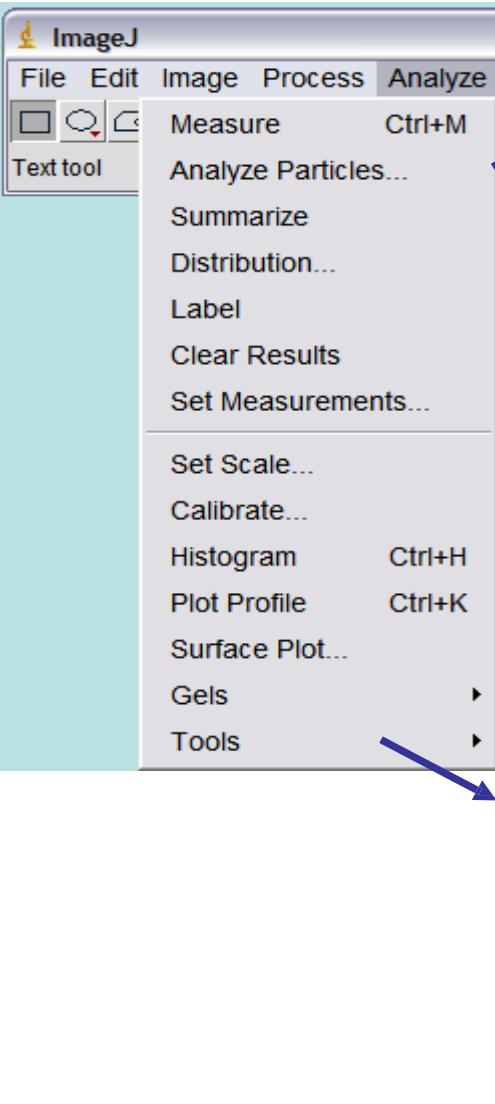


Menu Image



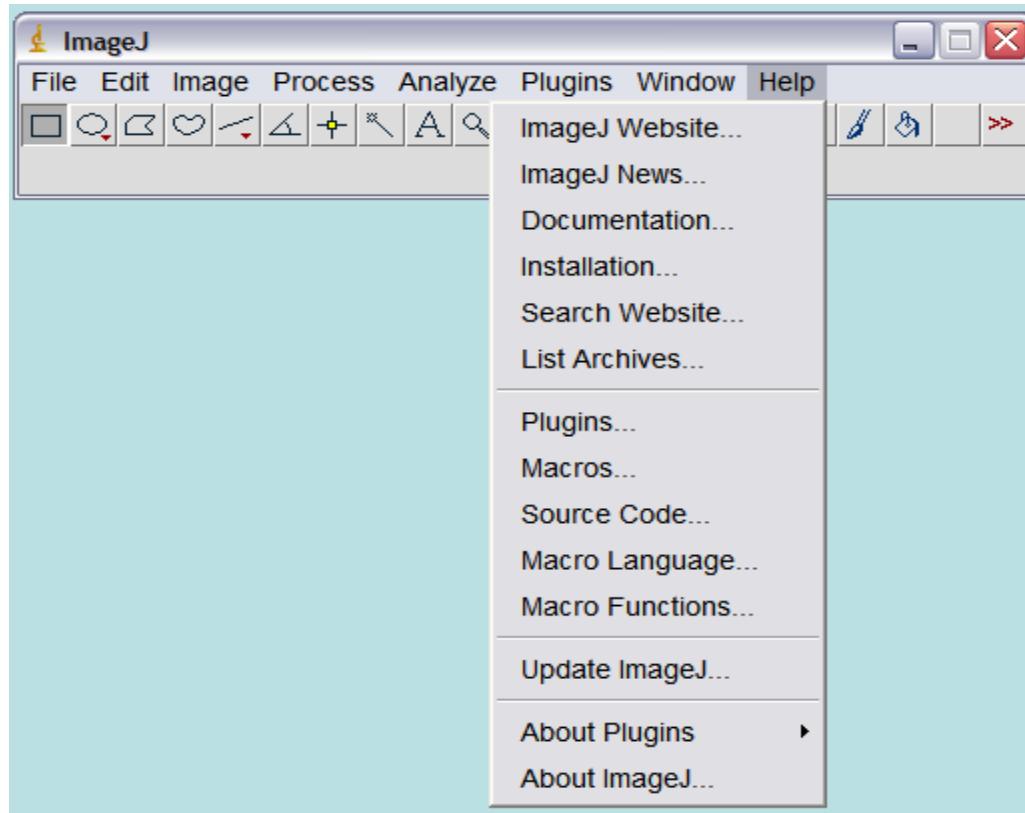
Menu Process





Menu Analyze

Menu help



Operations on images by transforming histogram

Transforming histogram → changing grey-level values of pixels

For each value of grey-level in initial image → new value in transformed operated image

Main transformations are:

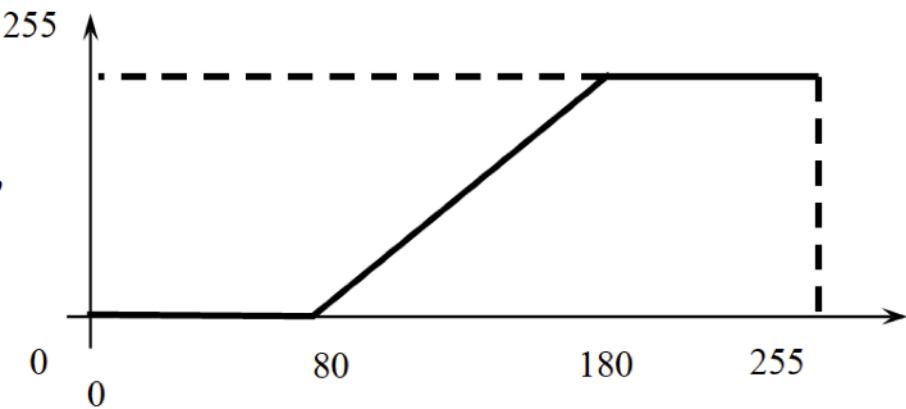
- Resizing dynamic
- Negative
- Equalization
- Exponential, logarithme,
- Thresholding

Transformation of grey-level : $v=f(u)$

u grey-level in initial image
 v grey-level in operated image

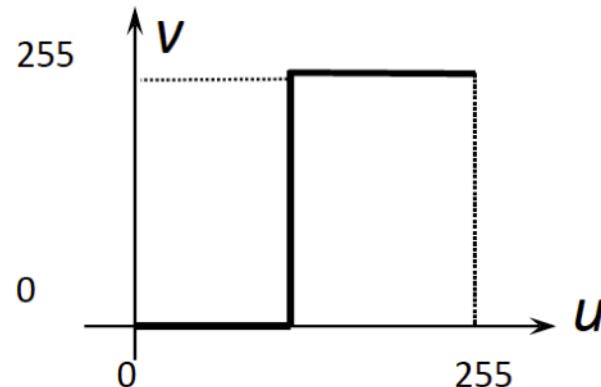
Example:

$$\left\{ \begin{array}{l} 0 < u < 80, \quad v = 0 \\ 80 < u < 180, \text{ relation linear} \\ 180 < u < 255, \quad v = 255 \end{array} \right.$$



Operations on images by transforming histogram

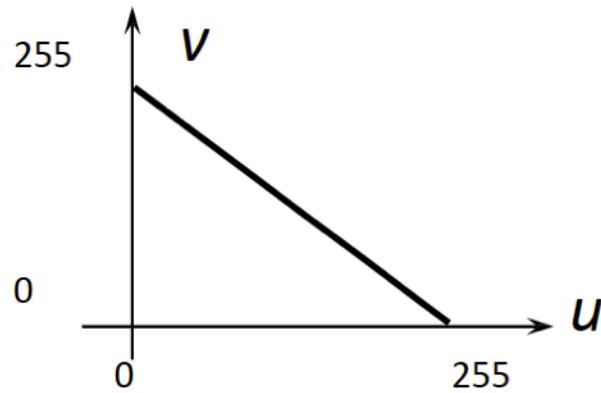
Thresholding (segmentation)



Negative



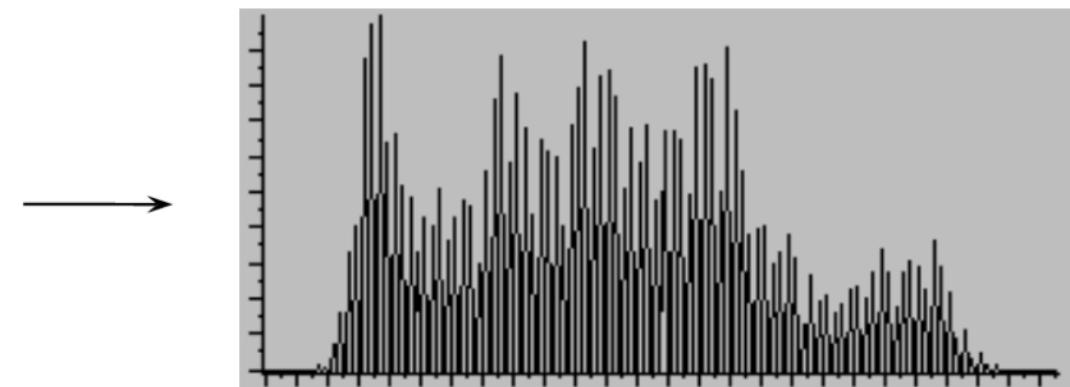
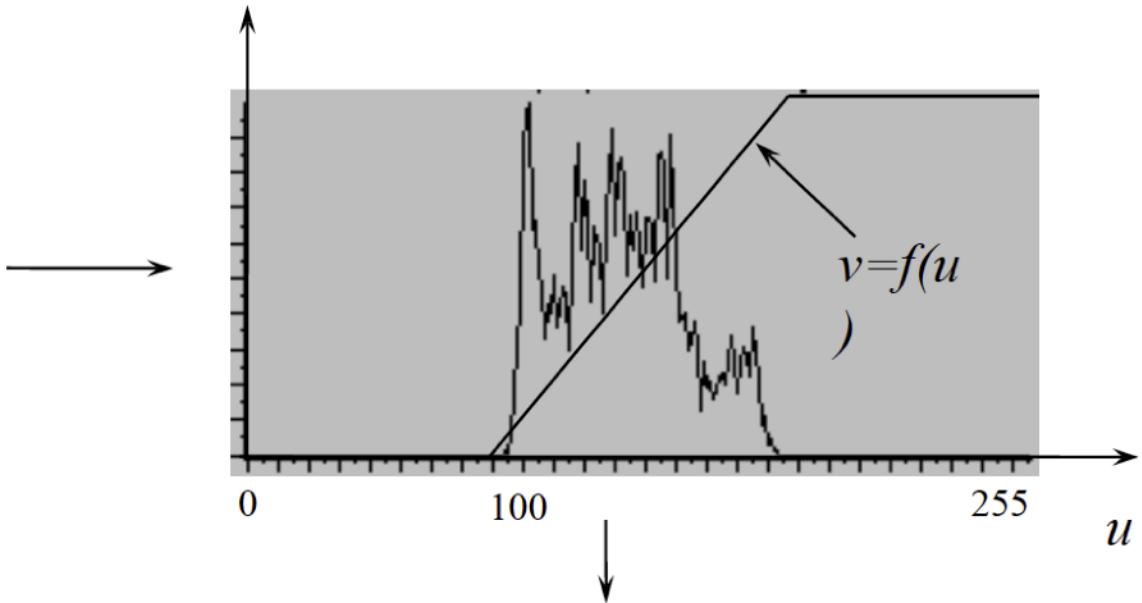
A



\bar{A}

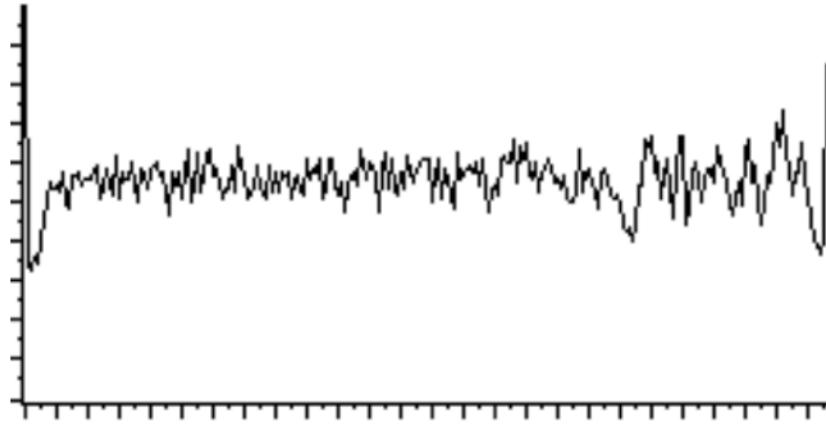
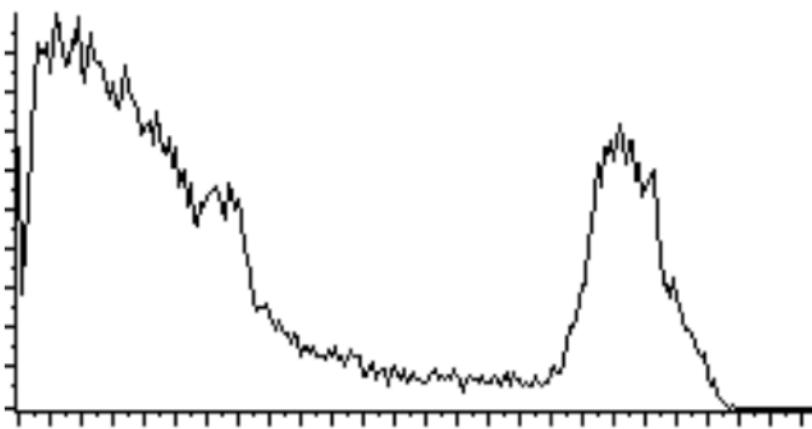
Operations on images by transforming histogram

Resizing dynamic



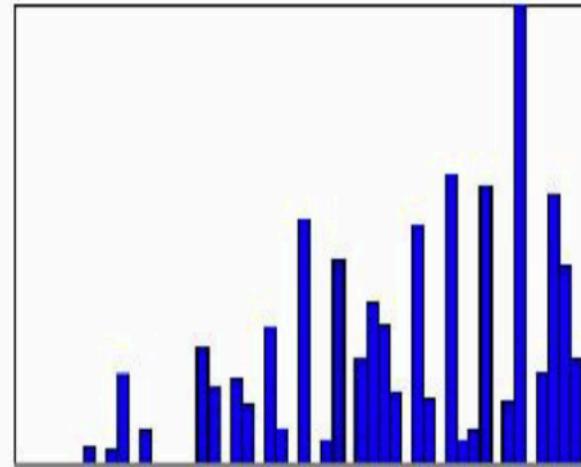
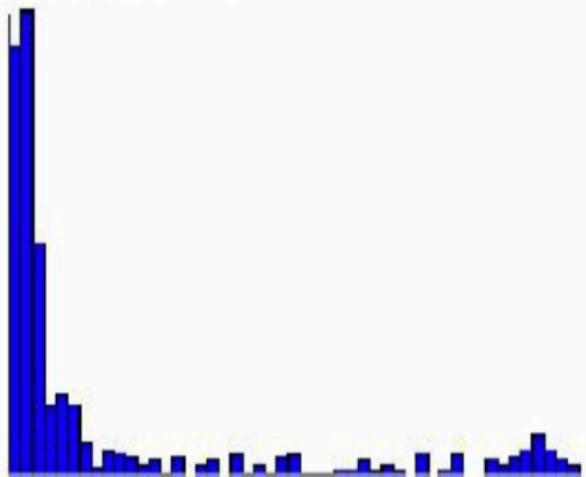
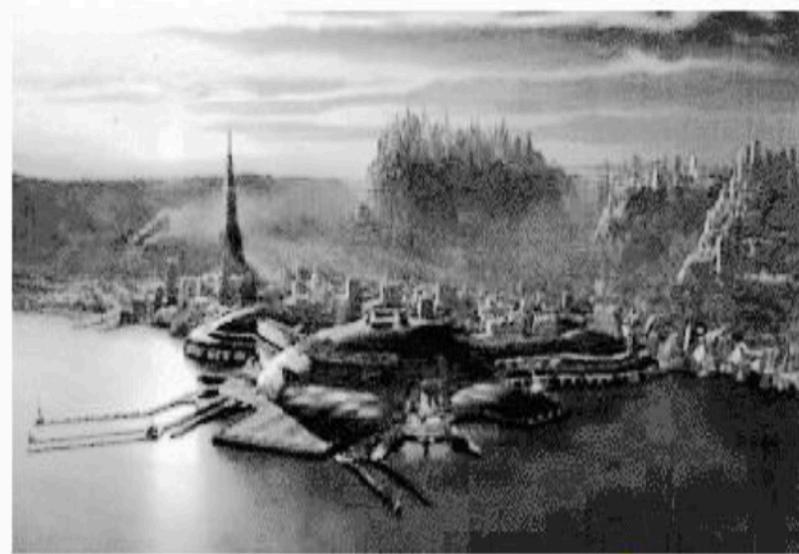
Operations on images by transforming histogram

Equalization (\rightarrow flattening the histogram)



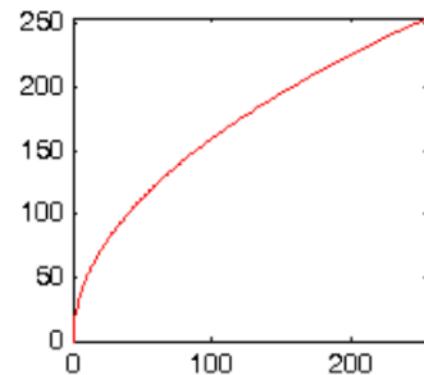
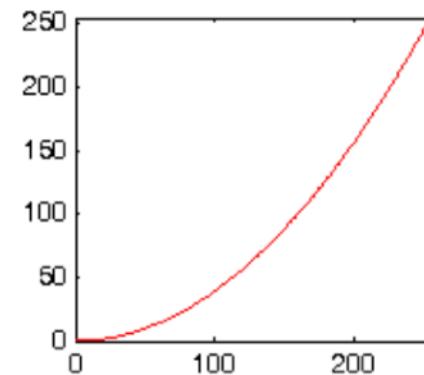
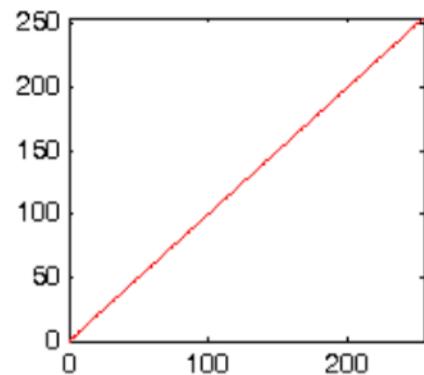
Operations on images by transforming histogram

Equalization (\rightarrow flattening the histogram)



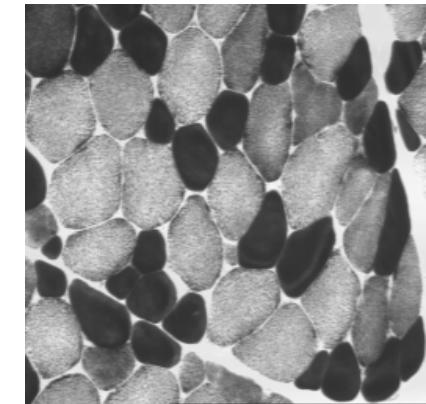
Operations on images by transforming histogram

Exp. Log.



HANDS ON n°2

- 1) load images clef.bmp & muscle.bmp
- 2) display their respective histograms
- 3) Can you explain the shape of histograms ?
- 4) Focussing on histogram of muscle.bmp, how can we decide which muscular fibers are denser (red (dark) and white (gray)) ?



HANDS ON n°3

Resizing dynamic

- 1) Load image quito.bmp.
- 2) Display histogram.
- 3) Resize the dynamic and display operated image

Equalization

- 1) Load image quito.bmp.
- 2) Display histogram.
- 3) Equalize histogram and display operated image
- 4) Repeat this operation. What happens ? Why ?



Operations on images with arithmetic calculus

Adding 2 images: $C = A + B$

- $f_C(i,j) = \{f_A(i,j) + f_B(i,j)\}/2$ Divided by 2 → intensity between 0-255
- Or $f_C(i,j) = \text{Min}\{f_A(i,j) + f_B(i,j) ; 255\}$

In order to

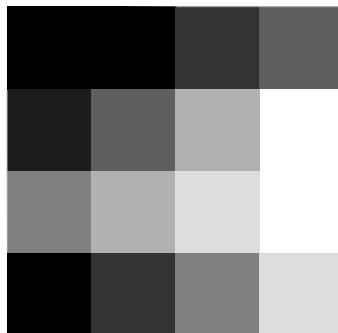
- Reduce noise in a stack of images
- Increase luminance



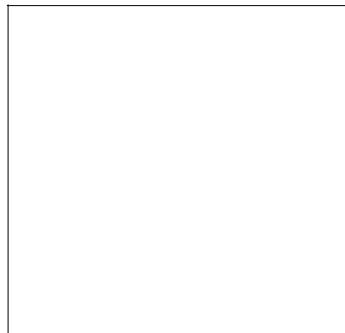
Operations on images with arithmetic calculus

Negative

A

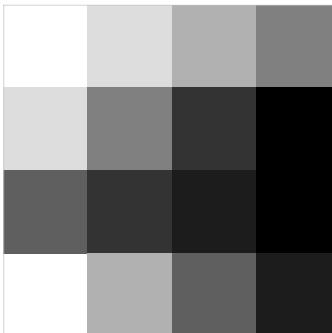


U_M



Calculate: U_M - A

$$= \underline{\underline{A}}$$



0	1	2	3
1	3	5	7
4	5	6	7
0	2	4	6

000	001	010	011
001	011	101	111
100	101	110	111
000	010	100	110

7	7	7	7
7	7	7	7
7	7	7	7
7	7	7	7

111	111	111	111
111	111	111	111
111	111	111	111
111	111	111	111

7	6	5	4
6	4	2	0
3	2	1	0
7	5	3	1

111	110	101	100
110	100	010	000
011	010	001	000
111	101	011	001

Operations on images with arithmetic calculus

Subtracking 2 images: $C = A - B$

- $C=A-B = \{A+(I-B)\}/2 = (A-B)/2 + I/2$
- $f_C(i,j) = \{f_A(i,j) - f_B(i,j)\}/2 + 128$
- $f_C(i,j) = \text{Max}(f_A(i,j)-f_B(i,j) ; 0)$

In order to

- Detect defaults
- Detect movements



$f(i,j)$



$g(i,j)$



$0,5\{f(i,j)+g(i,j)\}$

$g(i,j)-f(i,j)$

$f(i,j)-g(i,j)$



Calculation on images with logic operations

Binary images

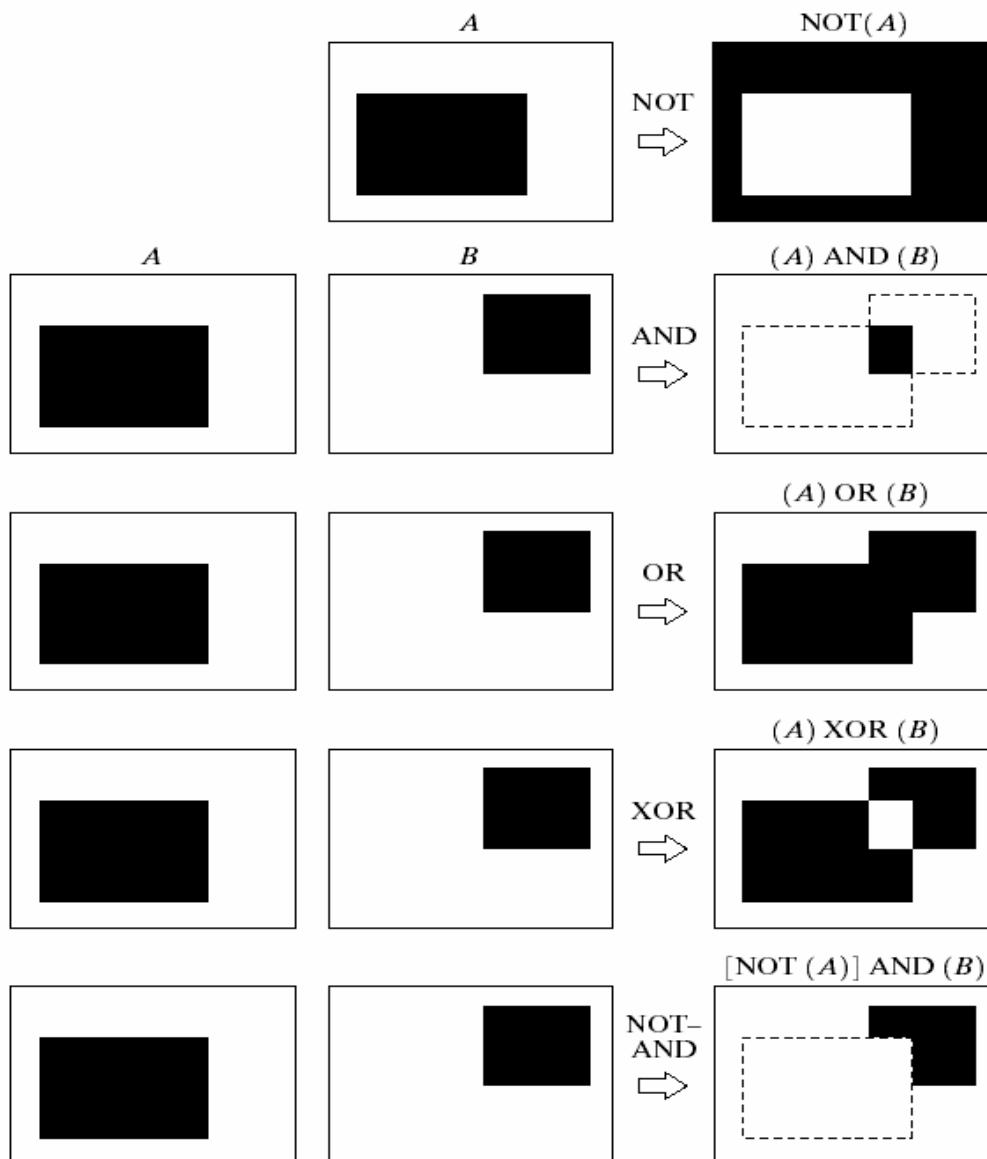
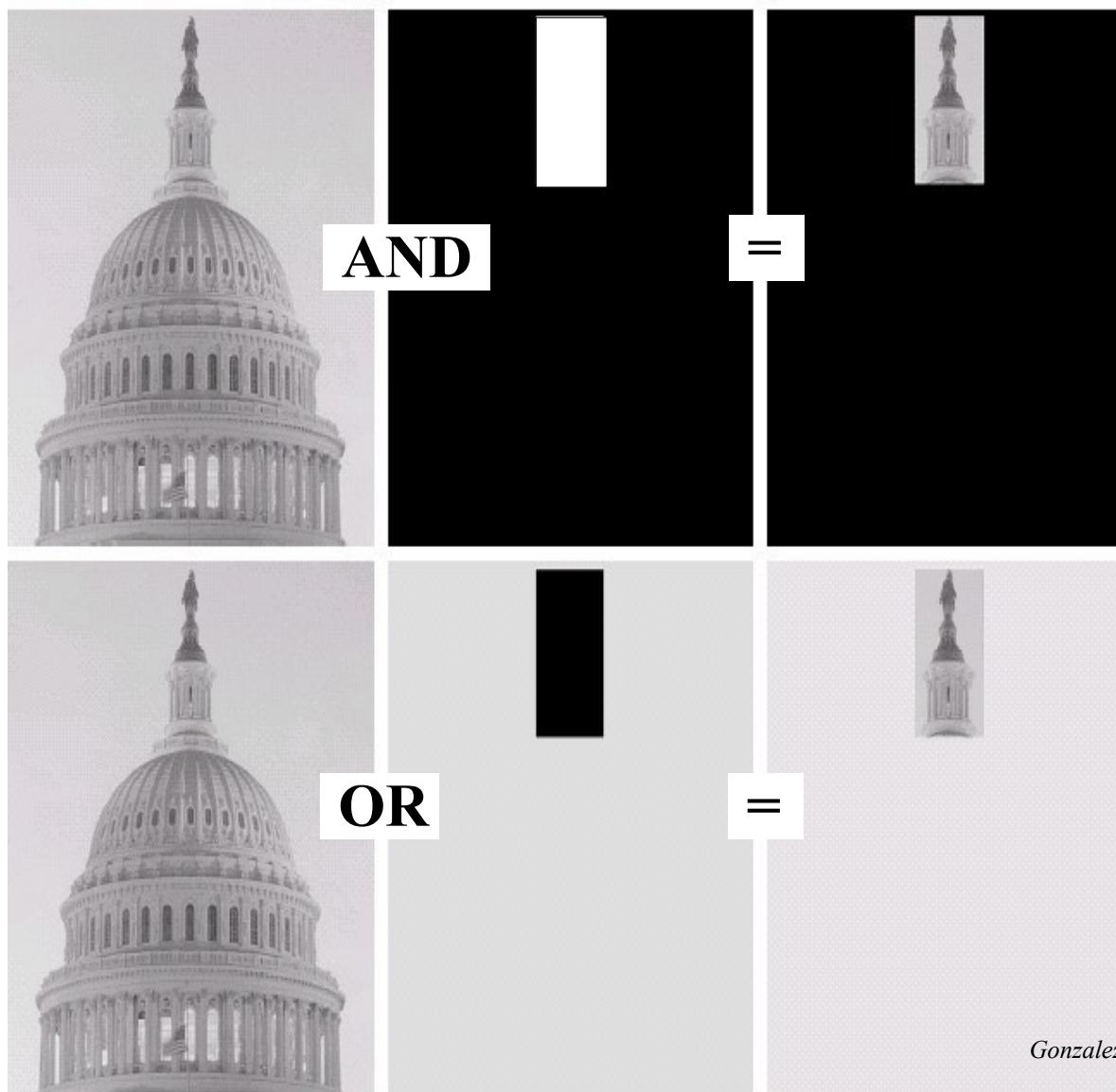


FIGURE 9.3 Some logic operations between binary images. Black represents binary 1s and white binary 0s in this example.

Calculation on images with logic operations



a	b	c
d	e	f

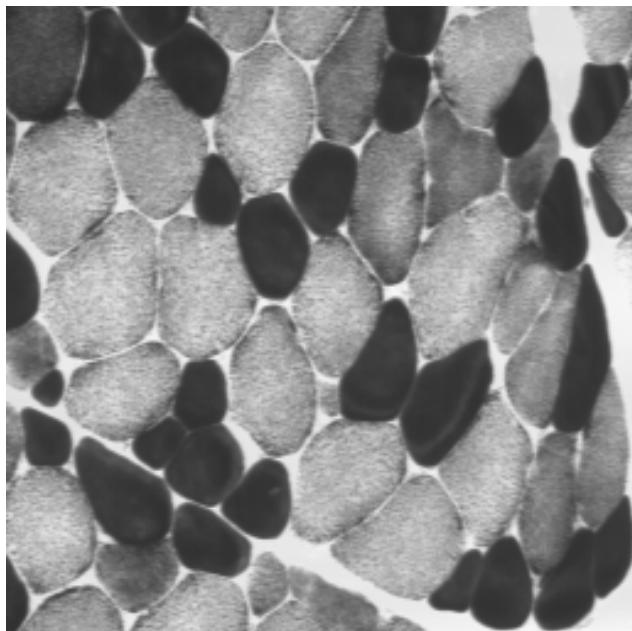
FIGURE 3.27

- (a) Original image. (b) AND image mask.
(c) Result of the AND operation on images (a) and (b). (d) Original image. (e) OR image mask.
(f) Result of operation OR on images (d) and (e).

→ Define/highlight R.O.I (Region Of Interest)

HANDS ON n°4

- 1) Load images muscle.bmp & spot.bmp
- 2) Observe gray-level on spot.bmp
- 3) Do operations OR, AND and XOR between the 2 images



Caractérisation fréquentielle des images

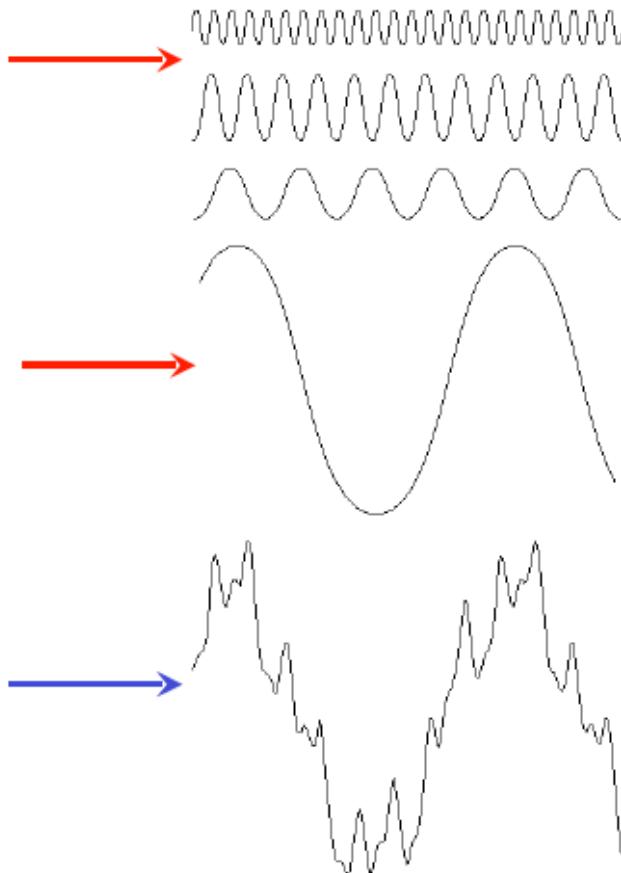
Notions de fréquences dans un signal

*Signal haute
fréquence*

...

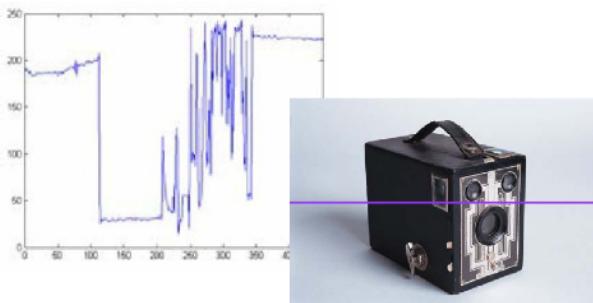
*Signal basse
fréquence*

*Ce signal représente
la somme des
quatre signaux ci-
dessus*

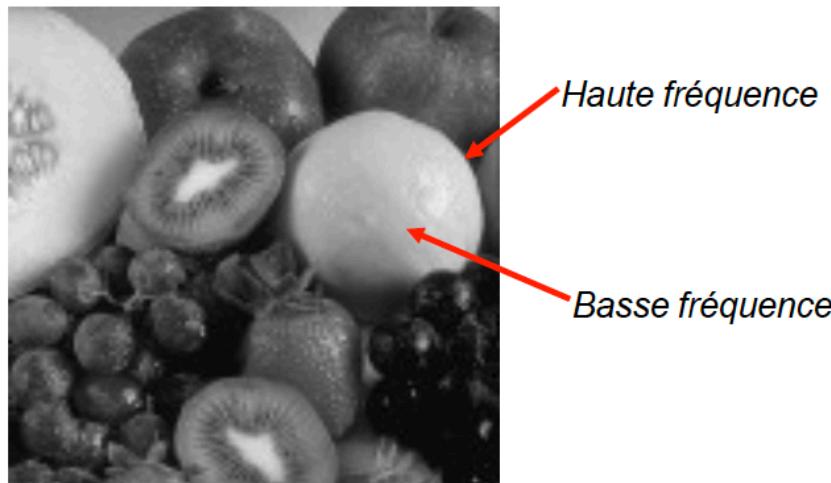


Source : Gonzalez and Woods. Digital Image Processing. Prentice-Hall, 2002.

C'est quoi les fréquences dans l'image ?



- Fréquence = changement d'intensité
- Basses fréquences : régions homogènes, flou
- Hautes fréquences : contours, changement brusque d'intensité, bruit



La plus grande partie de l'énergie d'une image se situe dans les basses fréquences.

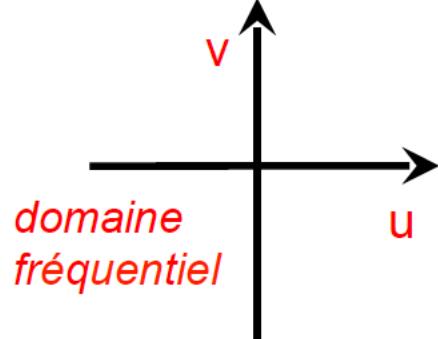
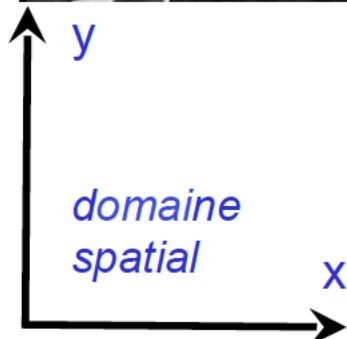
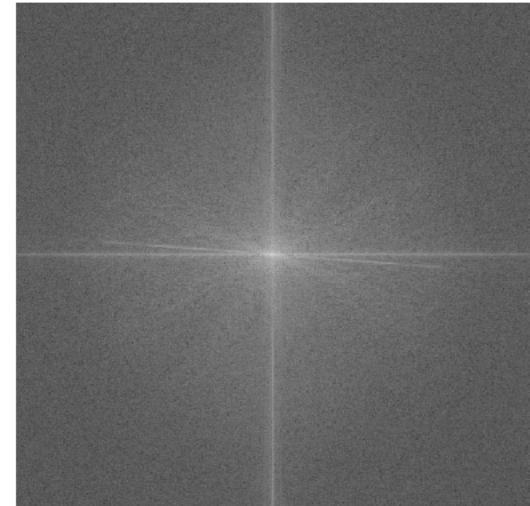
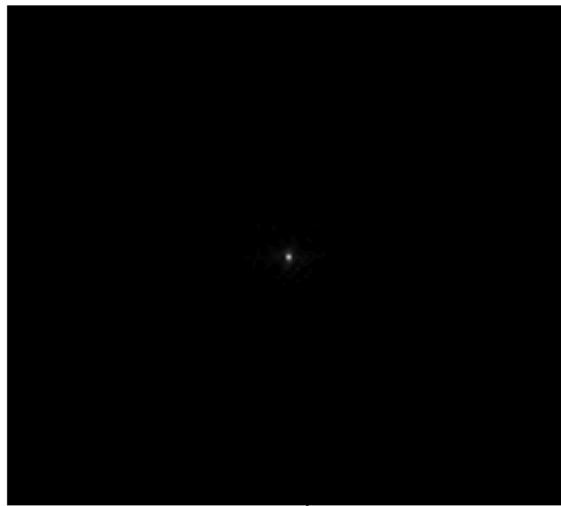
Transformée de Fourier (TF)

Implémentation algorithmique : Fast Fourier Transform

Image originale

Spectre de Fourier
 $|F(u,v)|$

Spectre rehaussée
 $\log(1 + |F(u,v)|)$



Transformée de Fourier (TF)

Implémentation algorithmique : Fast Fourier Transform

$$F(f_x, f_y) = \iint f(x, y) \exp(-2j\pi(f_x x + f_y y)) dx dy$$



Figure 4.24a Original jet fighter image corrupted with a diagonal periodic noise pattern.

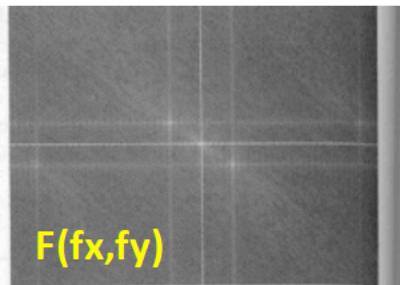


Figure 4.24b Fourier-transform image.

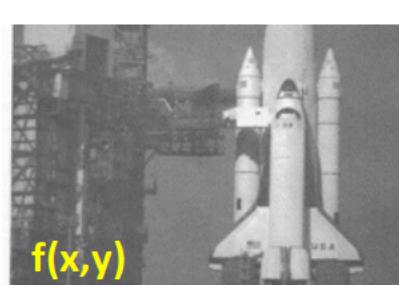


Figure 4.25a Original Space Shuttle image.

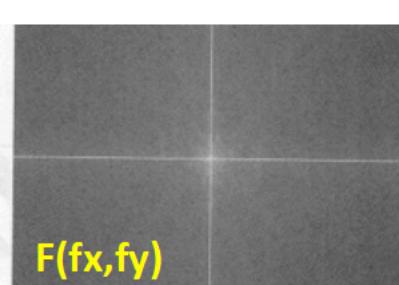


Figure 4.25b Fourier-transform image.

Hautes fréquences : loin du centre de la TF

Basses fréquences : proche du centre de la TF

Composante continue (DC) : centre de l'image
moyenne de l'image

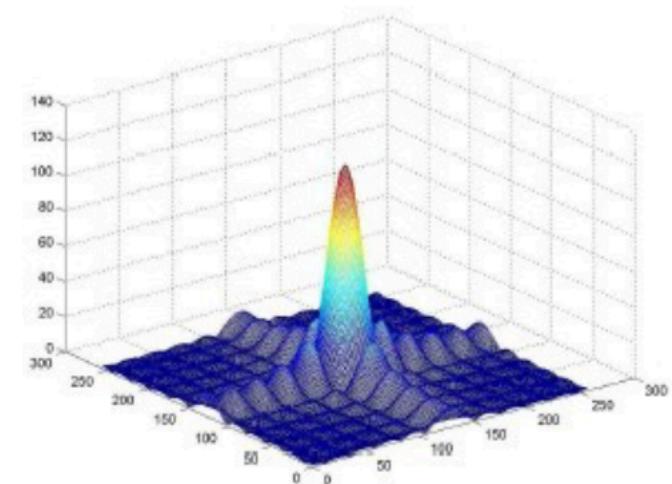
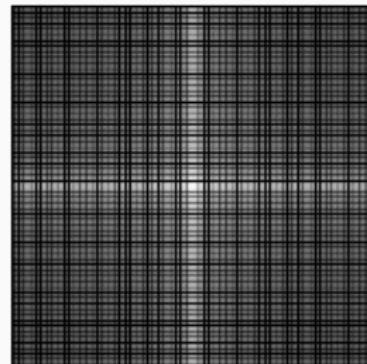
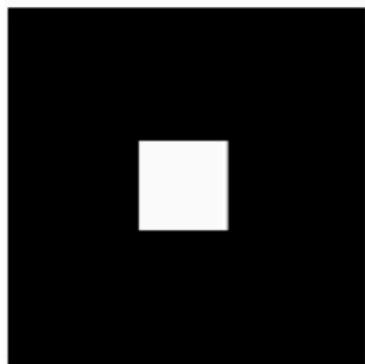
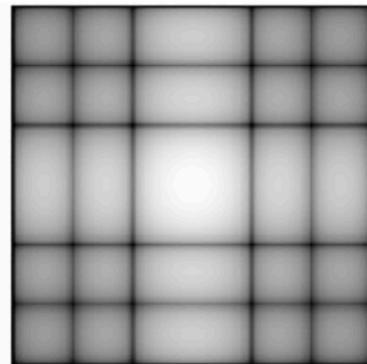
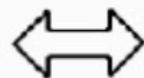
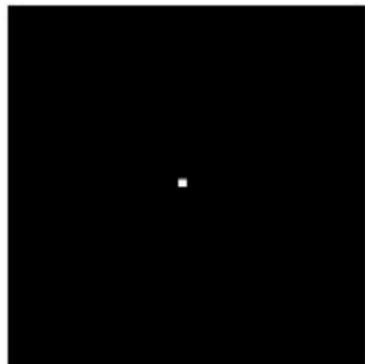
La TF d'une fonction réelle donne une fonction complexe

Entrée : une image

Sortie : image(partie réelle) +
image(partie imaginaire)

On utilise beaucoup la norme de la TF !

Exemples de TF



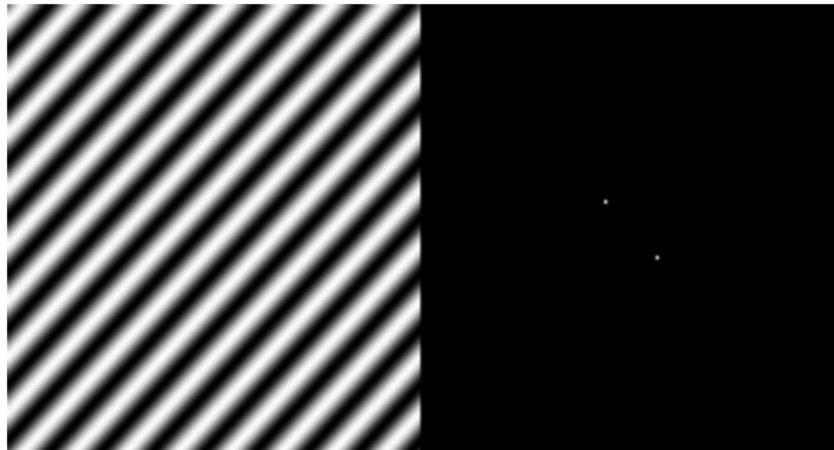
Images originales

TF

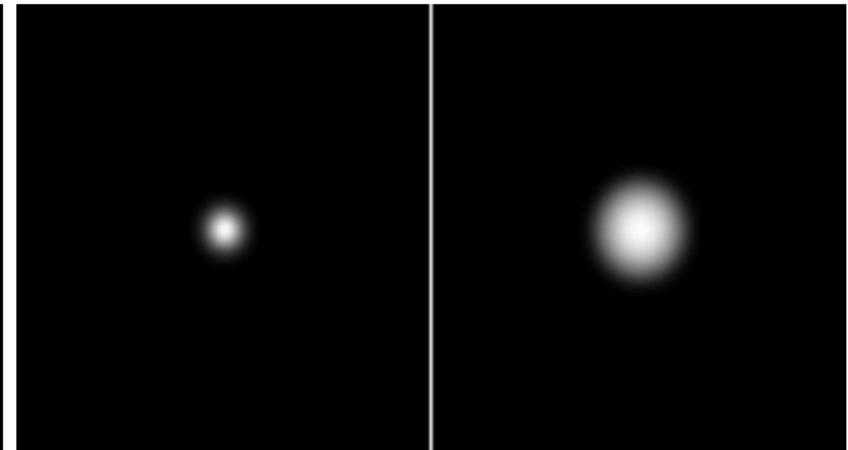
Vue 3D de la TF

Source : Thomas Guyet. *Images numériques. IUT Sérécom (France).*

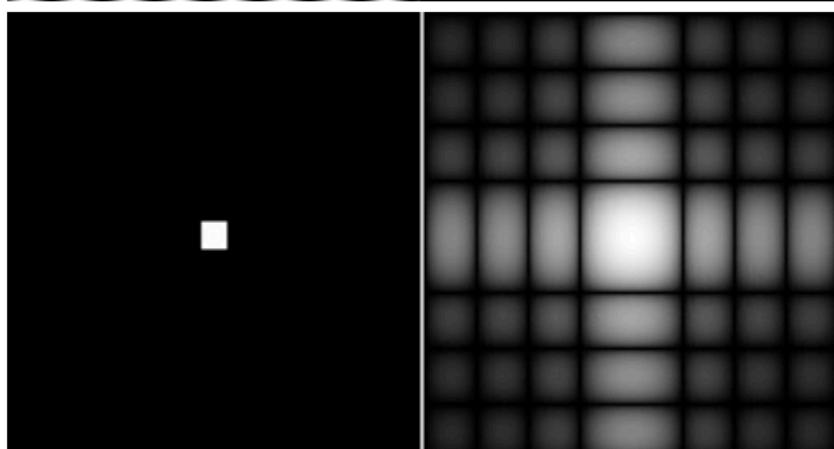
Sinus



Gaussienne



Carré

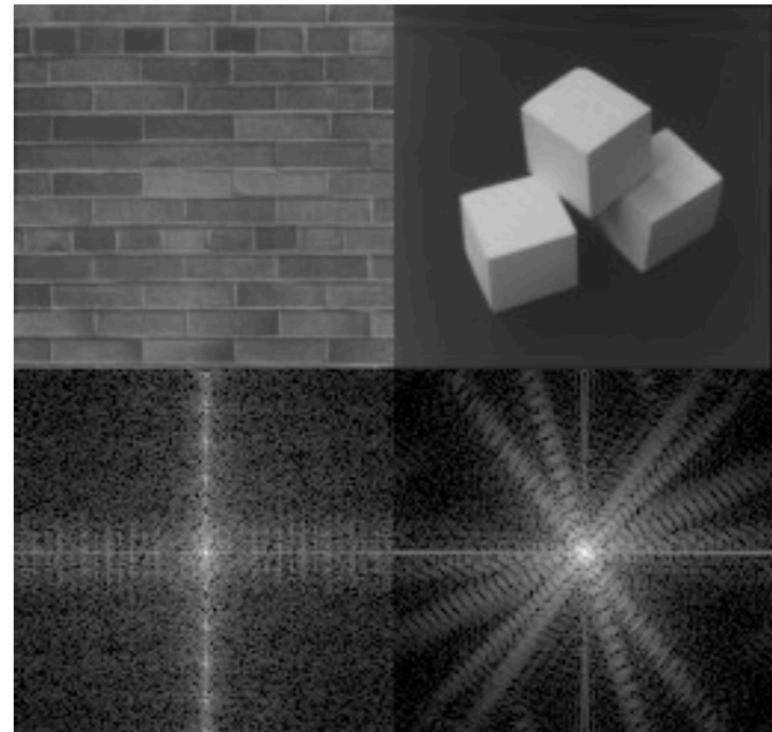
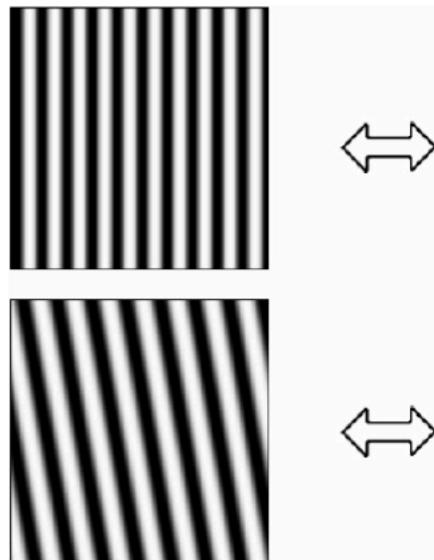


Impulsions

CVIPTools : Utilities>Create>{Rectangle, Sine Wave, Square Wave}

L'image de gauche possède des lignes horizontales/verticales qu'on retrouve dans sa transformée

L'image de droite possède des lignes dans toutes les directions qu'on retrouve aussi dans sa transformée



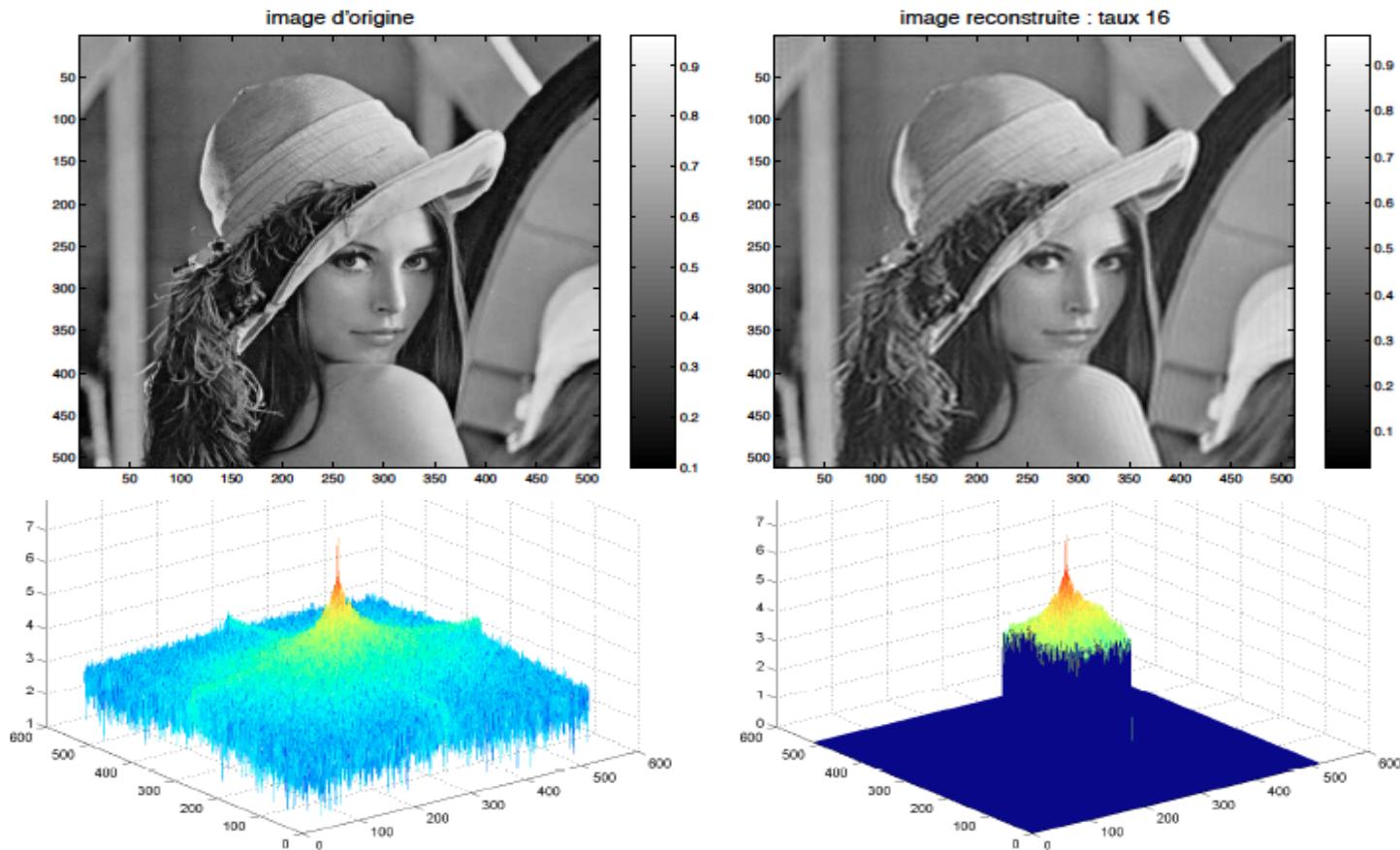
Rotation
d'images

→ rotation de la
TF (même angle)

Utilisations

Les applications qui en découlent sont très nombreuses. Par exemple :

- Caractérisation / traitement des images dans le domaine fréquentiel,
- Filtrage / réduction du bruit,
- Compression,
- ...



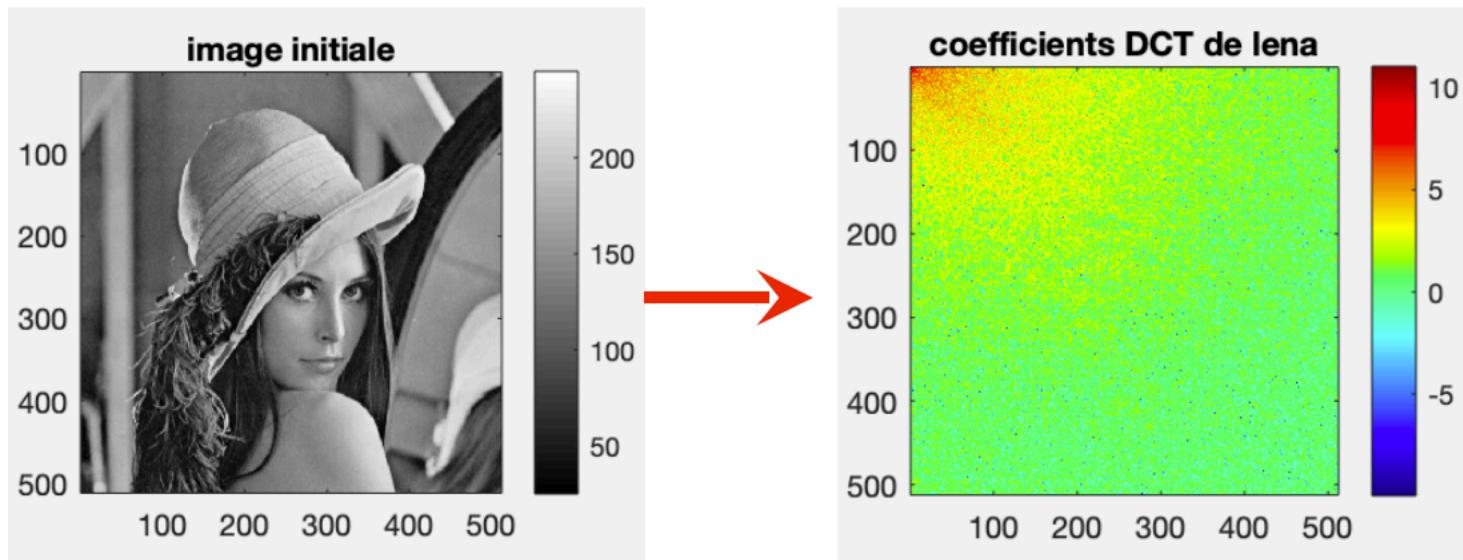
Le principe de base exploité dans le standard de compression jpeg.

Transformée en Cosinus

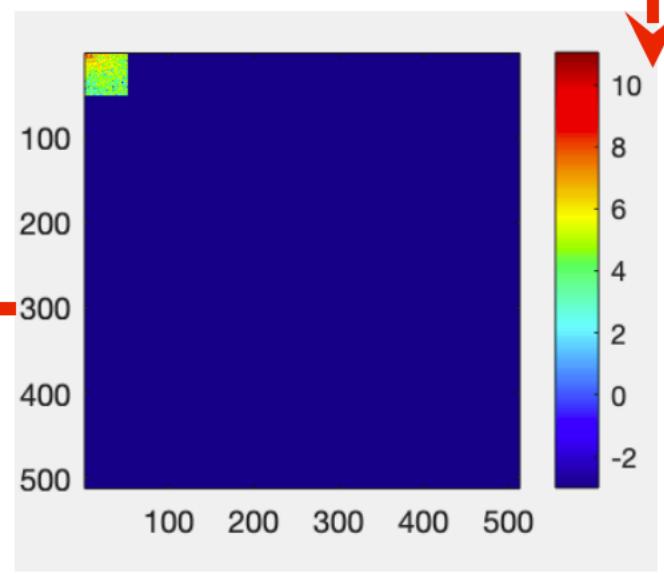
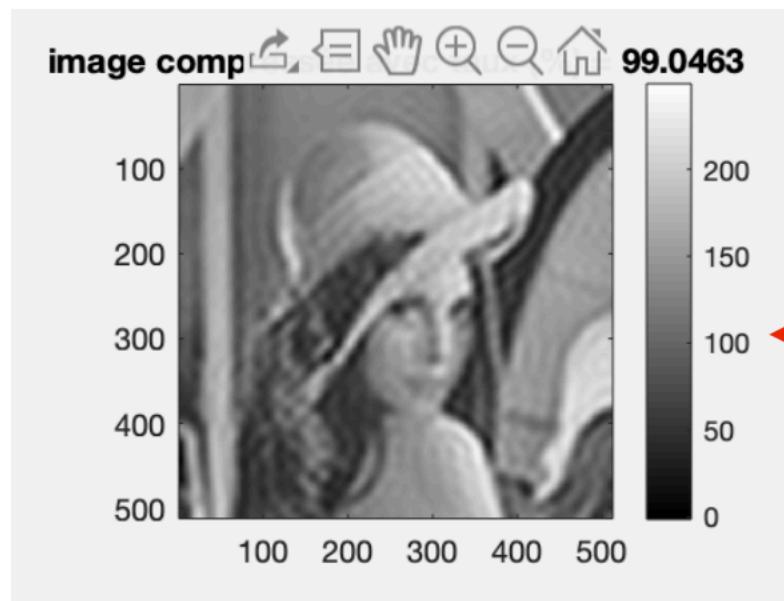
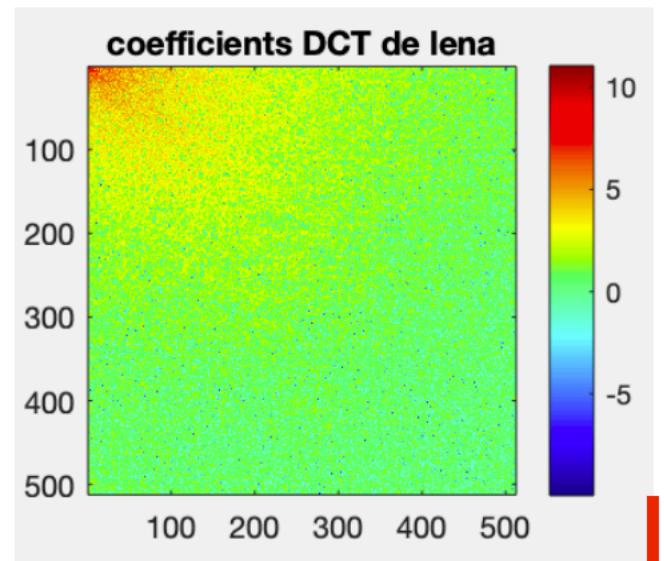
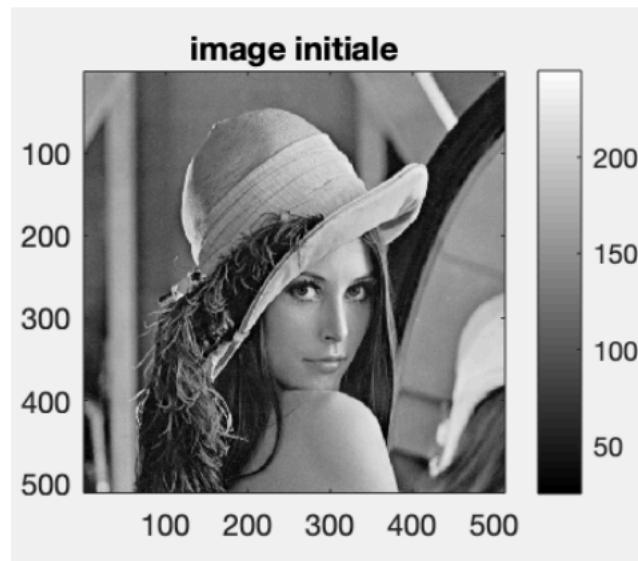
Implémentation algorithmique : Discret Cosinus Transform

Proche de la TF

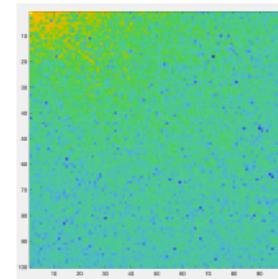
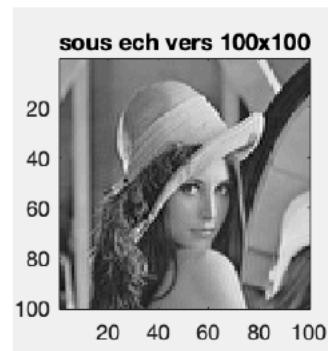
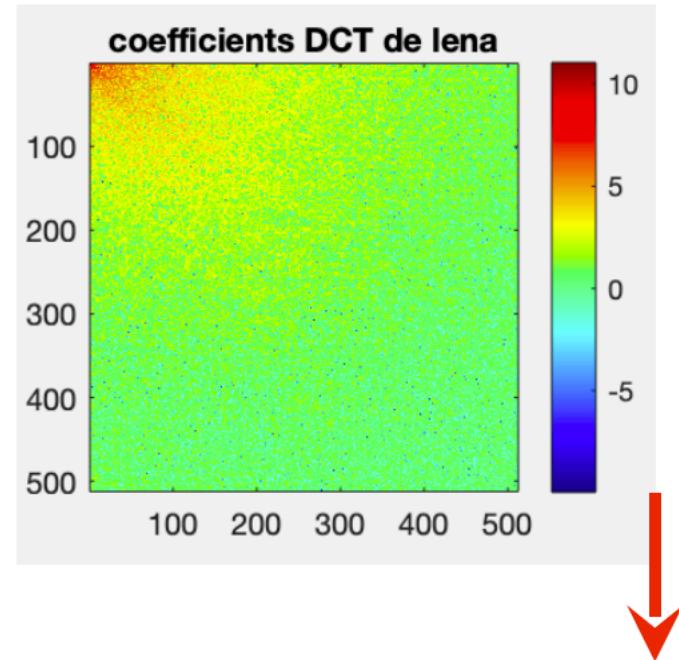
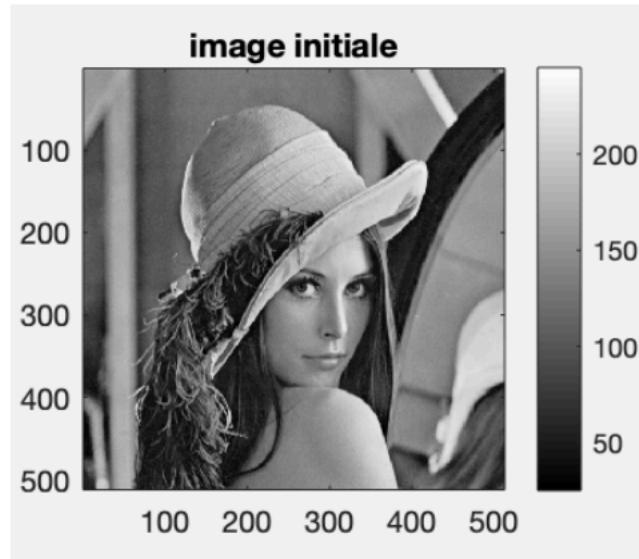
- deux fois plus de coefficients qu'avec la TF, mais tous sont réels
- utile pour la compression, du sous-échantillonnage ou de l'interpolation



Compression

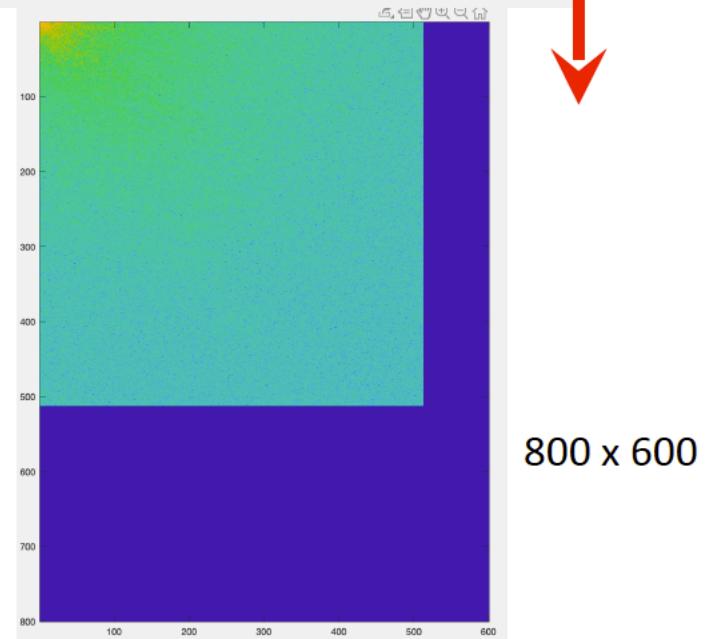
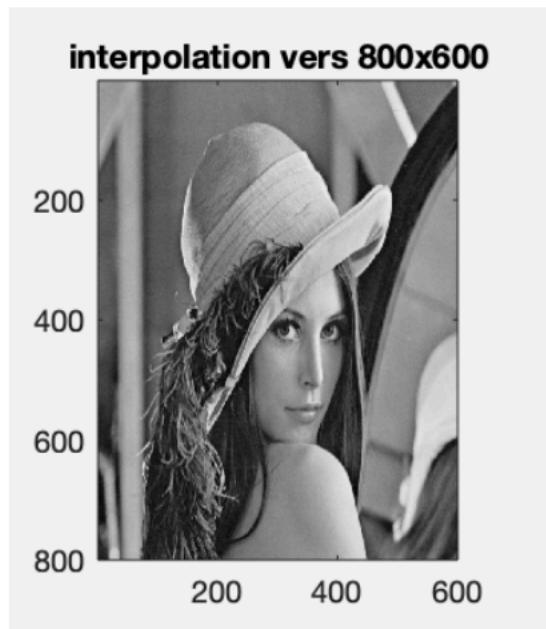
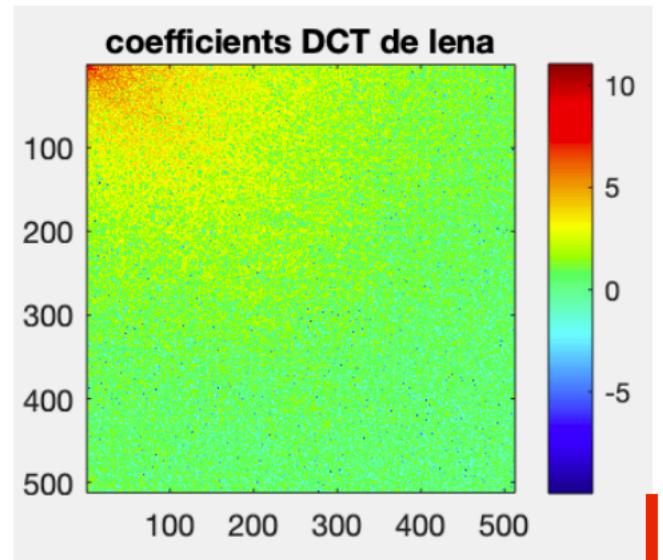
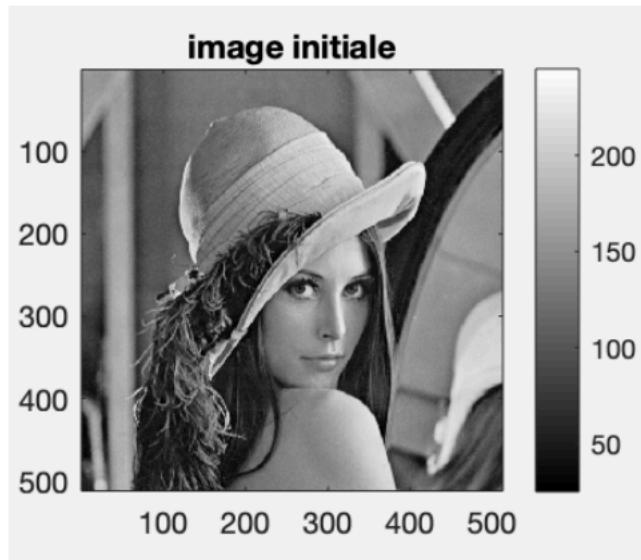


Sous-échantillonnage



100 x 100

Interpolation



HANDS ON n°5

Exercice 1 :

1. Ouvrir image Lena.jpg
2. A l'aide de fft2, calculer la FFT et observer (afficher) son spectre d'amplitude. Il se peut qu'il y ait des choses bizarres....
3. Multiplier par un masque binaire 100x100 qui préserve les basses fréquences.
4. Faire la FFT Inverse (ifft2) pour obtenir l'image ainsi filtrée.
5. Afficher image compressée obtenue.

Exercice 2 (plutôt Octave ou Matlab)

Ouvrir Lena.jpg

Vérifier que c'est 512x512 et si nécessaire convertir en NG.

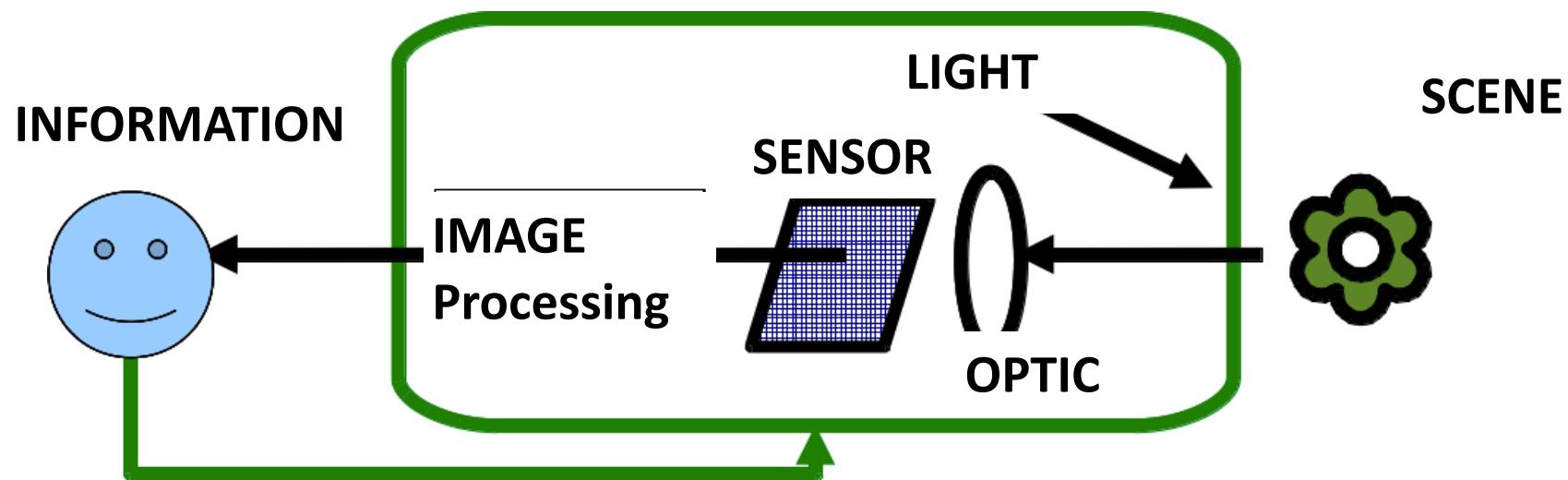
Calculer la DCT2 de l'image et afficher le spectre image.

A l'aide de la DCT (dct2 depuis le package signal **pkg load signal sous Octave**), faire :

- une compression en ne conservant que les 50x50 premiers coefficients.
- un sous échantillonnage à 100x100
- interpolation à 600x800

Chapitre 3 : Pré-traitements (bas-niveau)

What is noise in images ?



- We acquire images to extract the information they carry
- We call **noise** whatever is degrading the performances of the information task that motivated the image acquisition.
- An informational definition distinct from physics
- The level of noise depends on the informational tasks

Due to...

Light:

- Non uniformity → **slow variations of intensity**
- Shadows on objects sur l'objet → **fast variations of intensit**

Light properties:

- Light flow is not continuous (particles – photons)
- Poisson noise when weak flow of light

Objects can move. If motion time > integrated time of the camera → blurred image

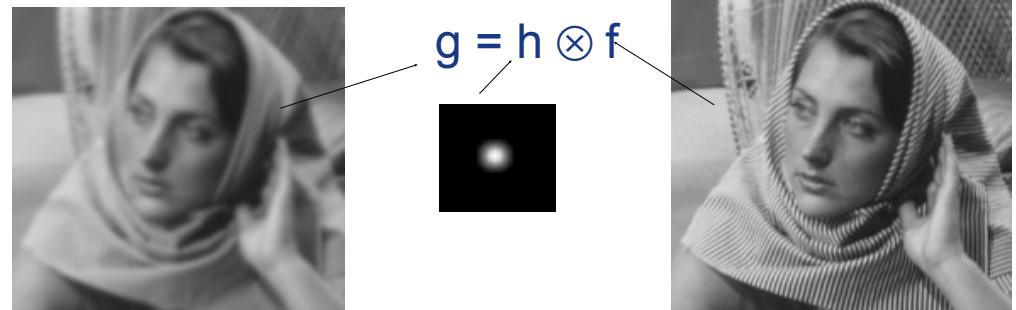


Optic:

Lens have aberrations

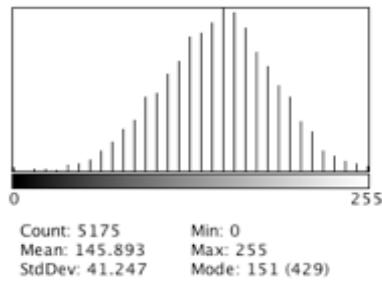
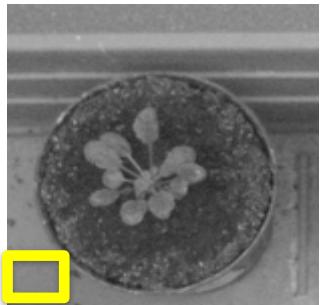
Optical filtering of fine details

→ blurred image



Due to...

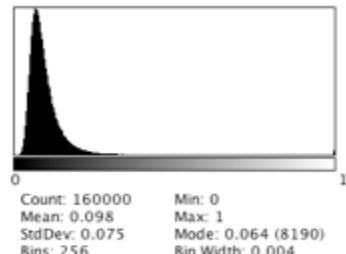
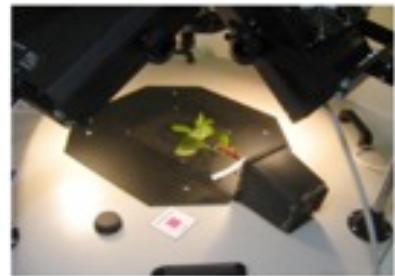
Sensor:



Thermal noise: independant and equally distributed according to Gaussian law



Impulse noise eq. to salt & pepper
Some pixels are not defined



Combination of image with thermal noise
(Ex.: $Q_{\text{max}} = (F_o - F_m) / F_m \rightarrow$ Cauchy noise)



Quantification:

Noise with non-uniform distribution

8 bits (256 levels) 4 bits (16 levels) 2 bits (4 levels)

Now let's try to denoise !

Let's assume we have a thermal noise, i.e. Gaussian noise.

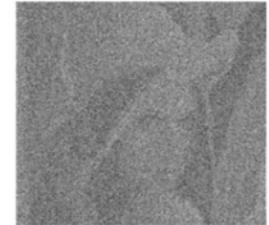
Optimal statistic (in the sense of the mean squares) is averaging.

A panel of techniques:

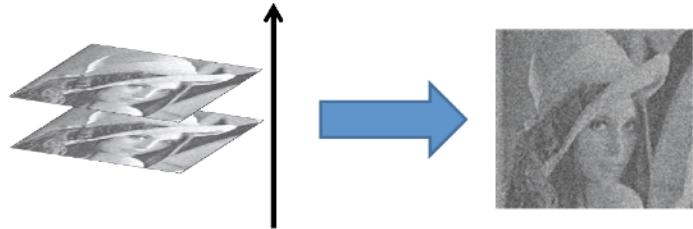
- Filtering by temporal averaging
- Filtering by spatial linear averaging (convolution, Fourier)
- Filtering by spatial nonlinear processing
- And so on: anisotropic filtering, non local mean, mean Shift, rolling ball, top hat, bilateral filtering,...

Temporal denoising

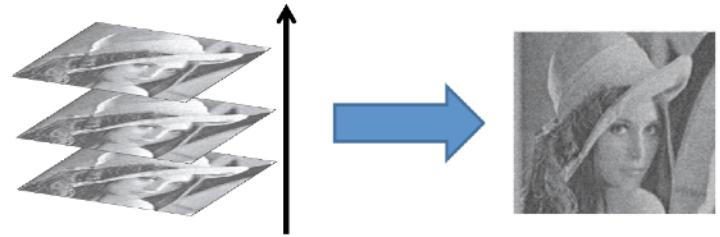
- We assume:
 - A static scene
 - Possibility to acquire multiple images at different times



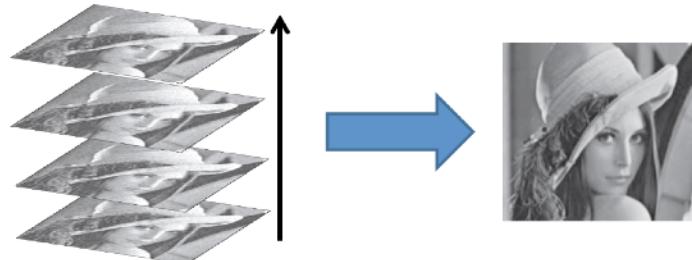
N=2 images



N=3 images



N=4 images



Spatial denoising

- If temporal denoising non accessible, we assume objects in images are piecewise constant
- Neighbors pixels are highly correlated
- Possibility to denoise locally

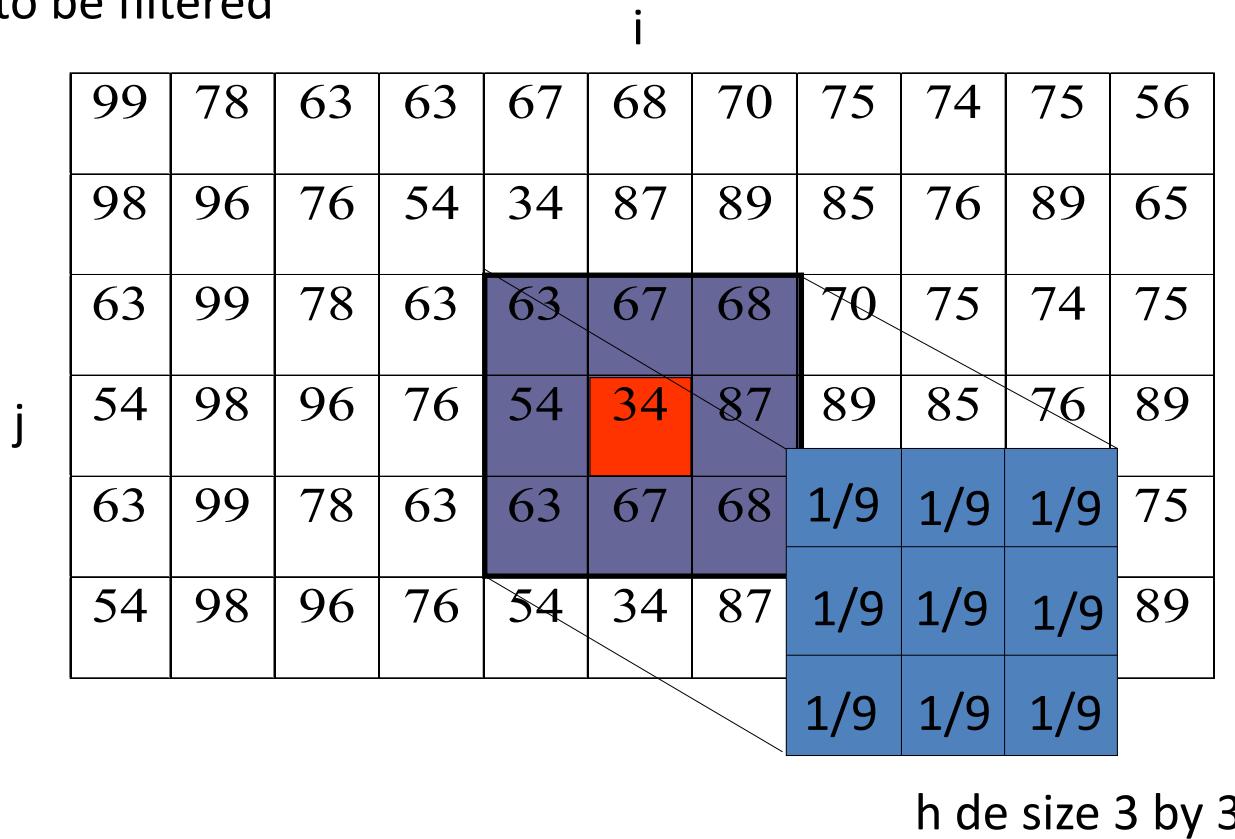


Convolution

$$g(i, j) = \sum_{(k,l) \in W} h(k, l) f(i+k, j-l)$$

- f image to denoise, g denoised image
- h coefficients of the filter, non zero only on the neighborhood W
- A pixel $f(i,j)$ is replaced by its weighted sum on the neighborhood
- This is realized on the entire image

f image to be filtered



Compute $g(i,j)$ the denoised version of pixel $f(i,j)$

$$g(i,j) = (63 \times 1 + 67 \times 1 + 68 \times 1 + 54 \times 1 + 34 \times 1 + 87 \times 1 + 63 \times 1 + 67 \times 1 + 67 \times 1) / 9 = ??$$

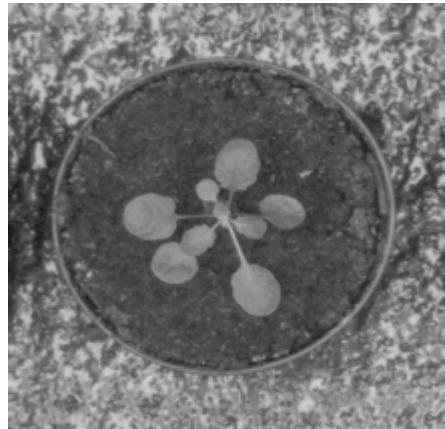
NB : for pixels edging images that can be calculated :

- Keep initial value
- Give value of the closest neigboored pixel calculated

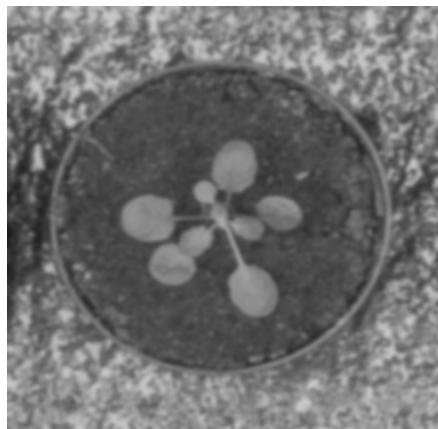
In ImageJ : Process->Convolve

Influence of the kernel size W of filter h

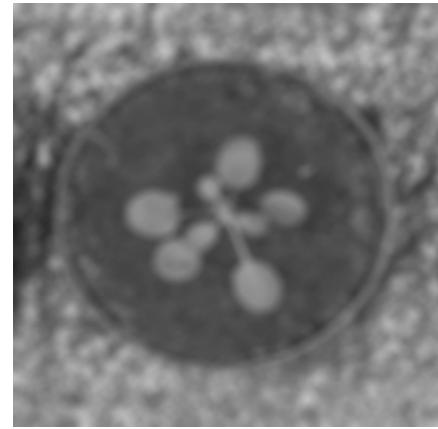
Original 530*565



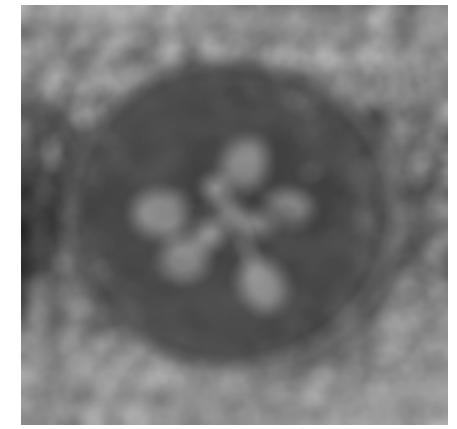
$W=3*3$



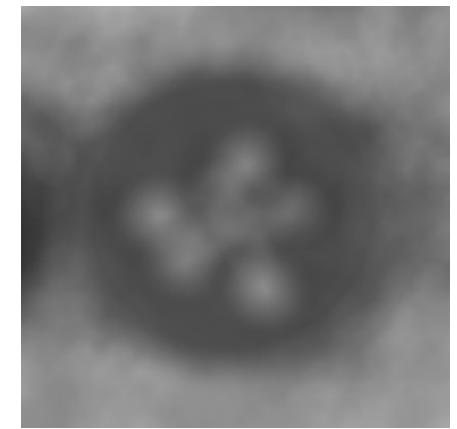
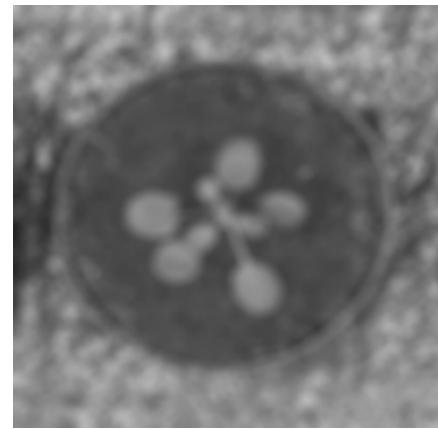
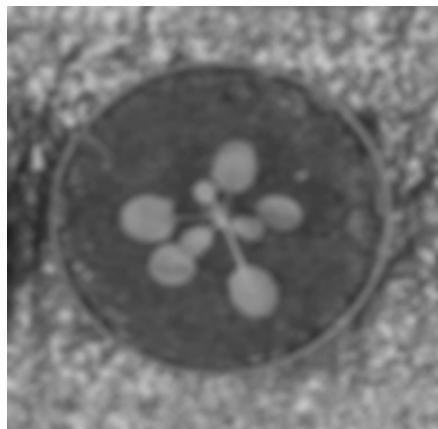
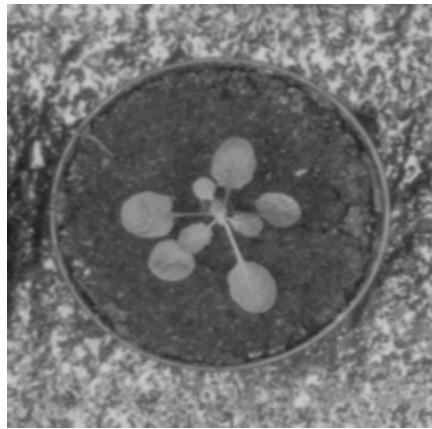
$W=5*5$



$W=9*9$



Original 265*282

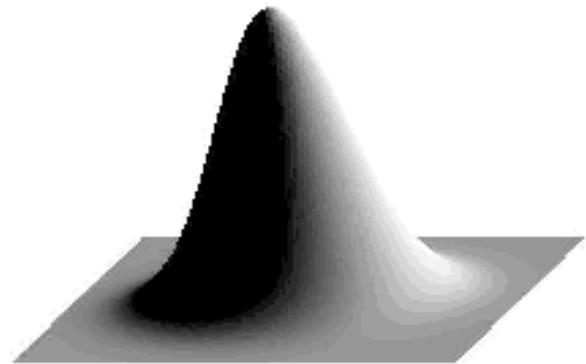


Choose the filter size smaller than the size of the structure to be preserved.

Beyond averaging

- Instead of giving the same weight to each pixel, it is possible to do a weighted averaging with the weight decreasing when we depart from the center of the window (for instance a Gaussian mask).

$$o(x, y) = I(x, y) * \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$



Filter with circular symmetry

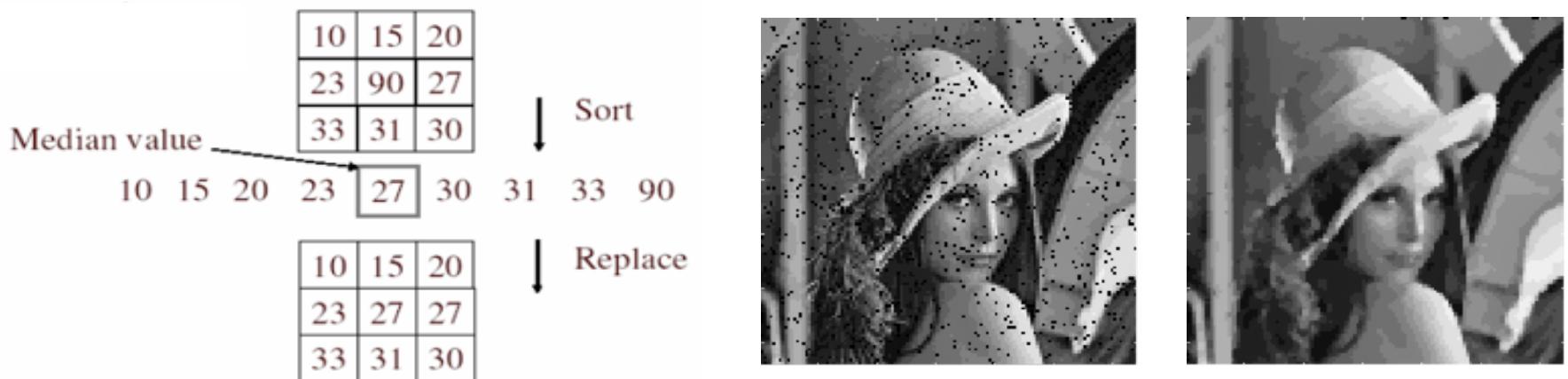
Parameter sigma

$$x^2 + y^2 = r^2$$

In ImageJ : Process-> Filter-> Gaussian blur

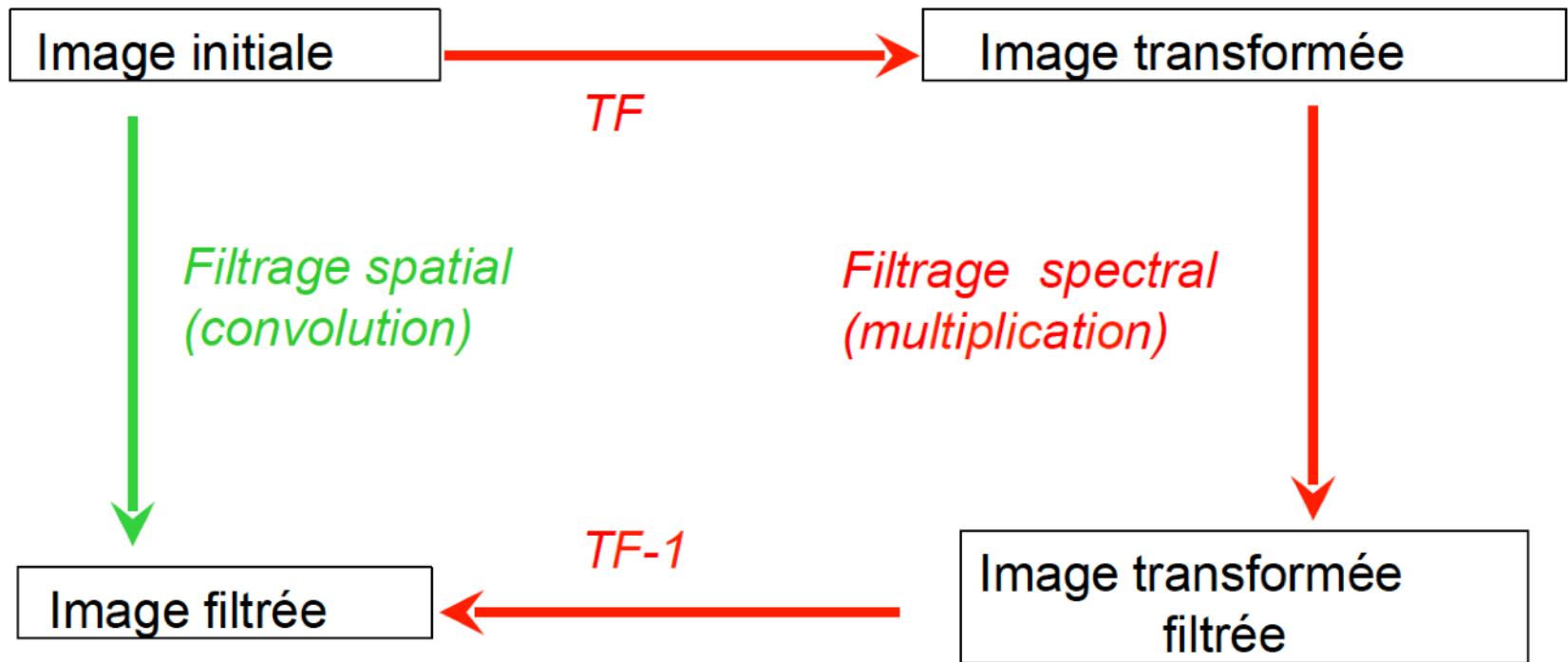
Non linear filtering

- If noise is non Gaussian, the optimal statistics is not the mean.
- For instance with impulsive noise, the median is more efficient than the mean.
- The central pixel is replaced by the median of its neighbourhood



In ImageJ : Process-> Filter-> Median

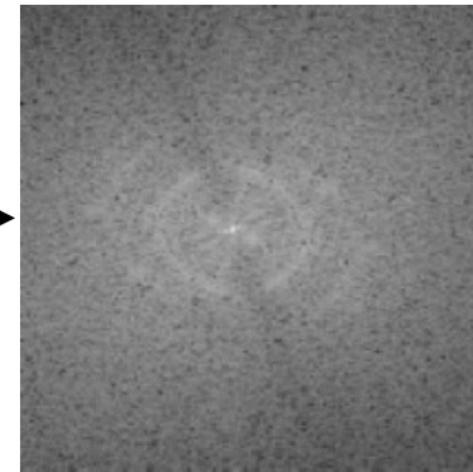
Filtrage dans le domaine spectral



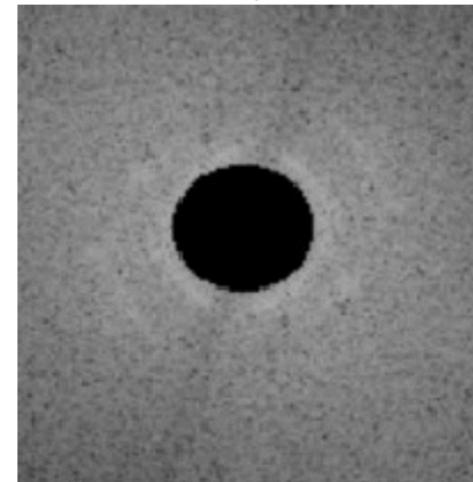
Dans le **domaine spatial**, le filtrage se fait par **convolution**. Dans le **domaine spectral** (ou **fréquentiel**), il se fait par **multiplication** (ou **masquage de l'image**).

Dans le cas des filtres dans le domaine fréquentiel non-multiplicatif, on ne peut pas obtenir le même résultat par convolution dans le domaine spatial.

Filtrage dans le domaine spectral

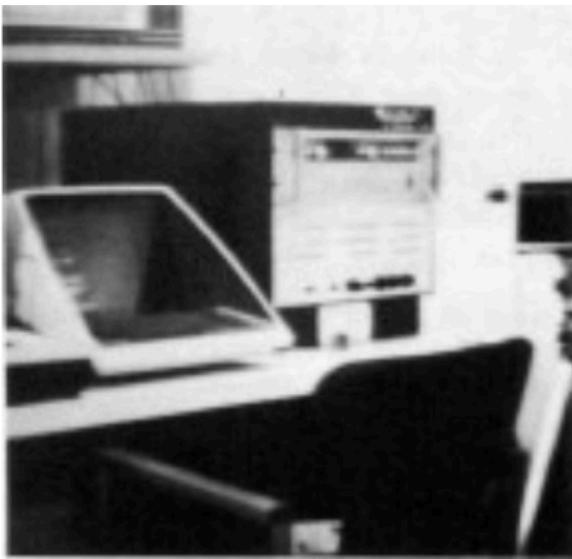


*Filtrage
spectral*

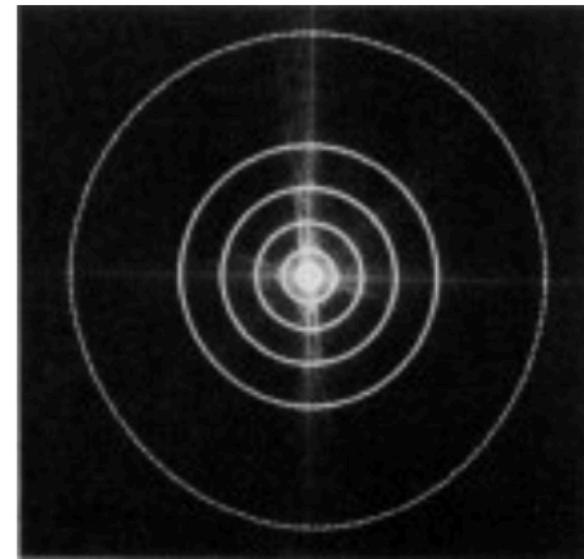


Bandes de fréquences

Image



Spectre de Fourier



Pourcentage de l'information de l'image inclus dans les cercles (plus petit vers le plus grand) :

90%



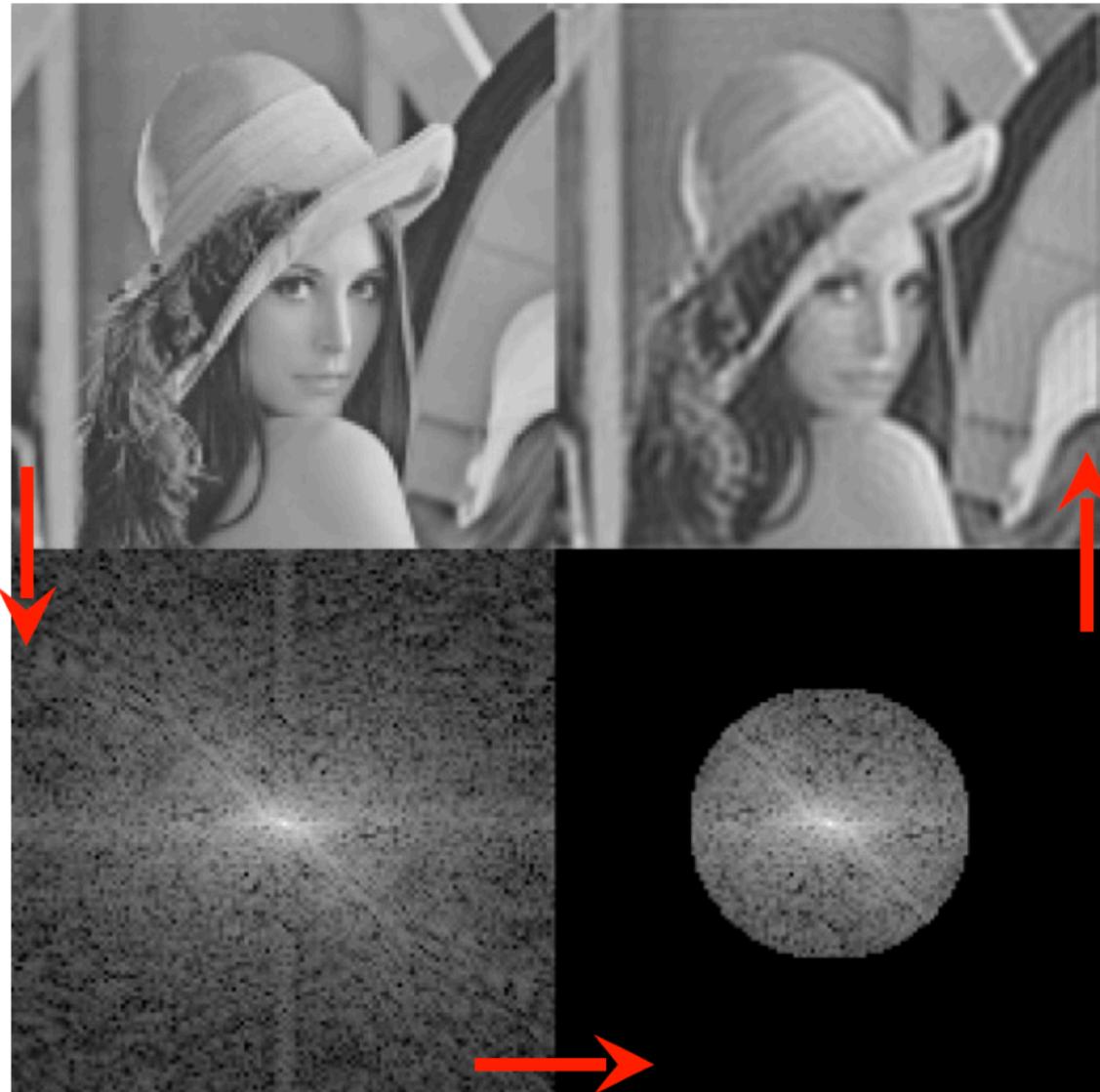
98%



99.5%

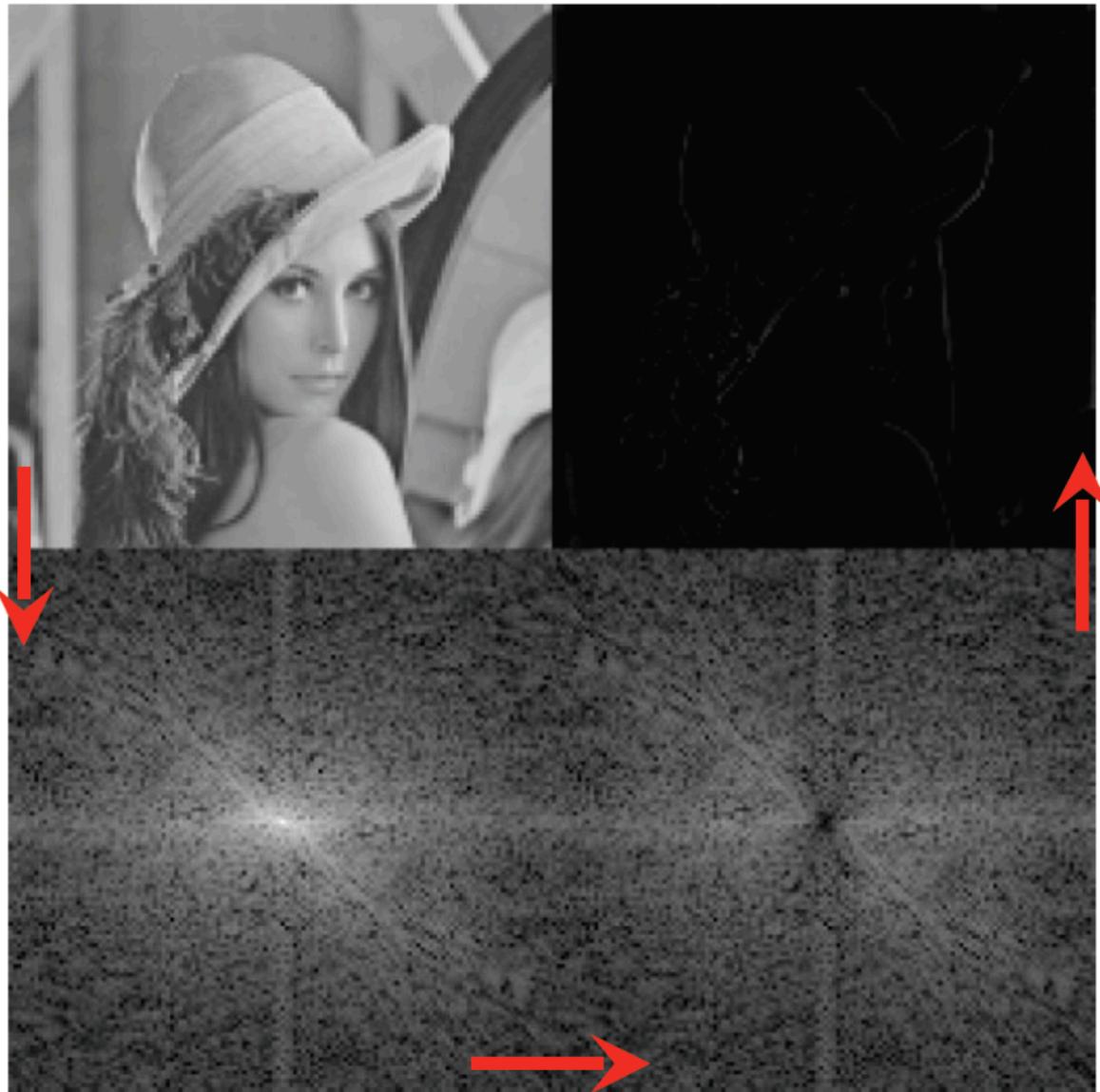


Filtrage passe-bas par TF



On efface les hautes fréquences de la TF en mettant les pixels loin du centre à zéro

Filtrage passe-haut par TF



On efface les basses fréquences de la TF en mettant les pixels au centre à zéro

FFT interest: removing structures

- Withdraw part of the spectrum corresponding to unwanted frequencies
- Easy when they are localised in the spectrum like with the periodic noise

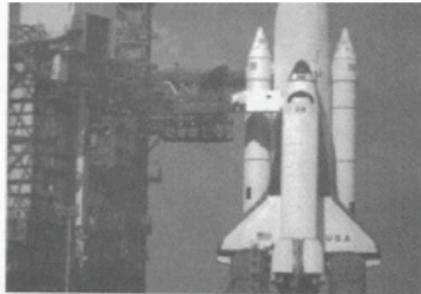


Figure 4.25a Original Space Shuttle image.

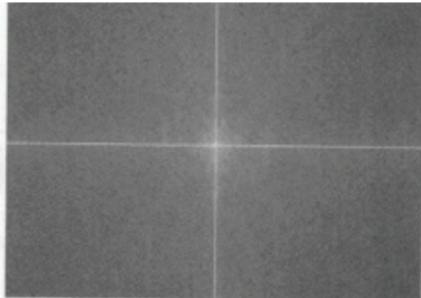


Figure 4.25b Fourier-transform image.

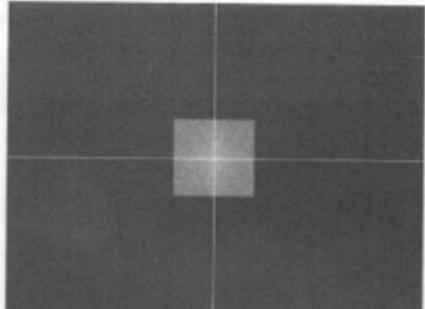


Figure 4.25c Fourier-transform image with the high frequencies zeroed.



Figure 4.25d Inverse Fourier-transform image showing a low-pass filtered version of the original image.

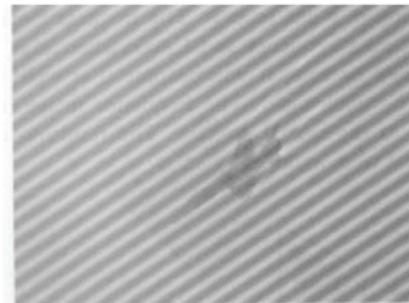


Figure 4.24a Original jet fighter image corrupted with a diagonal periodic noise pattern.

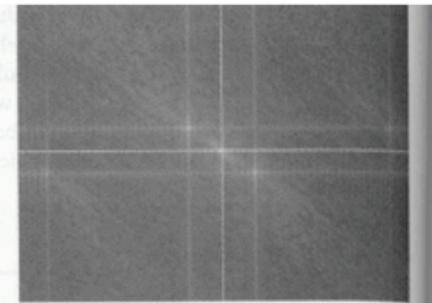


Figure 4.24b Fourier-transform image.

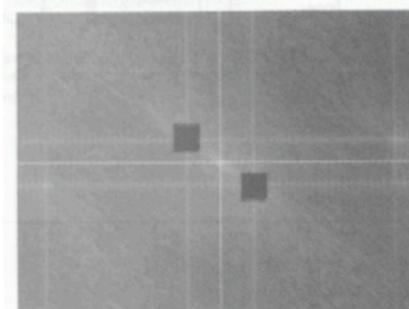
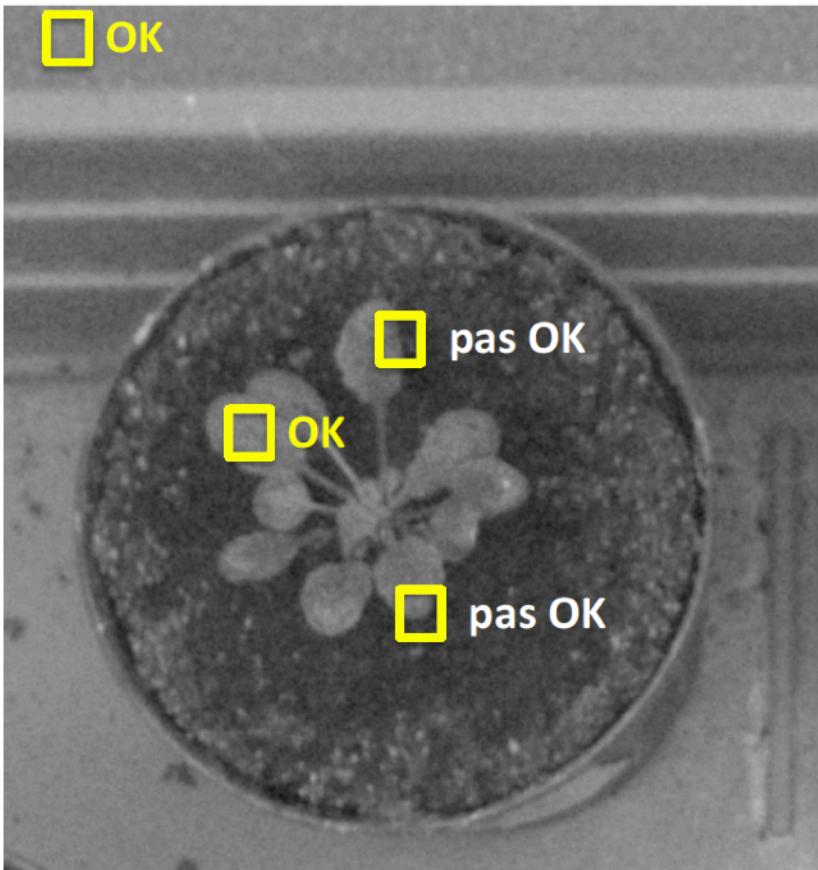


Figure 4.24c Fourier-transform image with the noise frequencies zeroed.



Figure 4.24d Inverse Fourier-transform image with most of the periodic noise removed—visibility of the aircraft is greatly improved.

Limites du filtrage par convolution



- Parfois filtrage pas suffisant
- Il faudrait pouvoir appliquer plusieurs fois dans les régions homogènes
- Tout en préservant les contours
- Idées :
 - Filtrage anisotropique
 - Filtrage non local
 - Mean shift

Filtrage anisotropique

On convole plusieurs fois par un filtre gaussien dont les coefficients sont forts si il n'y a pas de bords.

Perona, P. and Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion et (Barash 2002) IEEE Transactions on Pattern Analysis and Machine Intelligence, 12:629– 639

Image à l'itération t

$$I^{t+1}(\mathbf{x}) = \frac{\sum_{i=-1}^1 \sum_{j=-1}^1 I^t(\mathbf{x}_{i,j}) \cdot w^t(\mathbf{x})}{\sum_{i=-1}^1 \sum_{j=-1}^1 w^t(\mathbf{x})}$$

Tribute to Thomas GRENIER

Avec :

$$\mathbf{x} = (x_1, x_2)$$
$$w^t(\mathbf{x}) = e^{-\frac{(d^t(\mathbf{x}))^2}{2 \cdot \sigma_D^2}}$$
$$\mathbf{x}_{i,j} = (x_1 + i, x_2 + j)$$
$$d^t(\mathbf{x}) = \|\nabla I^t(\mathbf{x})\|$$

$\sigma_D = 2$

$\sigma_D = 1$

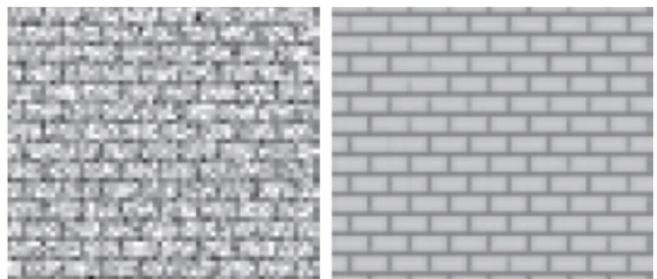
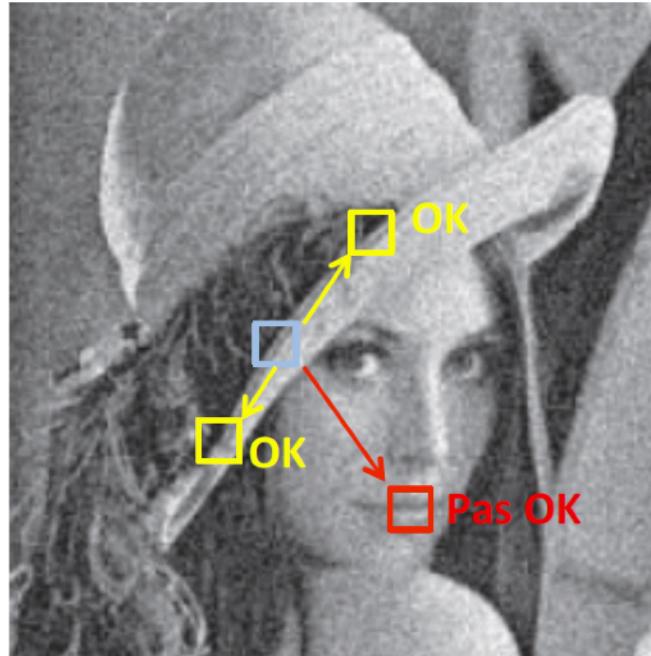
$\sigma_D = 0,5$

- **Paramètres :**

- Le nombre d'itérations : t (plus on itère plus on diffuse)
- Le seuil : sigma (plus il est haut plus on diffuse les bords)
- La taille de la fenêtre : ij (elle définit une échelle spatiale de diffusion).

Filtrage non local

- Moyenner des pixels dont l'environnement est proche dans le contenu de niveau de gris pas forcément proche spatialement.



Cela marche d'autant mieux que l'image de départ contient des motifs répétitifs

Paramètre : taille du patch.

Filtrage Mean shift

Le principe est d'appliquer l'algorithme suivant pour chaque pixel l'un après l'autre :

- Soit C la couleur/niveau de gris initiale du pixel.
- C est remplacée par la moyenne des couleurs dans le voisinage, mais calculée seulement sur les pixels dont la couleur/niveau de gris diffère de moins de I_{\min} de C .
- L'étape précédente est renouvelée jusqu'à ce que C ne varie plus
- Dans l'image résultat, le pixel prend la couleur finale C .

C'est une sorte de filtrage non local. Le filtrage par mean shift préserve les contours

Illustration : images débruitées

original



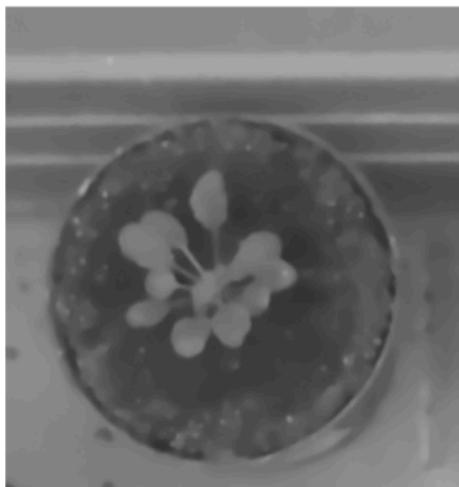
Gaussien sig 2 pixels



Median taille 6 pixels



Diffusion anisotrope



Moyenne non locale



Mean shift



200 iter, edge 5

Taille fenêtre 15

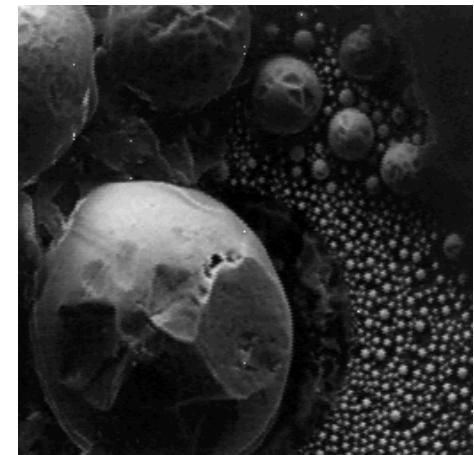
Taille fenêtre 15
Dist niveau de gris : 20

HANDS ON n°6

Exercice 1 (plutôt Fiji...)

- 1) Load images meb.bmp and display its histogram
- 2) From original image (use Image/duplicate), apply different filters (Process/smooth, Sharpen, Process/filters/ mean and gaussian) using a radius of 1.5.
- 3) Display all histograms.
- 4) Evaluate the influence of the size of the radius.

- 5) On the same image, add some impulse noise as Salt&Pepper (Process/noise)
- 6) Duplicate this image
- 7) Compare mean and median (Process/filter/median) filters with radius = 1.5



HANDS ON n°6

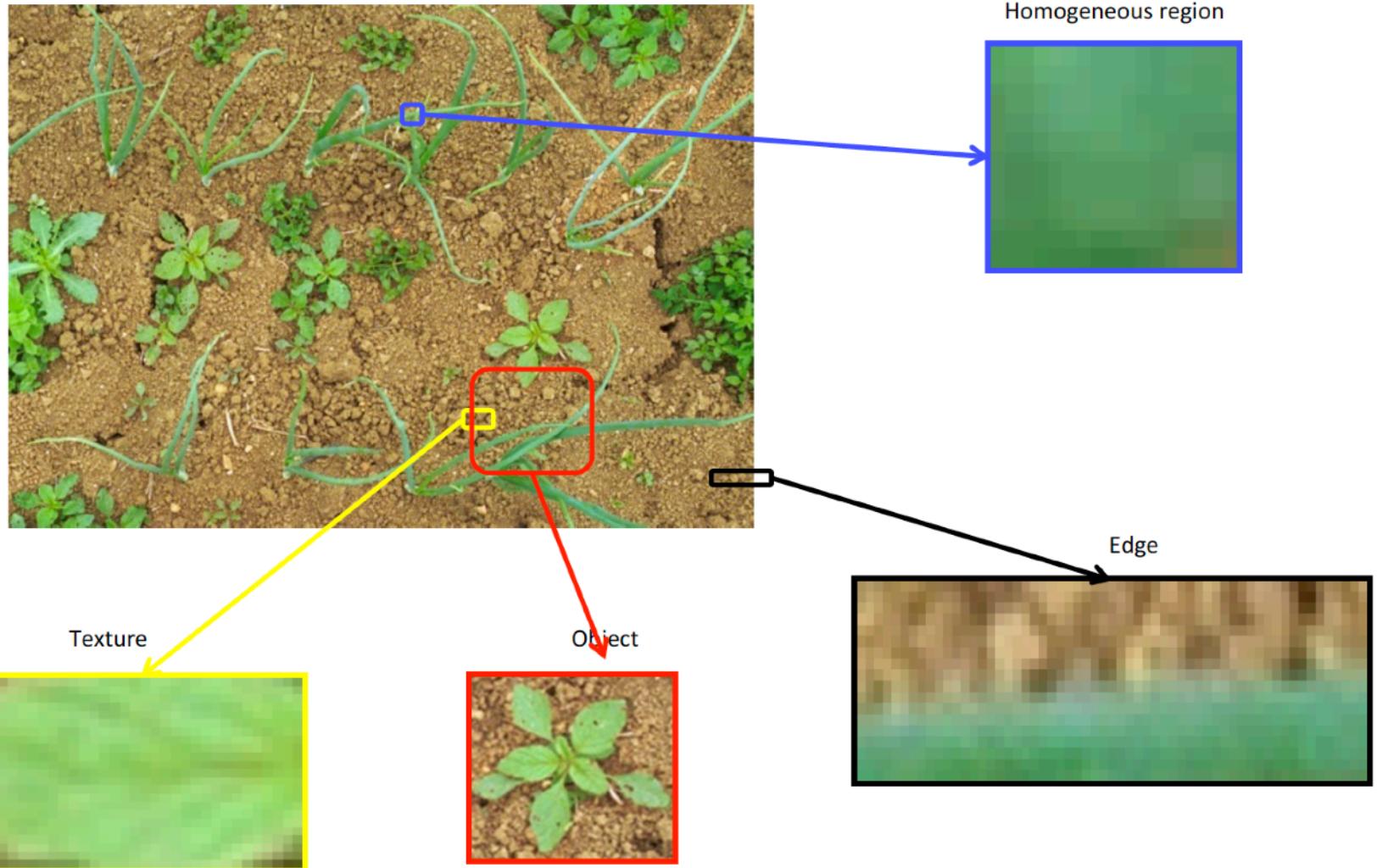
Exercice 2 (plutôt Octave/Matlab...)

1. Ouvrir image HOLMES_NB.BMP
2. Ajouter du bruit impulsionnel ‘salt & pepper’ avec variance = 0.5 . l'aide la fonction imnoise.
3. Afficher image originale + image bruitée
4. Réaliser un filtrage passe-bas par convolution spatiale avec masque de taille 5x5 à l'aide de la fonction filter2.
5. Faire de même dans le domaine fréquentiel ! (vous aurez besoin de FFT et des choses faites à l'exercice précédent...)
6. Afficher image originale + image bruitée + image filtrée spatiale + image filtrée fréq.
7. Tester différentes tailles de masques :)

Exercice 3 (idem...)

1. Ouvrir image HOLMES_NB.BMP
2. Ajouter du bruit impulsionnel ‘salt & pepper’ avec variance = 0.2 à l'aide la fonction imnoise.
3. Afficher image originale + image bruitée
4. Réaliser un filtrage gaussien (voir les fonctions filter2 (ou imfilter) et fspecial)
5. Réaliser un filtrage median (fonction medfilt2)
6. Comparer « visuellement » les résultats obtenus.

Chapitre 4 : Segmentation (tache de plus haut-niveau)



Chapitre 4 :

Segmentation (tache de plus haut-niveau)

- La segmentation vise à sous-diviser l'image en constituants distincts – objets
- La segmentation est liée à la reconnaissance
- La segmentation est normalement basée sur:
 - les discontinuités
 - les arêtes, les changements abrupts, ...
 - les similitudes (zones homogènes)
 - couleurs, textures, intensités, ...
- La segmentation est le découpage d'une image en **differentes régions et/ou contours.**

Segmentation régions/contours

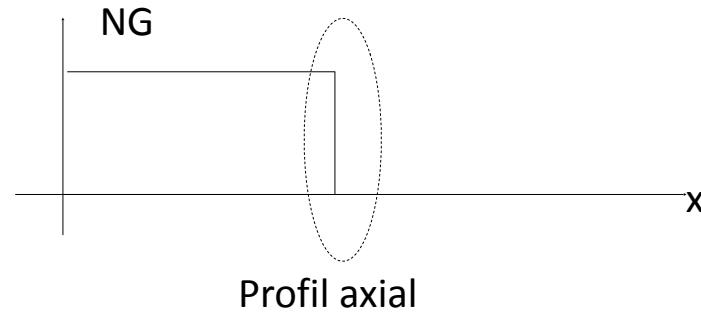
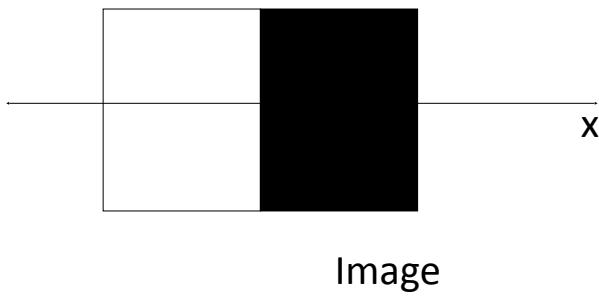
- **Approches régions/textures**
 - Rechercher les zones dans l'image sur un critère d'homogénéité (seuil, critères,...)
- **Approches contours**
 - Rechercher les discontinuités entre régions
- Il existe une dualité entre **régions (REGIONS)** et **contours (EDGES)**
 - Une région est une zone homogène d'un certain point de vue et est délimitée par un contour
 - Un contour sépare deux régions adjacentes



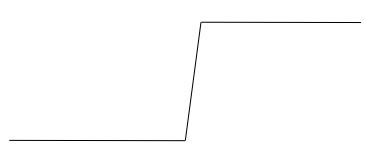
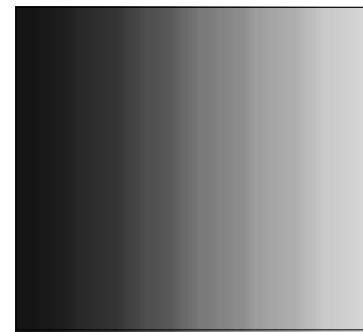
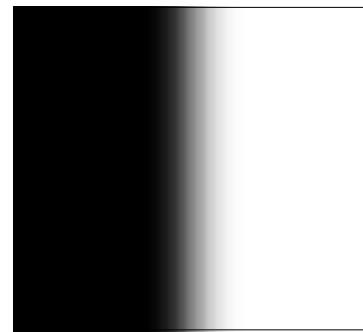
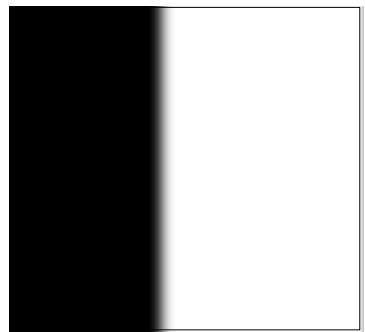
Segmentation contours : detecting edges



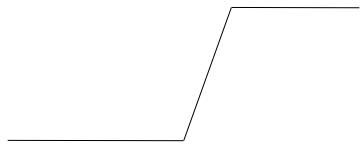
An edge is a rapid variation from a certain point of view



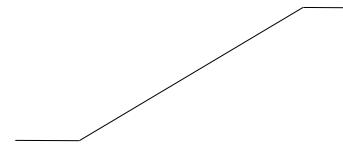
What is a rapid variation ?



Edge

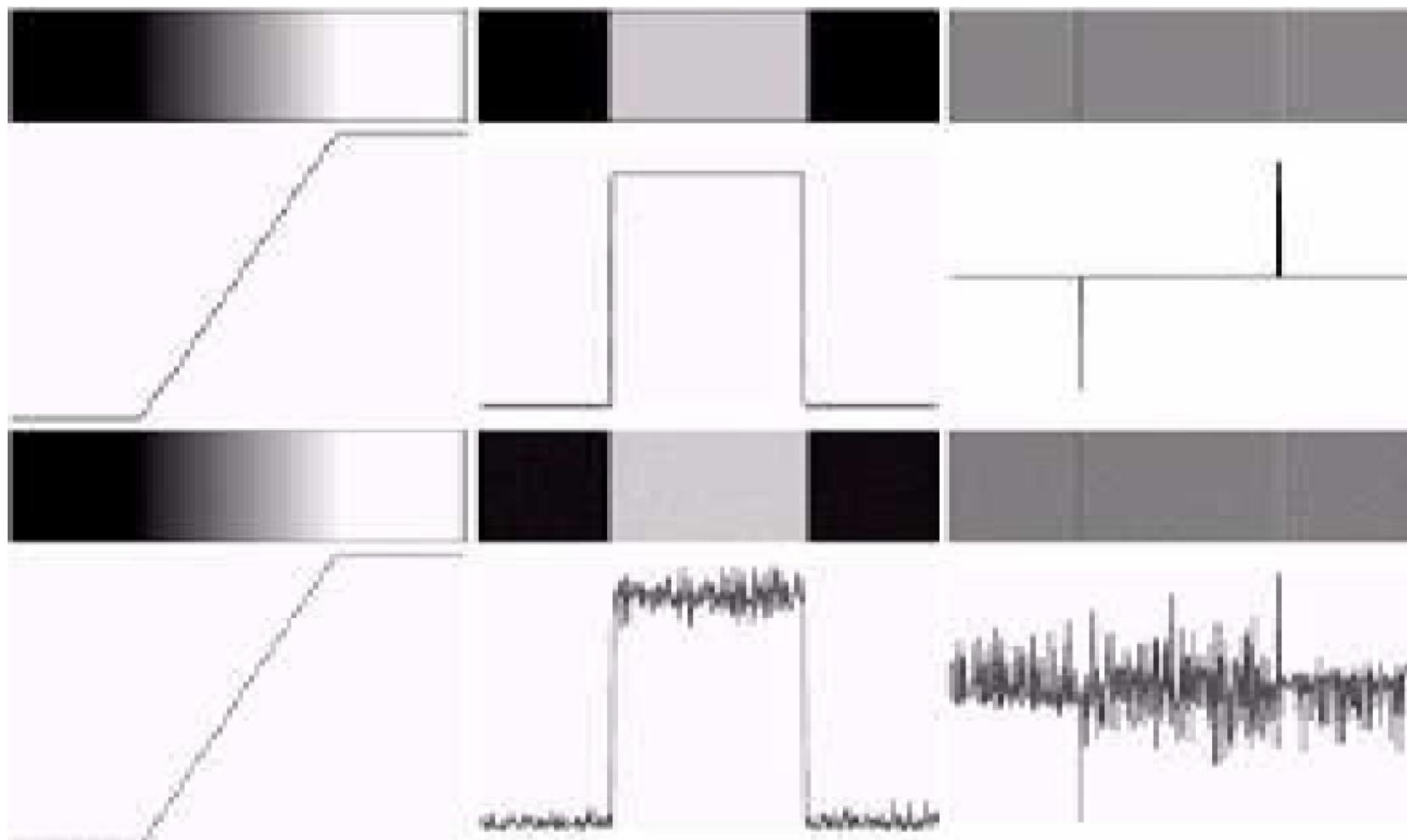


Edge?

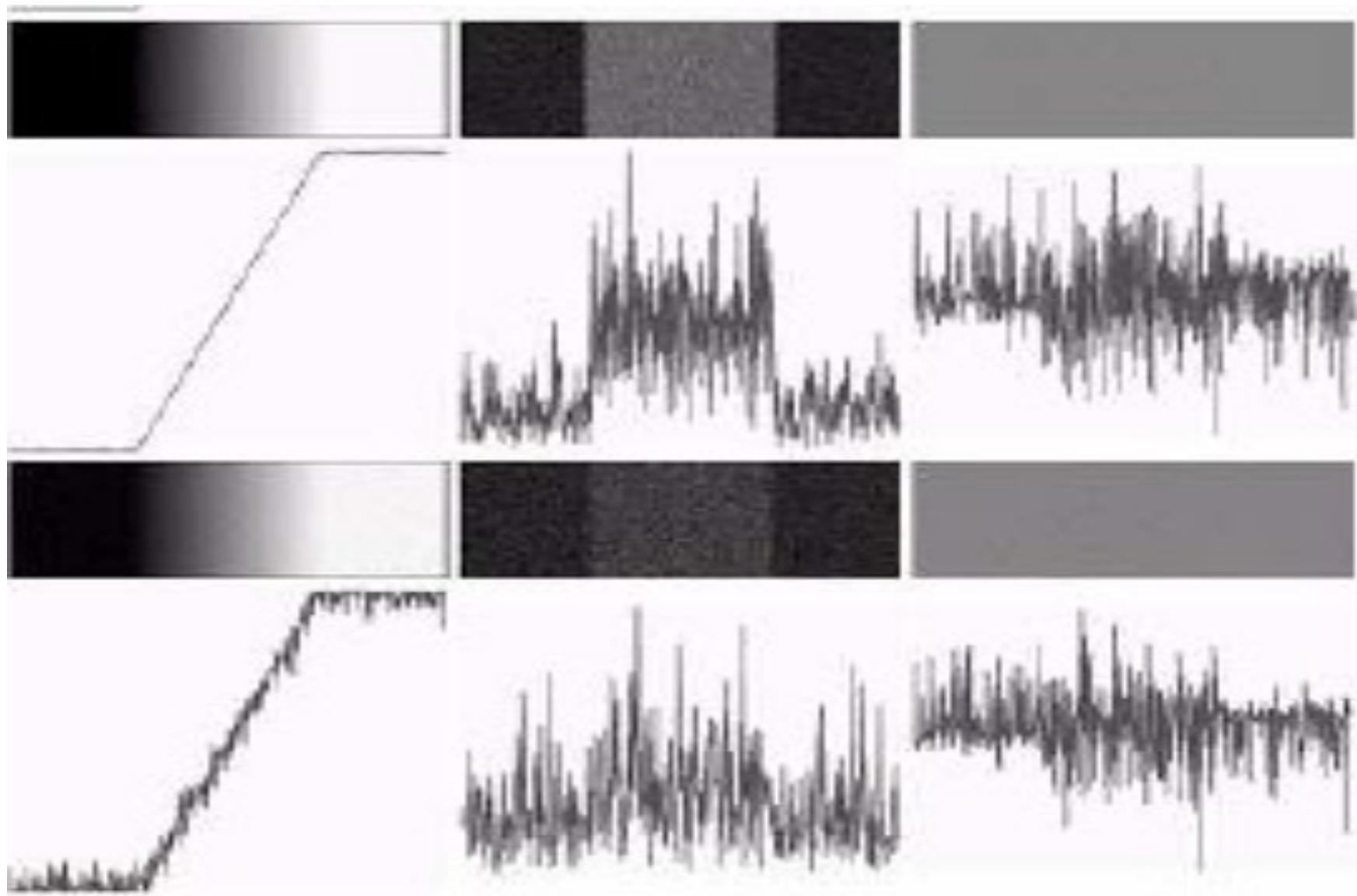


~~Edge~~

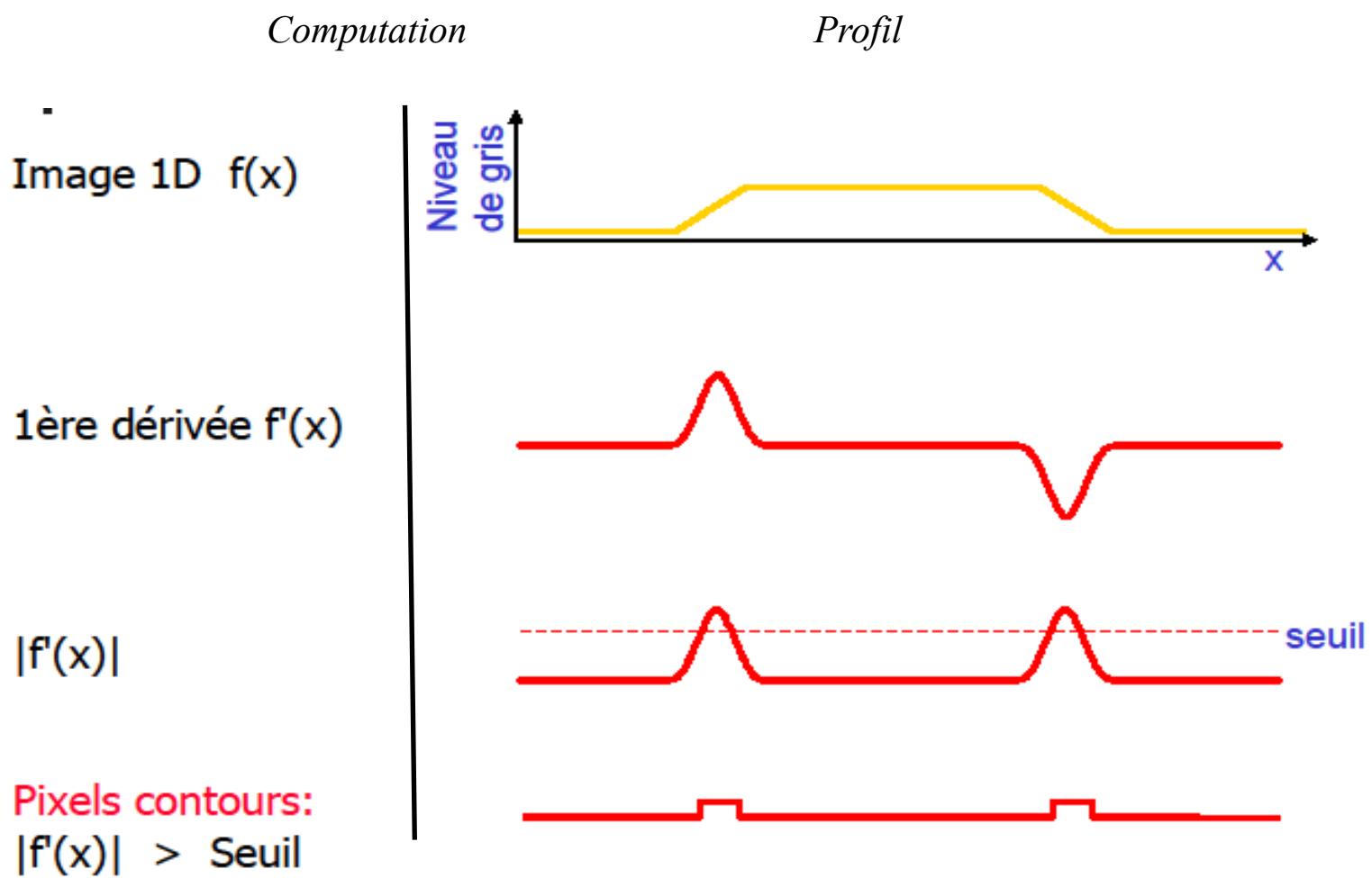
Edges with little noise



Edges with lots of noise



Gradient

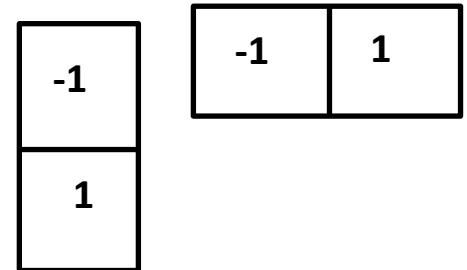


Gradient

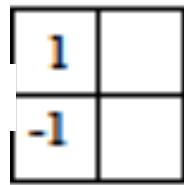
- First derivative along x axis

$$G_x = \frac{\Delta I}{\Delta x} = \frac{I(x + \Delta x) - I(x)}{\Delta x}$$

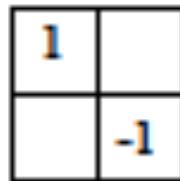
- Approximation of discrete derivative
- Same operation along y
- Norm of the gradient



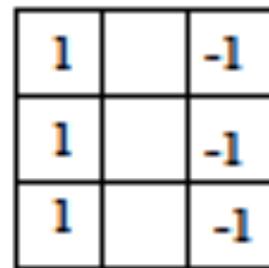
$$|G| = \sqrt{G_x^2 + G_y^2} \approx |G_x| + |G_y|$$



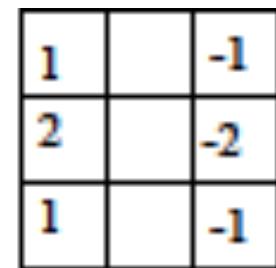
gradient



Roberts



Prewitt

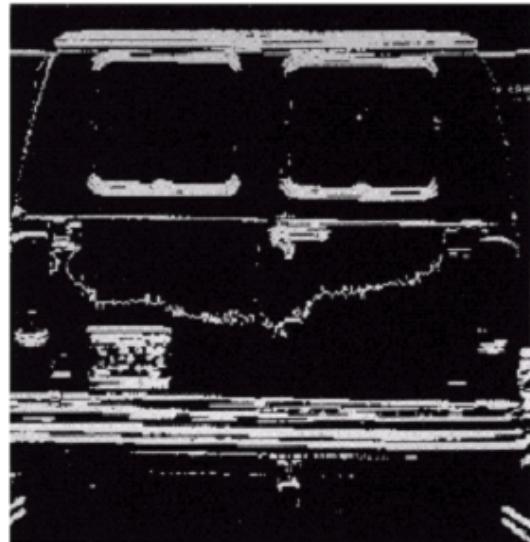


Sobel

Gradient

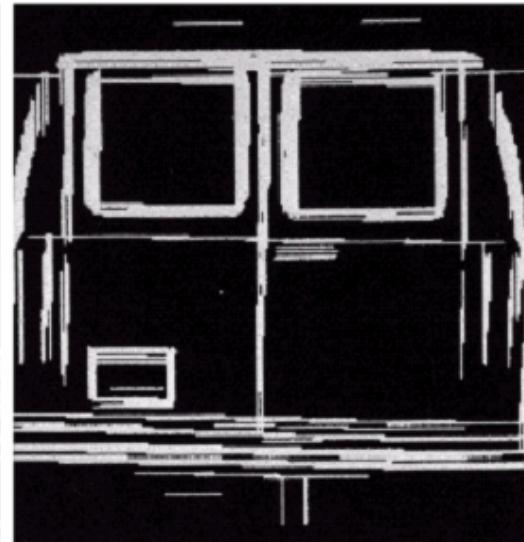
Horizontal edges

$$\frac{\Delta I}{\Delta y}$$



Vertical edges

$$\frac{\Delta I}{\Delta x}$$



Norm of gradient

$$\frac{\Delta I}{\Delta x} + \frac{\Delta I}{\Delta y}$$

Second derivative: Laplacian

- Another approach to find edges in images is to use the second derivative
- Detect zero crossings
- Laplacian

$$\nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

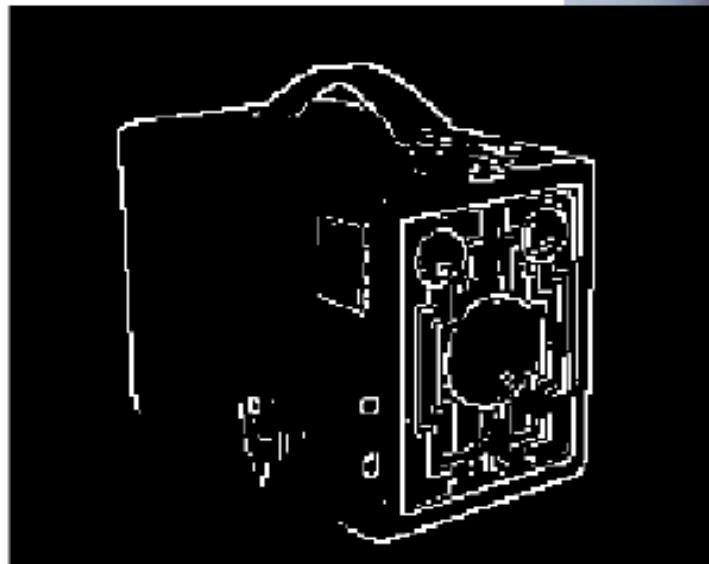
- Discrete approximation

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

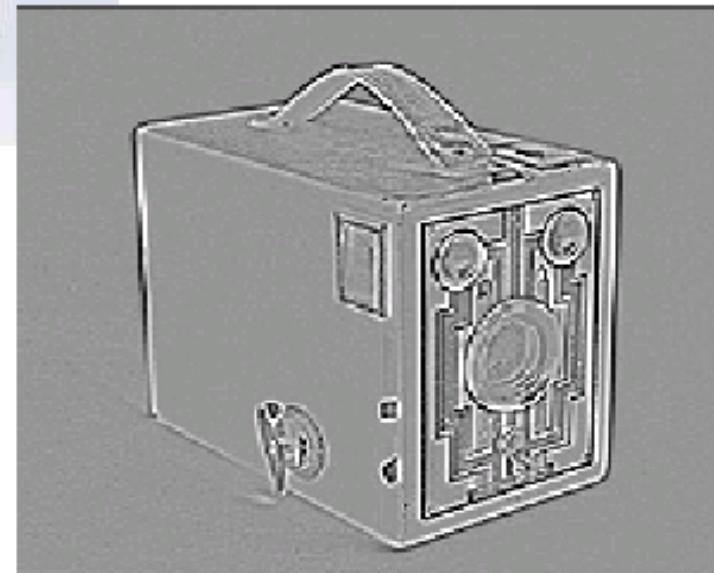
Comparison: gradient VS laplacian



Gradient



Laplacien



Illustrations sur des images débruitées

original



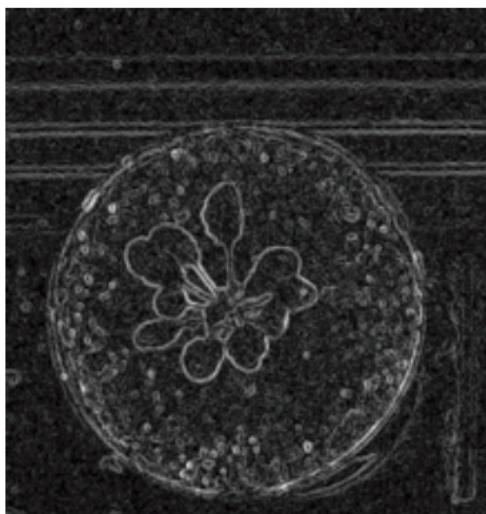
Gaussien sig 2 pixels



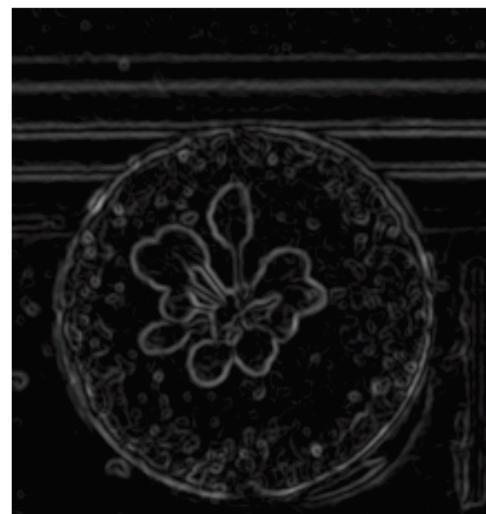
Median taille 6 pixels



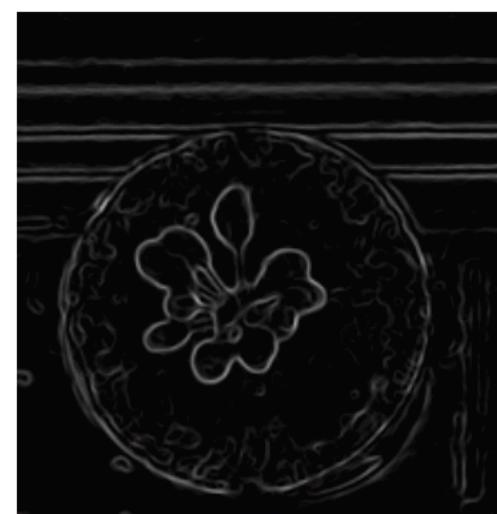
contours



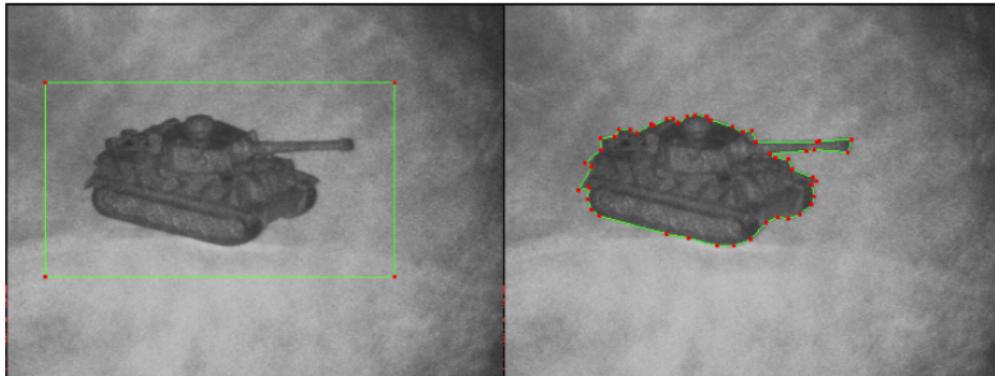
contours



contours

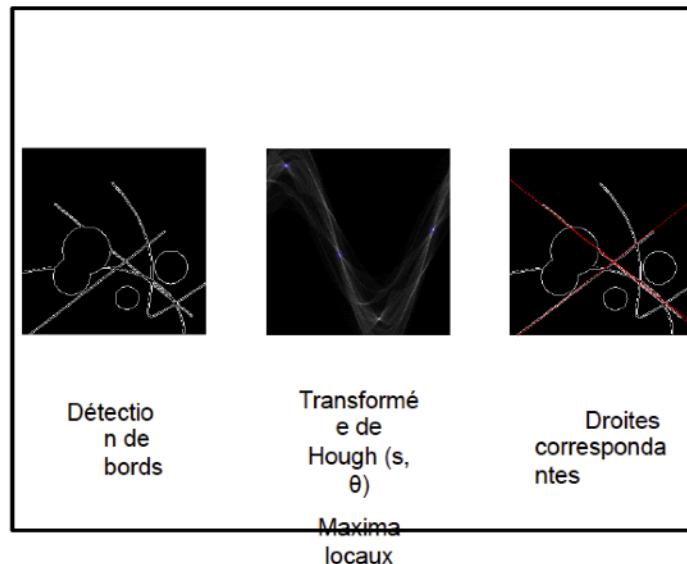
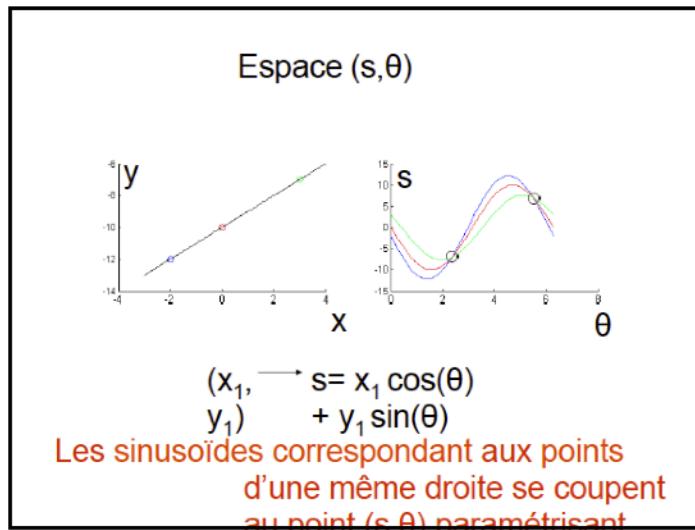


Autres méthodes « exotiques » de détection de contours



Contours actifs – Snake

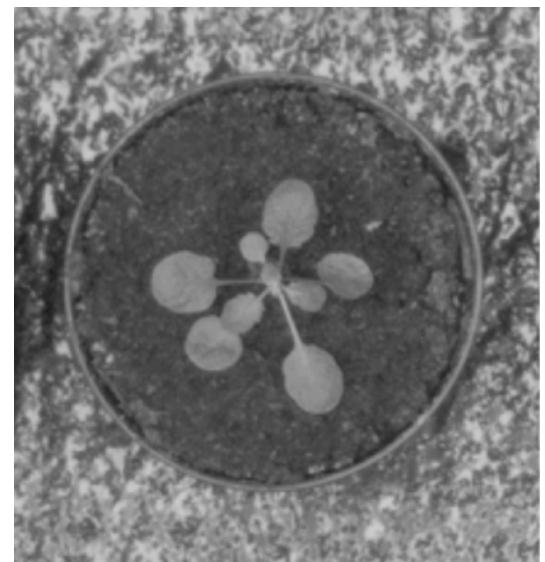
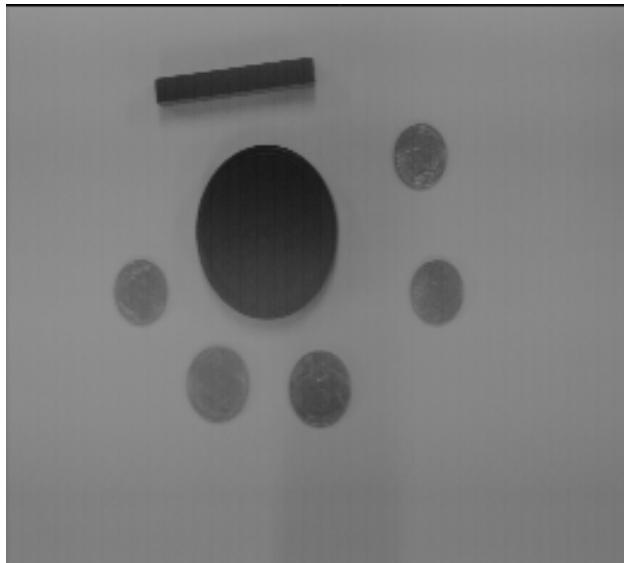
- Contour initial
- Noeuds cohésion
- Tend vers les forts gradients



Transformée de Hough – DéTECTEUR de ligne, cercle,...

HANDS ON n°7

- 1) Load morpho.bmp
 - 2) Detect edges with Process/Find edges
 - 3) Duplicate image morpho.bmp
 - 4) Add noise in morpho.bmp
 - 5) Detect edges in the noisy image with find edges.
Conclusion.
 - 6) Apply a filter on the noisy morpho.bmp
 - 7) Detect edges in the denoised image.
Conclusion.
- 8) Do the same with Arabette2

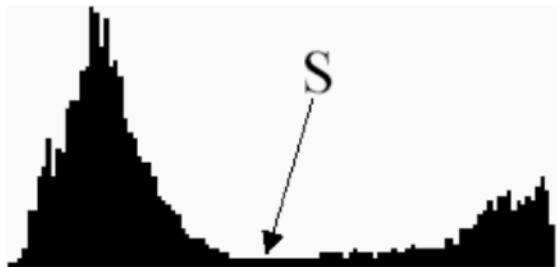


Segmentation par seuillage

- Le seuillage est une méthode simple et très populaire pour la segmentation d'objets dans les images numériques.
- Le seuillage peut être de nature
 - *Globale* : un seuil pour toute l'image
 - *Locale* : un seuil pour une portion de l'image
 - *Adaptative* : un seuil qui s'ajuste selon les images/parties de l'image.
- Seuillage de base (2 classes) – principe :
 - Si $valeur(pixel) \geq seuil$ alors $valeur(pixel) = 1$
 - Si $valeur(pixel) < seuil$ alors $valeur(pixel) = 0$
- Le résultat du seuillage est une image binaire.
- Il est aussi possible d'avoir n seuils pour séparer l'image en $n-1$ classes.
- **Problème** : choix du seuil ?

Seuillage d'histogramme

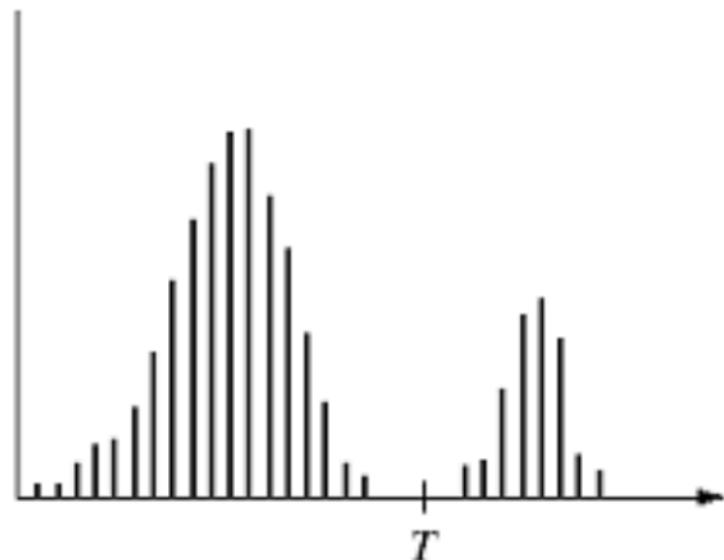
- Avantages
 - Universel, temps réel, simplicité
 - Fonctionne bien sur des histogrammes multi-modaux
- Inconvénients
 - Connaître le nombre de classes
 - Apparition de faux éléments (aucune prise en compte de la composante spatiale)
 - Nombre de modes souvent nombre de classes attendu



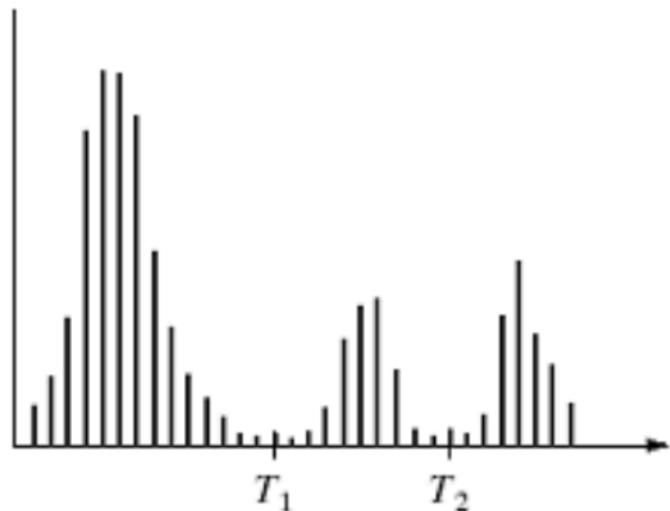
2 modes pour 3 classes

Seuillages d'histogramme

$$g(x, y) = \begin{cases} 1 & \text{si } f(x, y) > T \\ 0 & \text{si } f(x, y) \leq T \end{cases}$$

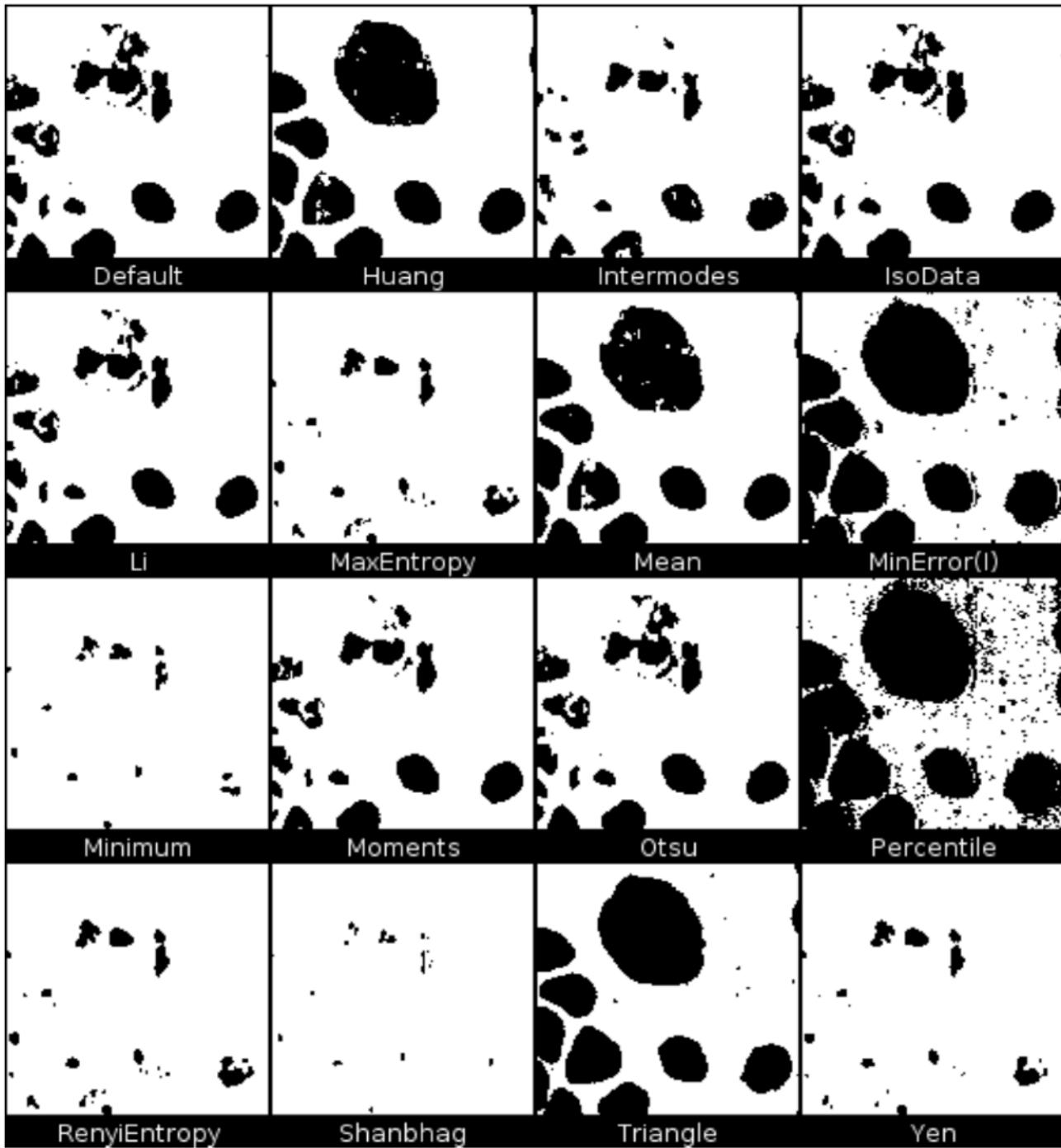
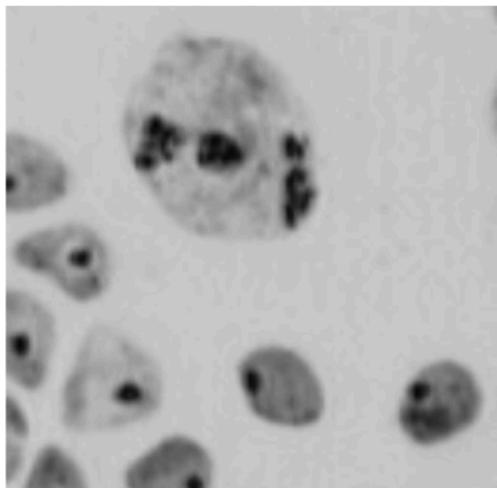


$$g(x, y) = \begin{cases} 2 & \text{si } f(x, y) > T_2 \\ 1 & \text{si } T_2 \geq f(x, y) > T_1 \\ 0 & \text{si } f(x, y) \leq T_1 \end{cases}$$



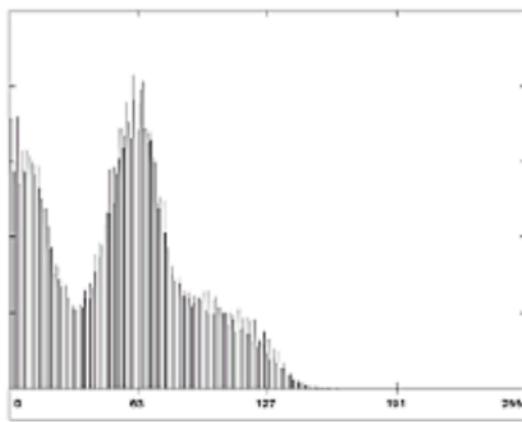
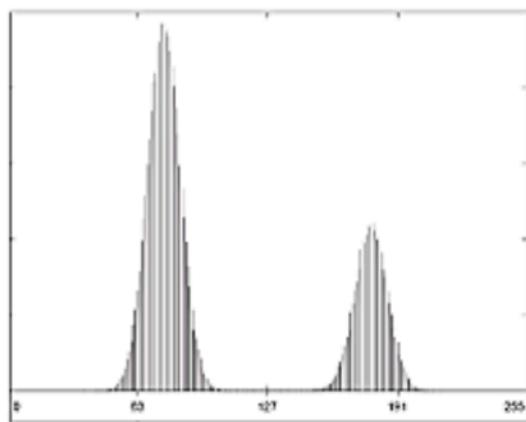
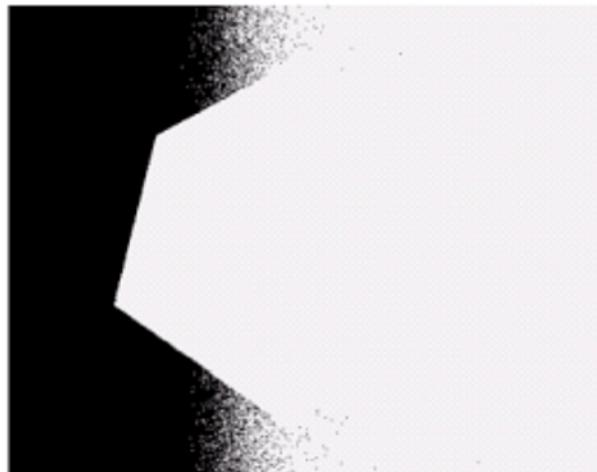
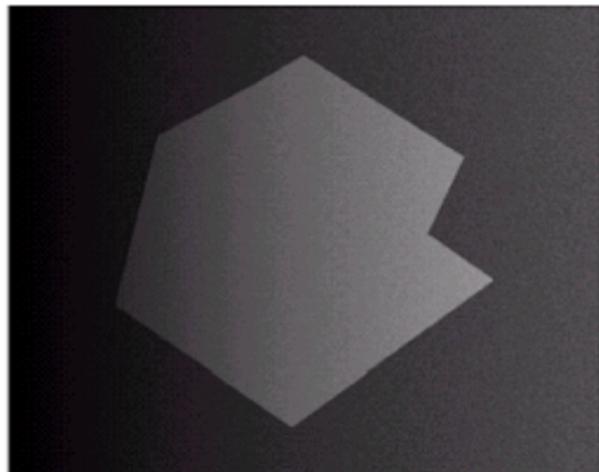
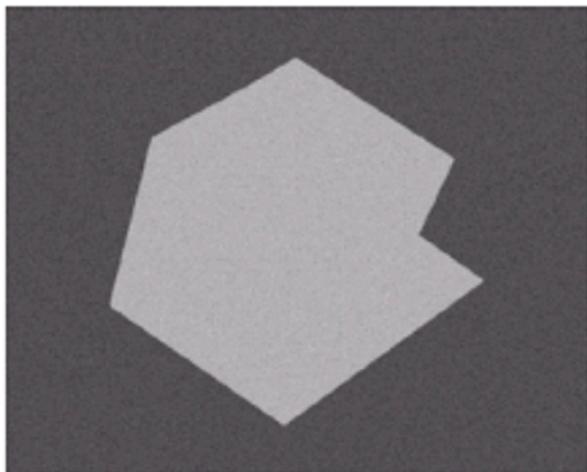
Choix du seuil global ?

Various methods



Seuillage global - problème

Problème d'éclairage ?



Seuillage local adaptatif

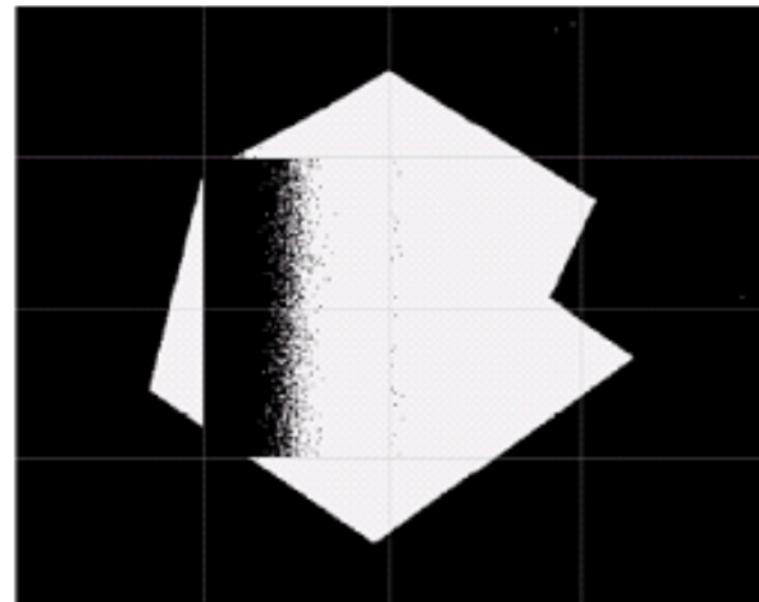
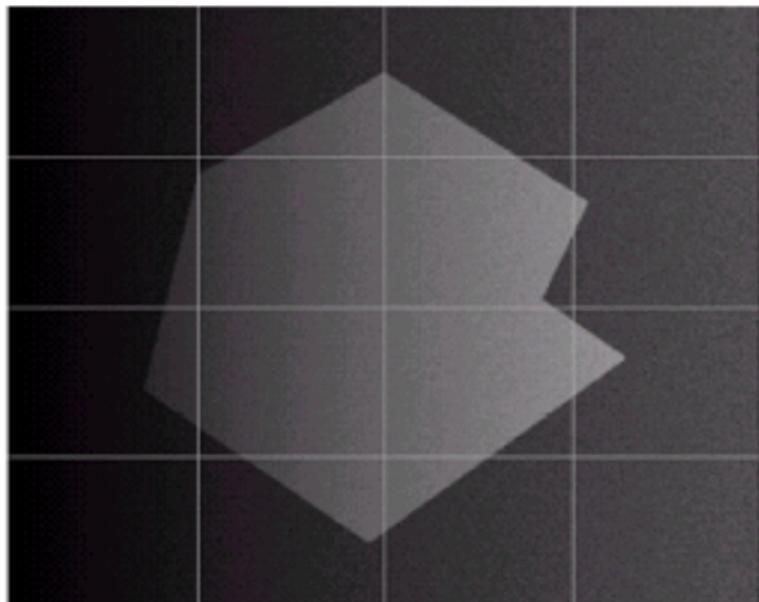
Nous avons besoin de séparer l'image en sous images, et de traiter chacune avec son propre seuil

Le choix de la dimension des sous-images est critique

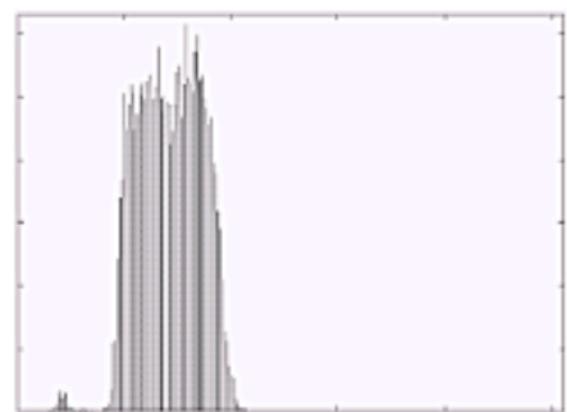
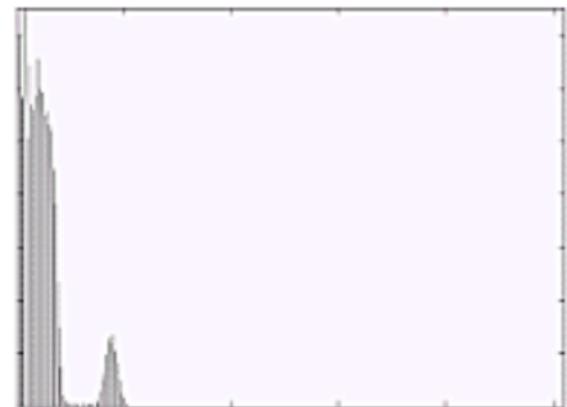
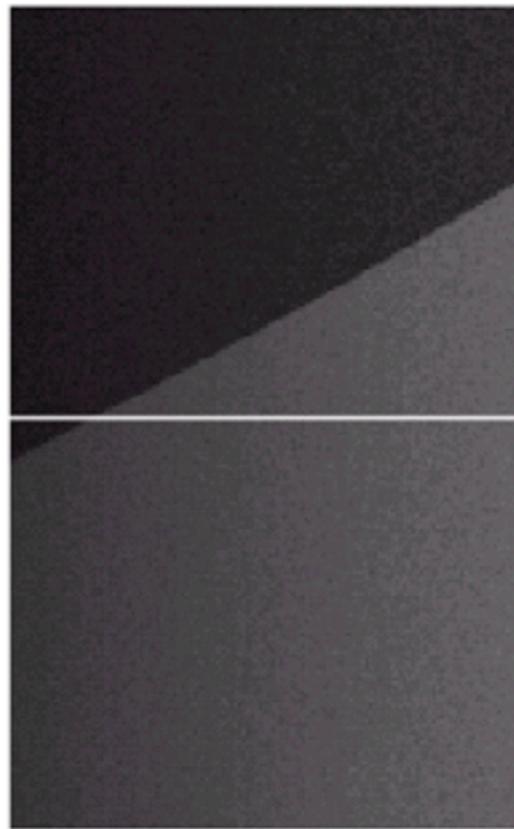
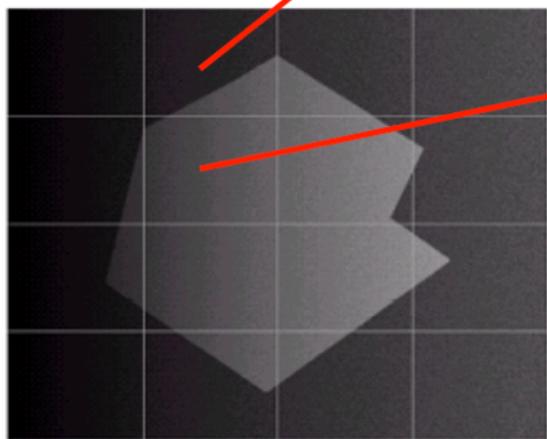
Avant de traiter chaque sous-image, nous vérifions la variance des tons de gris pour décider s'il existe un besoin de segmentation

Exemple : pas besoin si $\text{variance} < 100$

Les 4 sous images de coins ne sont pas traitées car $\text{variance} < 100$



Bimodal



Bimodal ??

Division-Fusion de régions

Initialisation

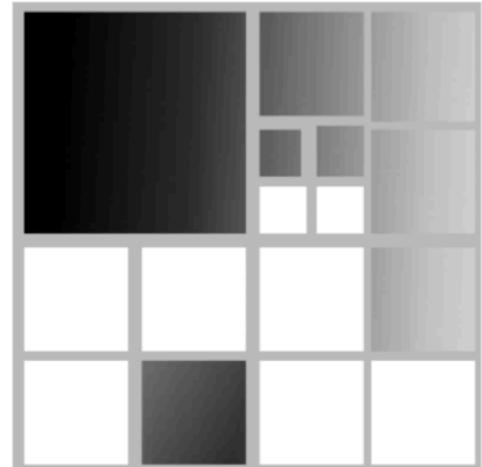
l'image initiale entière forme un bloc

Etape de division

Diviser récursivement tout bloc non-homogène selon un prédictat défini (variance, max-min, ...)

La division d'un bloc donne 4 sous-blocs

Les attributs de chaque sous-bloc sont recalculés



Etape de fusion

Regrouper les blocs adjacents représentant des régions homogènes selon un prédictat défini



Post-traitement

Lissage pour éliminer la forme carré des régions

Croissance de régions

L'algorithme débute avec un pixel, et "collectionne" les pixels voisins qui répondent à un critère d'appartenance :

- Variance faible

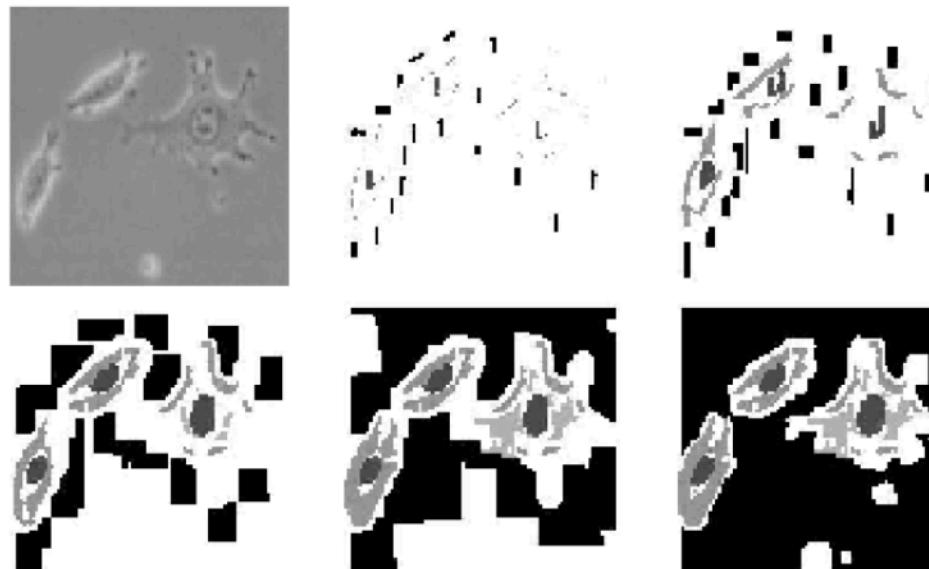
- Niveau de gris répondant un un seuil

- ...

Le(s) pixel(s) initiaux (ou initiaux) sont appelés "germes" ou "semences".

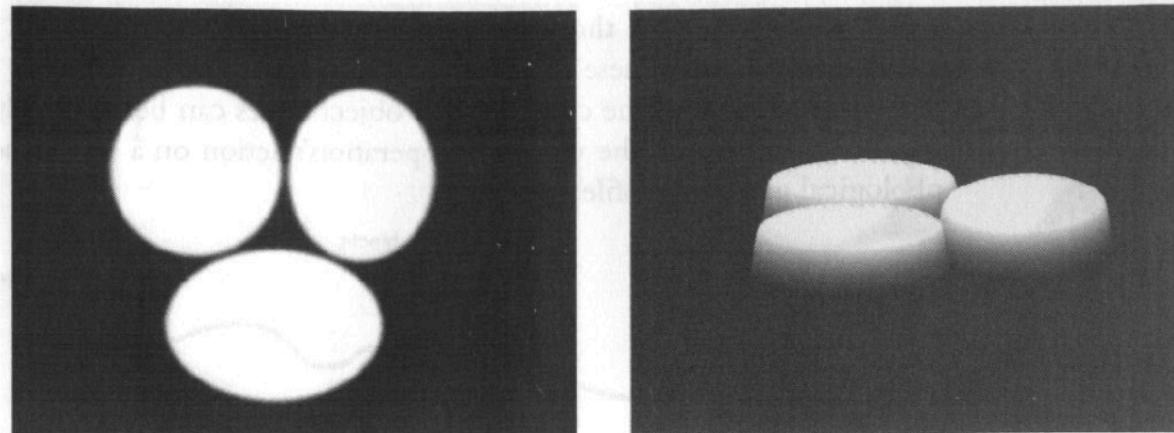
La région "grandit" à partir de son germe.

Une connaissance a priori est utile pour sélectionner les germes.

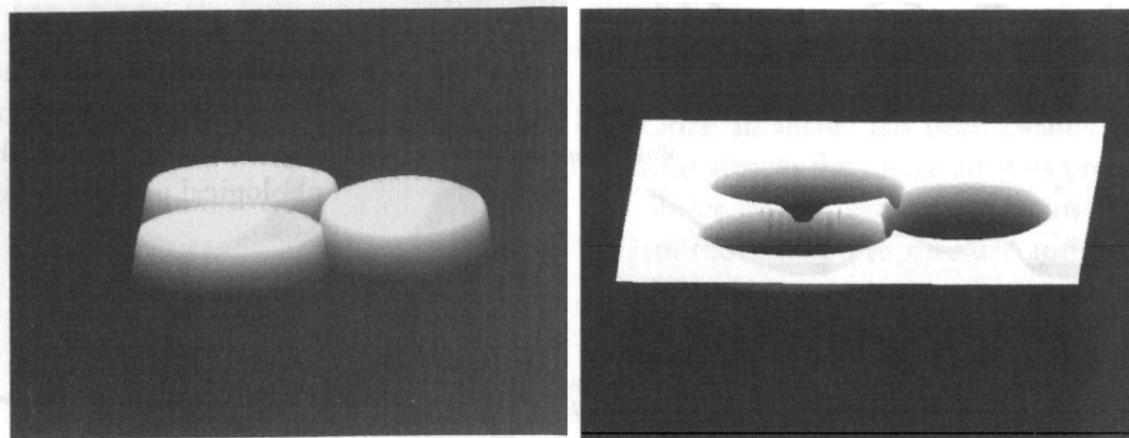


Segmentation par partage des eaux / Watershed

On considère la visualisation de l'image en 3D, en utilisant le ton de gris comme troisième dimension

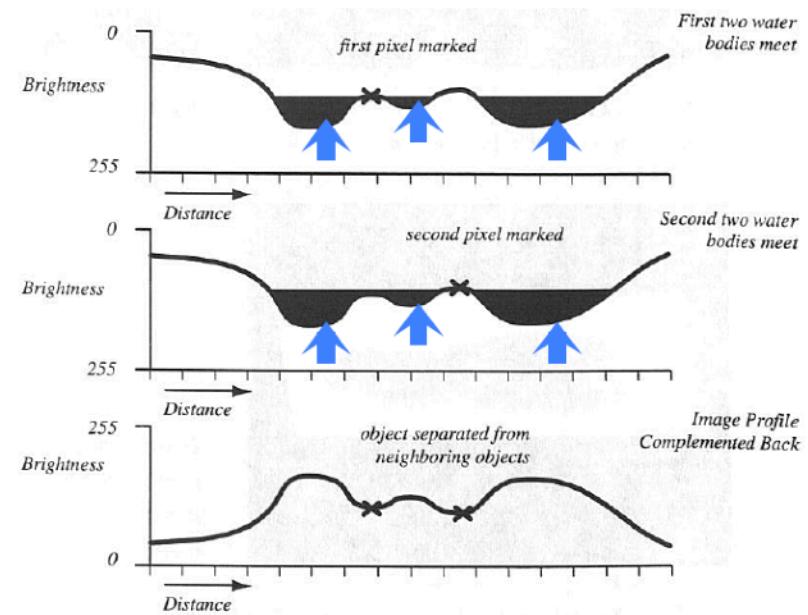
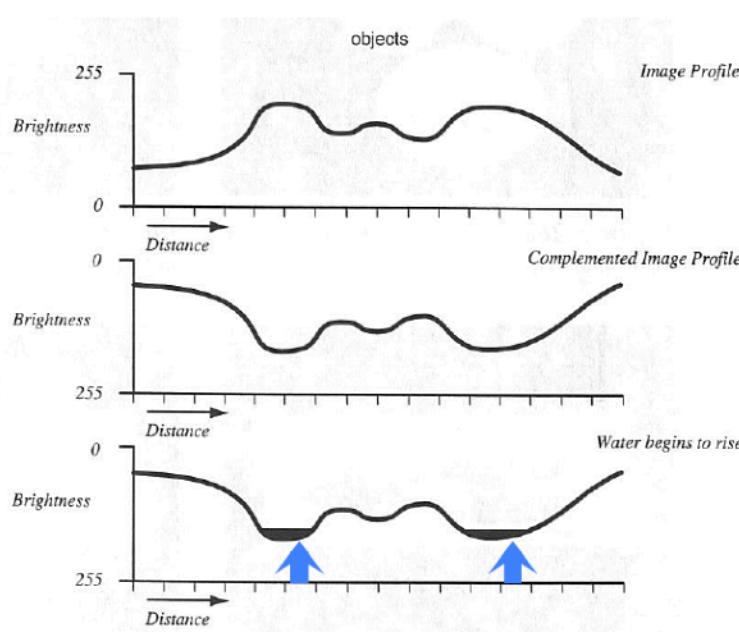
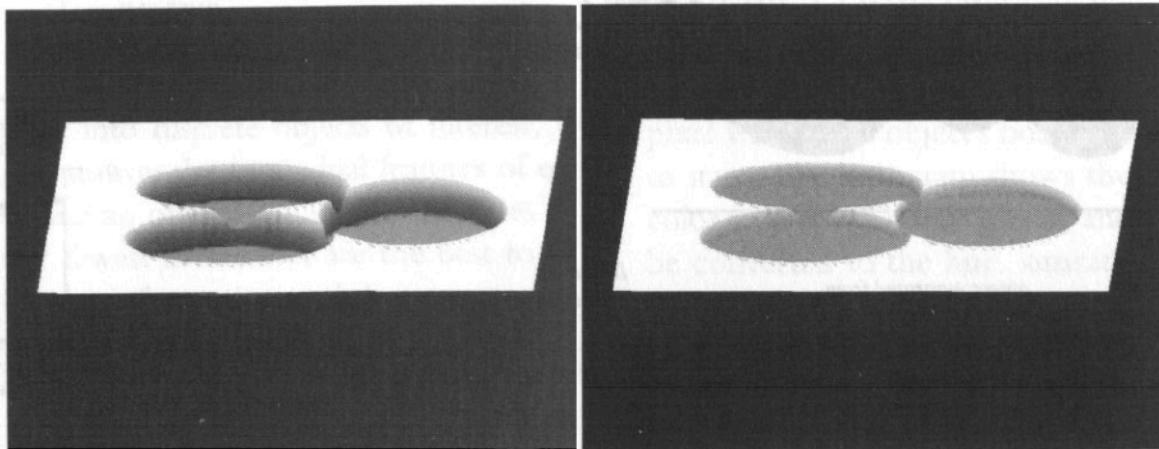


Ensuite on "complémente" les valeurs pour créer des zones inondables.



Segmentation par partage des eaux / Watershed

Enfin, on "infiltre" les cavités des zones inondables



Segmentation – conseils

La segmentation d'une image cause encore aujourd'hui beaucoup de problèmes.

- Aucune méthode ne fonctionne pour toutes les images.
- Pas de garantie, pas de recette miracle!

Pour certaines applications, on réalise qu'on peut éviter la segmentation complète de l'image. C'est souvent mieux.

Le pré-traitement des images, la sélection de capteurs et sources d'énergie appropriées, et la prise contrôlée des images rendent cette étape plus facile et plus efficace.

Le problème d'évaluer le résultat d'une segmentation n'est pas évident. Bien souvent, il est subjectif et varie d'une personne à l'autre.

Un des principaux problèmes est de définir le but de la segmentation : Qu'est-ce qu'on recherche exactement dans l'image ?

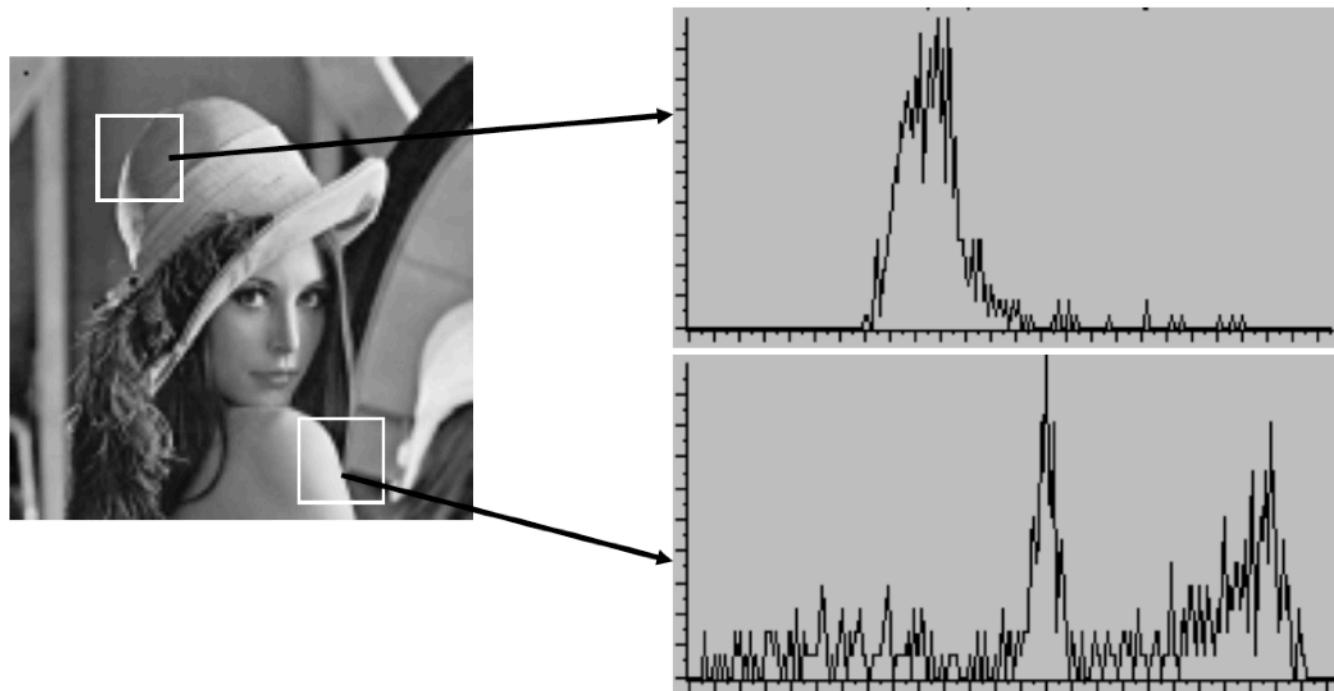
- Eléments globaux de l'image ou détails fin de la composition ?
- Présence d'un humain ou détails du visage ?

Il est bon de se poser aussi la question de ce que l'on veut faire ensuite avec la segmentation. Cela permet de définir le degré de précision nécessaire.

Segmentation par approche région

Features description

Measures on a local scale :



Computation on a 16x16 scale



128x128 pixels



Mean



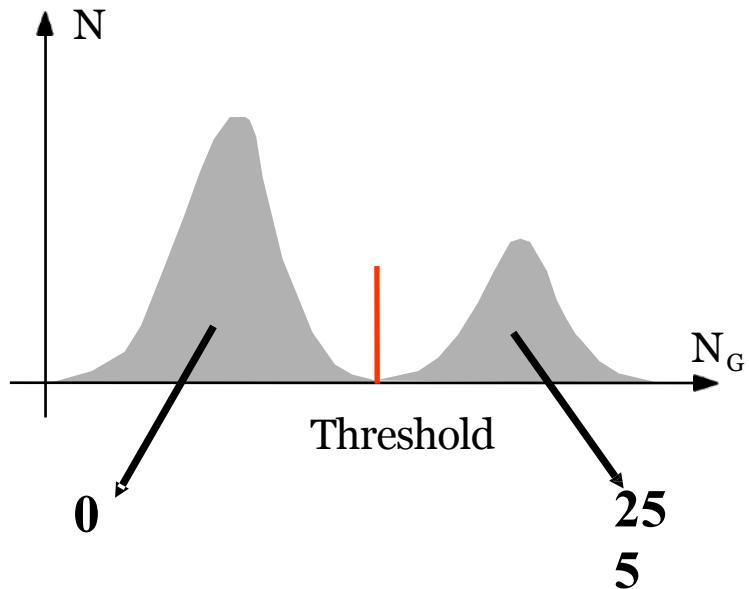
Variance

Thresholding

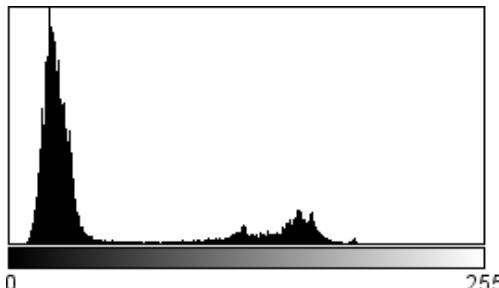
Single thresholding:

- Pixels with grey-level value < threshold \rightarrow classe 0
- Pixels with grey-level value > threshold \rightarrow classe 1 (1 or 255)

Simple technique/approach for region segmentation \rightarrow various improvements



Thresholding



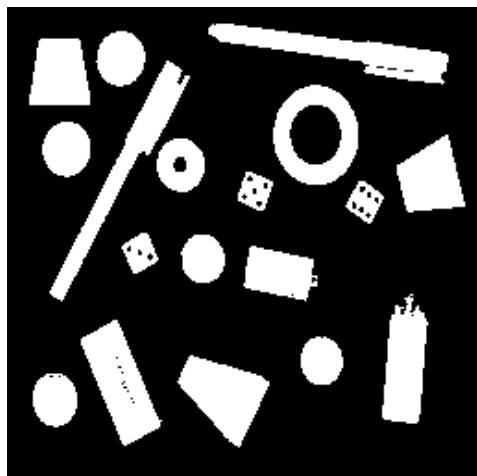
Count: 65536
Mean: 53.293
StdDev: 51.807

Min: 7
Max: 188
Mode: 21 (3542)



Default B&W

Dark background

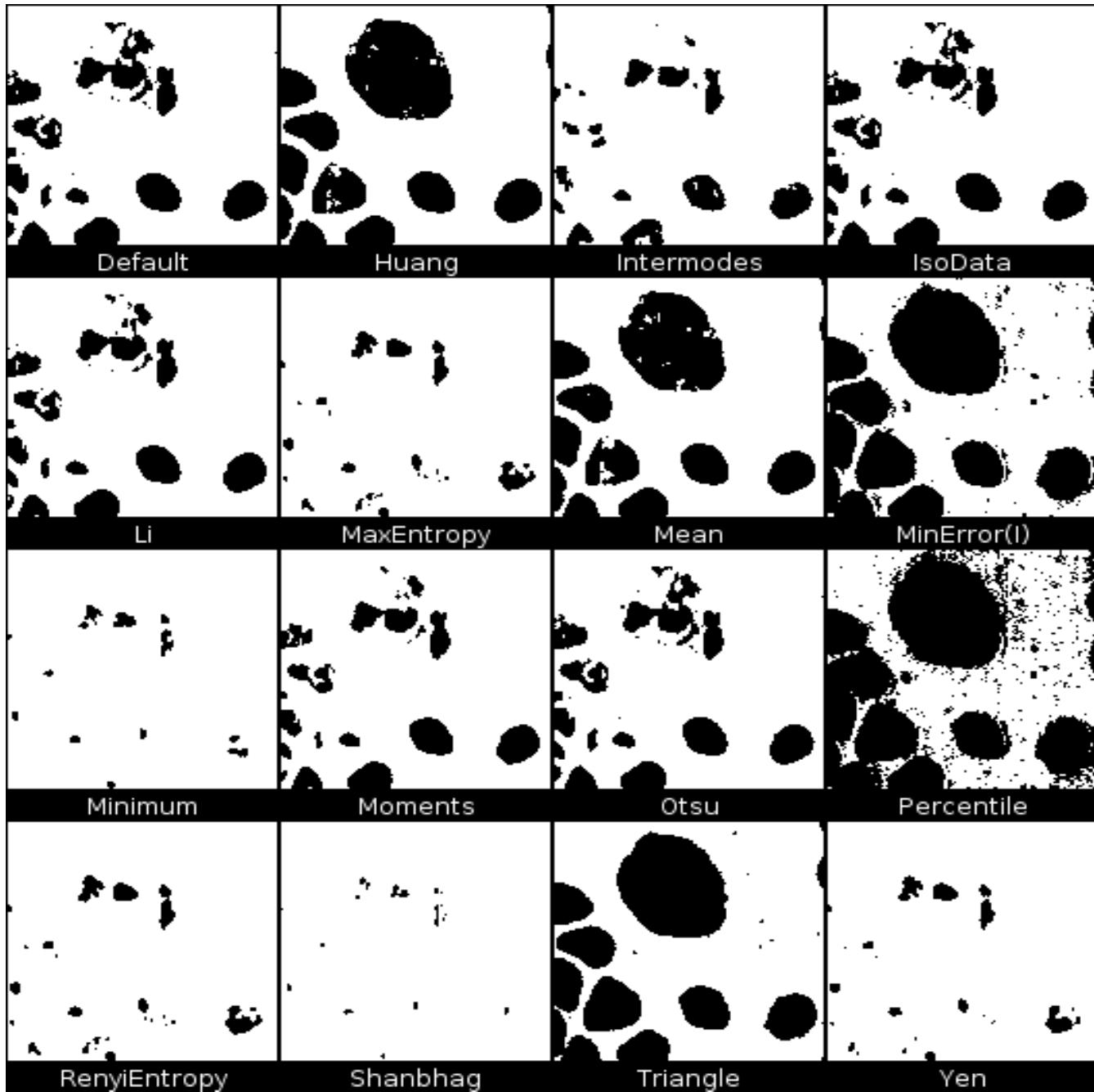
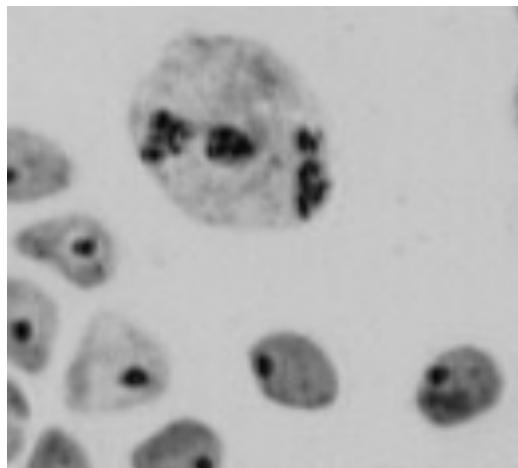


Count: 50190
Mean: 255
StdDev: 0

Min: 255
Max: 255
Mode: 255 (50190)

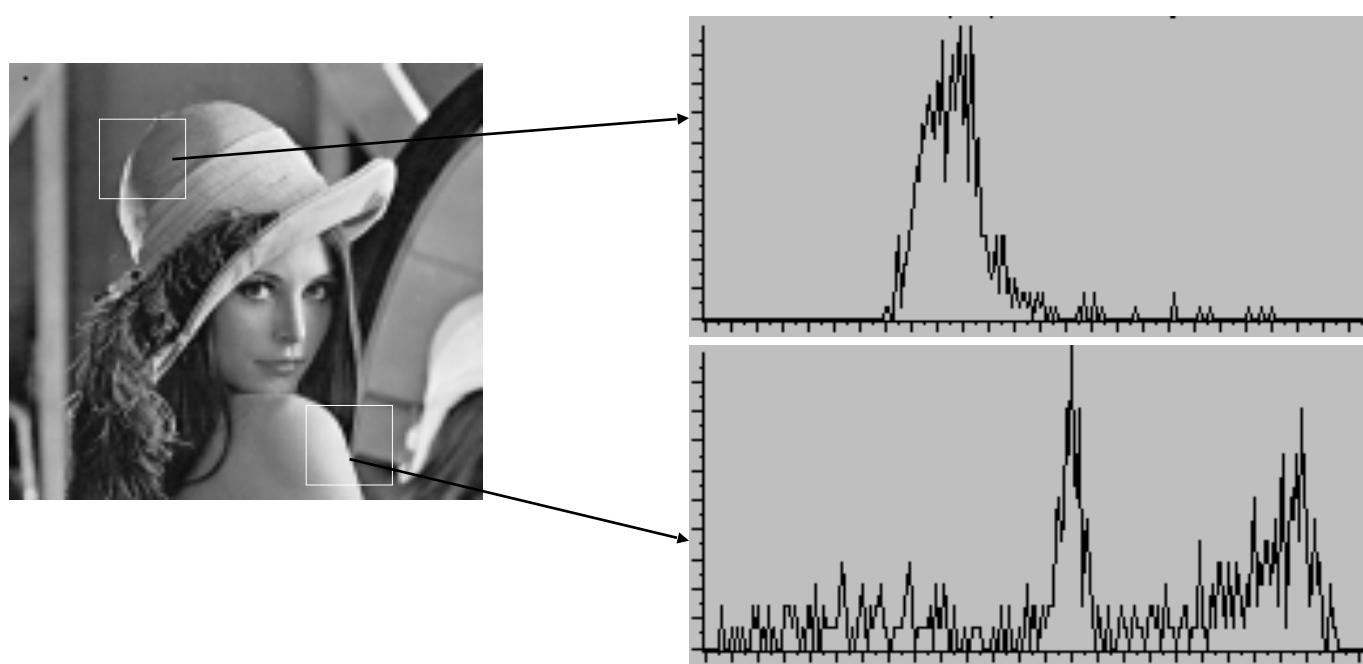
Choice of threshold

?



Examples of spatial features

Measures on a local scale :



Computation on a 16x16 scale



128x128 pixels



Mean



Variance

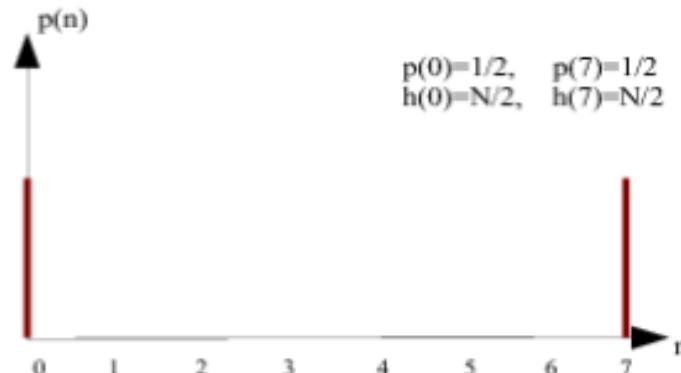
First order statistics

Satistiques du premier ordre

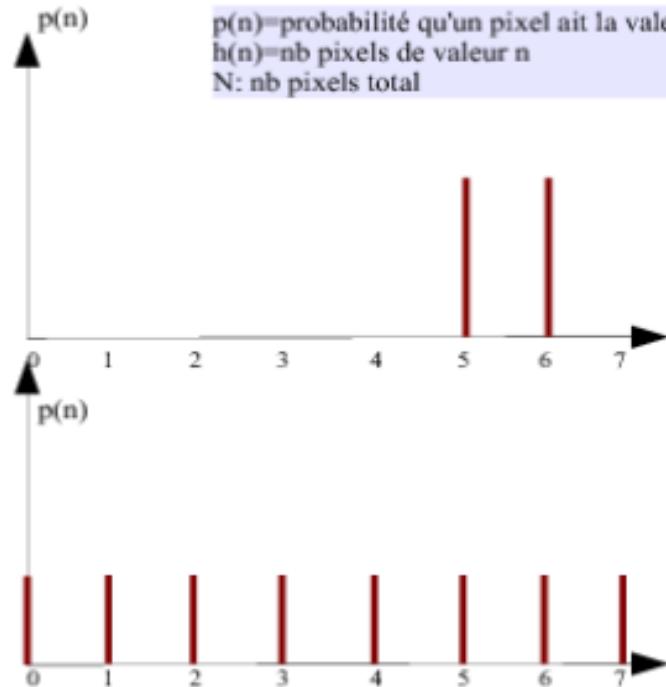
Se déduisent de la probabilité $p(n)$ du niveau de gris n ou de l'histogramme $h(n) \simeq Np(n)$, avec N le nombre de pixels de l'image.

- **Moments d'ordre k** $\mu_k = \sum_n n^k p(n)$ (μ_1 : moyenne)
 - **Moments centrés d'ordre k** $\eta_k = \sum_n (n - \mu_1)^k p(n)$
- En particulier : la variance $\sigma^2 = \eta_2$, le biais $\gamma_1 = \eta_3/\sigma^3$, le kurtosis (aplatissement)
 $\gamma_2 = \eta_4/\sigma^4 - 3$
- **Énergie** $W = \sum_n |p(n)|^2$
 - **Entropie** $E = -\sum_n p(n) \log p(n)$
 - **Contraste** $C = (\max(n) - \min(n))/(\max(n) + \min(n))$
 - **Dynamique** $D = \max(n) - \min(n)$
 - **Coefficient de variation** $\varsigma = \mu/\sigma$
 - **Dimension fractale**

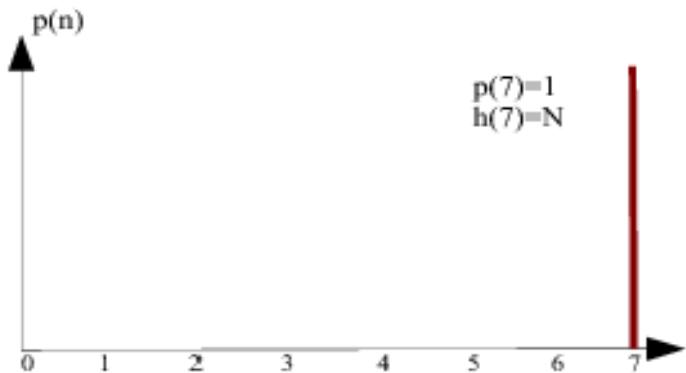
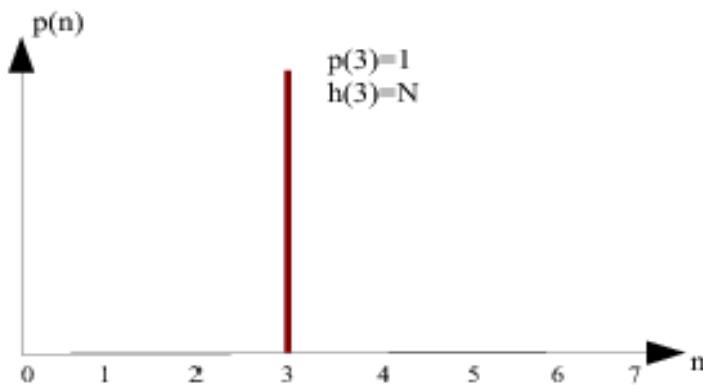
Example of first order statistics



$p(n)$ =probabilité qu'un pixel ait la valeur n
 $h(n)$ =nb pixels de valeur n
 N: nb pixels total



Example of first order statistics



Illustration

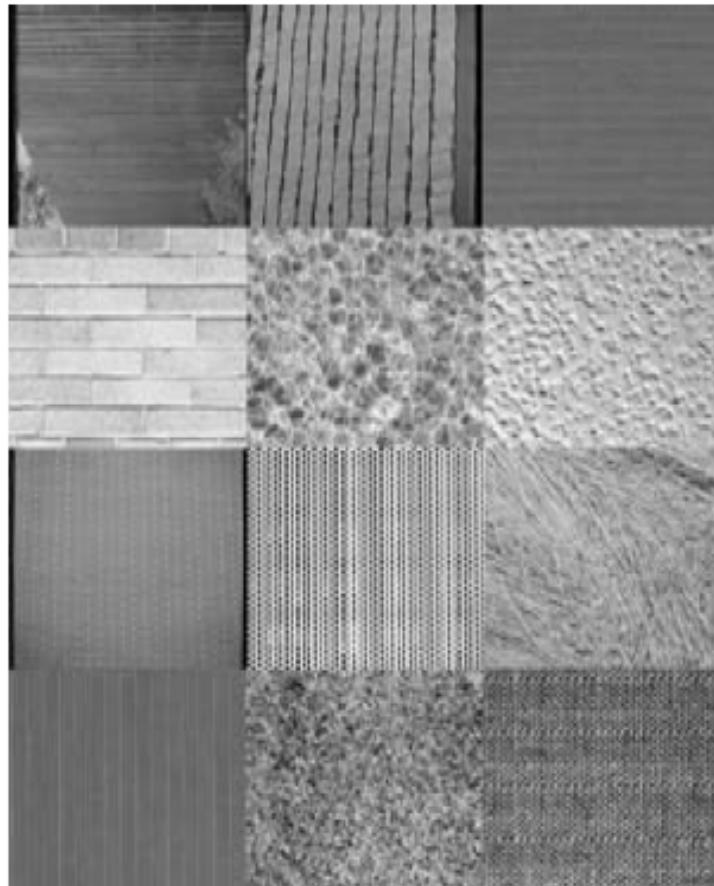
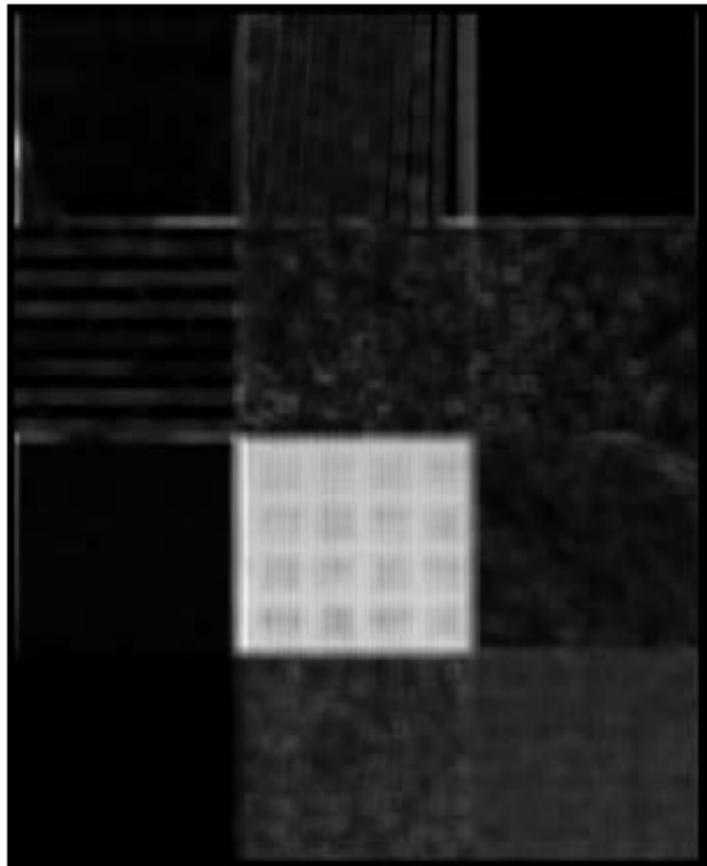


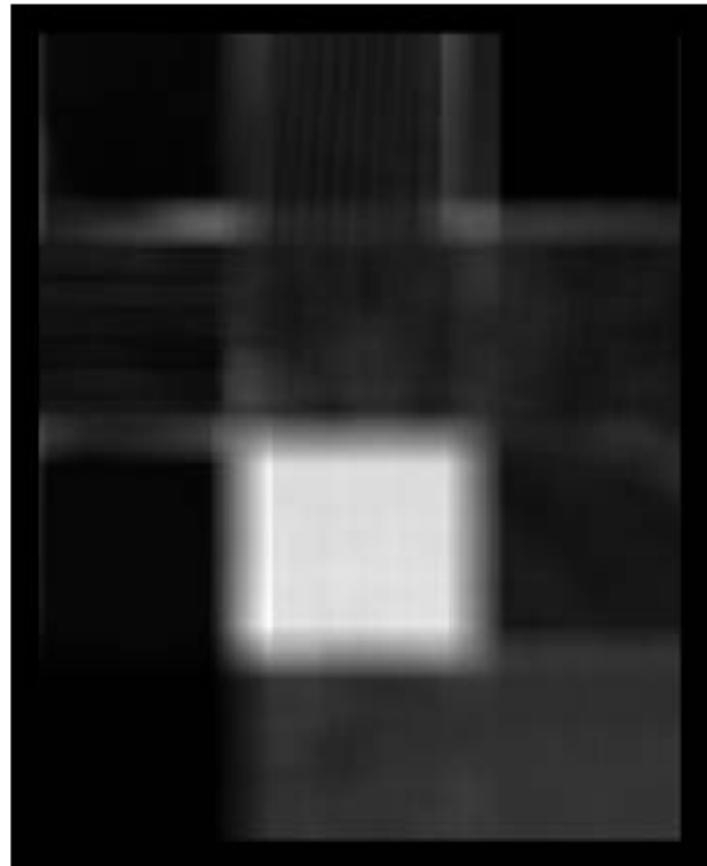
Image de résolution 1600×800

Soit W la taille de fenêtre : $W = 15$ et $W = 50$ pour les exemples.

Variance



$W=15$



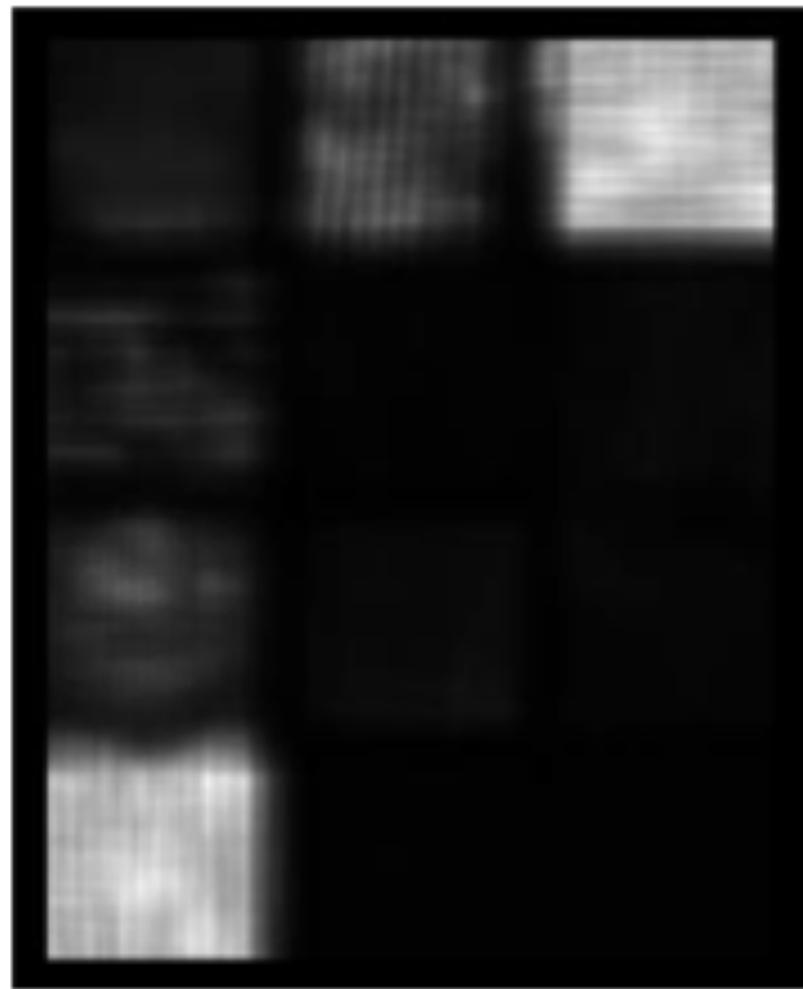
$W=50$

Variance : mesure l'ampleur des variations d'intensité par rapport à la moyenne.
Moment centré d'ordre 3 = 0 si les intensités sont équitablement réparties autour de la moyenne (histogramme symétrique autour de la moyenne)

Energy



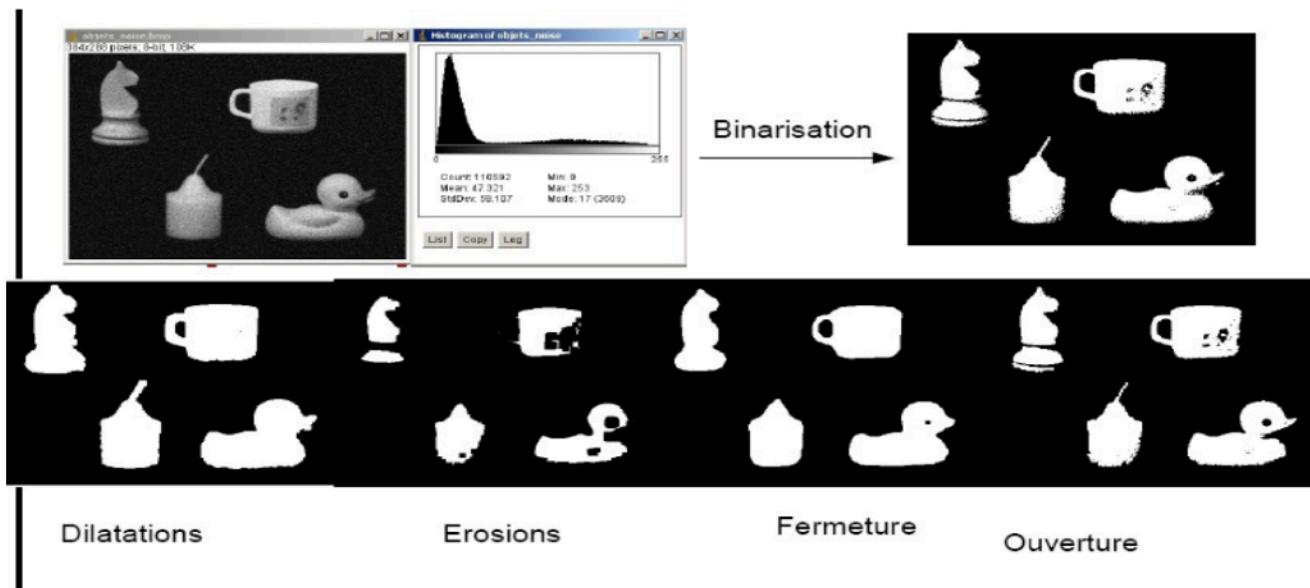
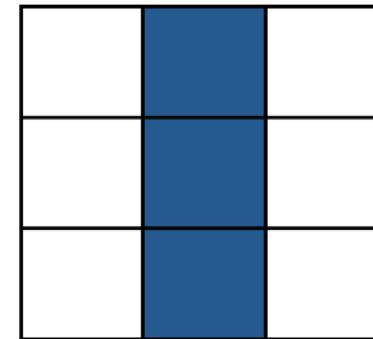
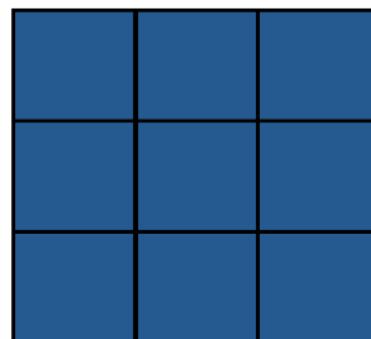
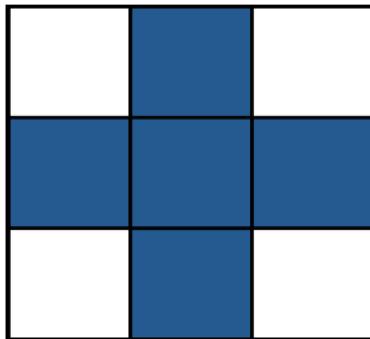
$W=15$



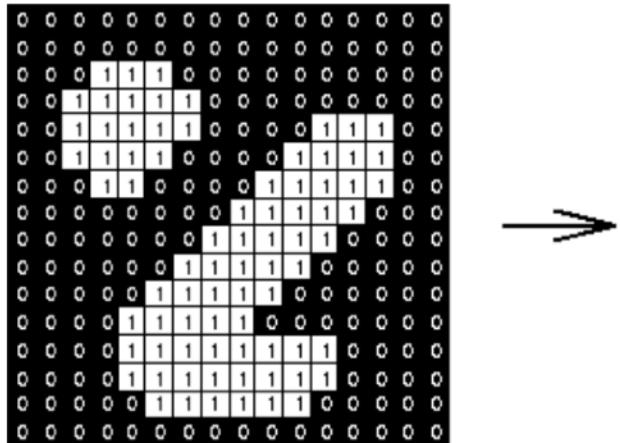
$W=50$

Mathematical morphology

- Convolution of a Structural Element (operator, filter,...)) on a binary image to highlight visual elements or filter
- Exemples

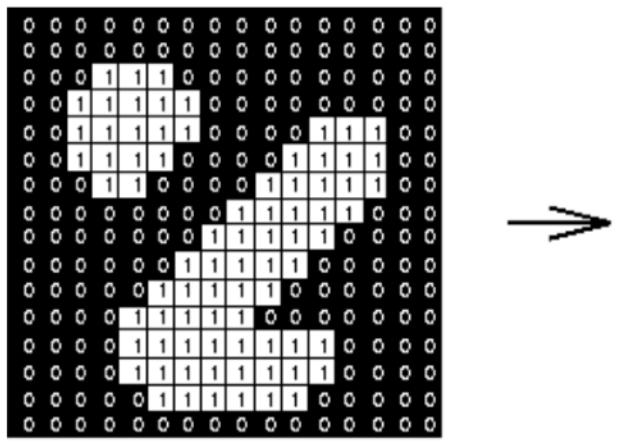


Erosion & Dilatation



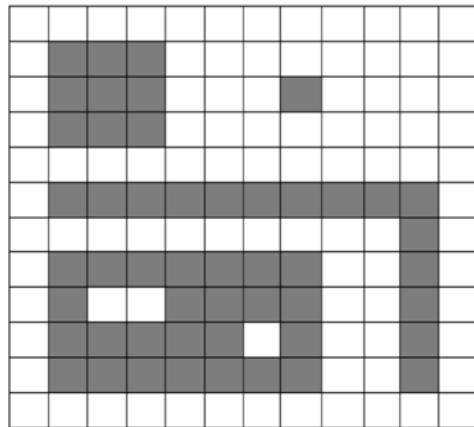
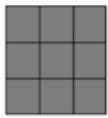
A 10x10 grid of binary digits (0s and 1s). The pattern consists of a vertical column of 1s in the center, with a diagonal line of 1s extending from the top-left towards the bottom-right. There are also a few isolated 1s scattered in the grid.

Élément structurant
carré de 3x3



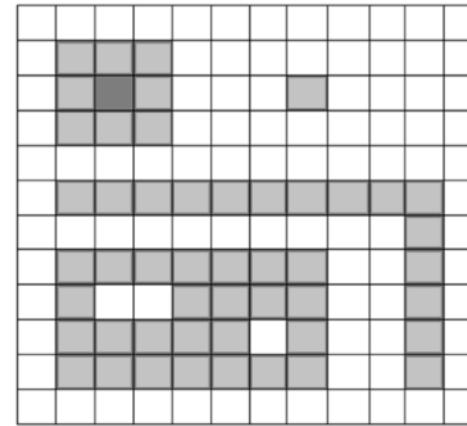
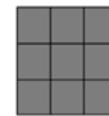
Élément structurant
carré de 3x3

Erosion



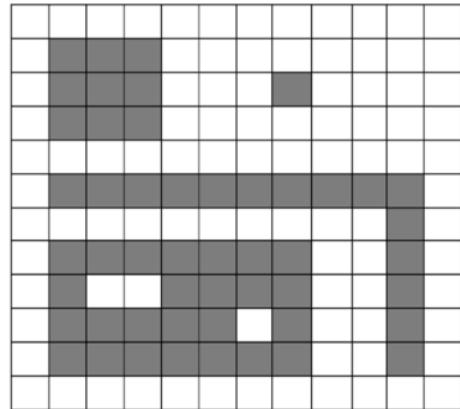
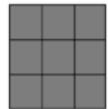
Tribute to Michele GOUIFFES

Solution



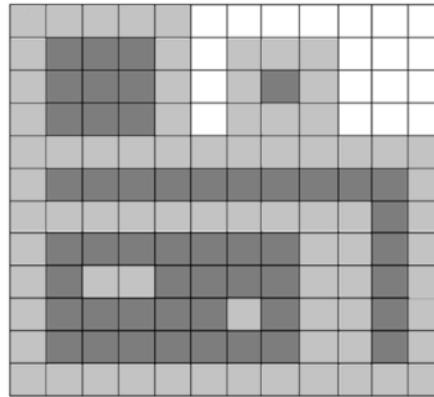
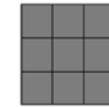
Tribute to Michele GOUIFFES

Dilatation



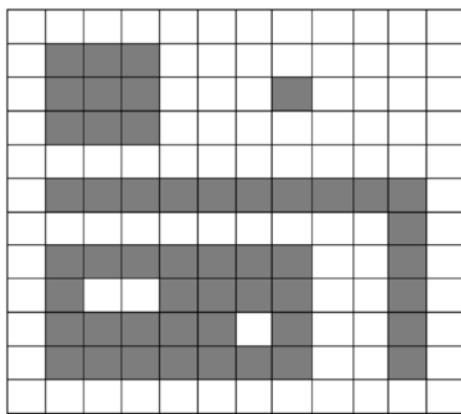
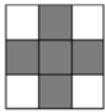
Tribute to Michele GOUIFFES

Solution

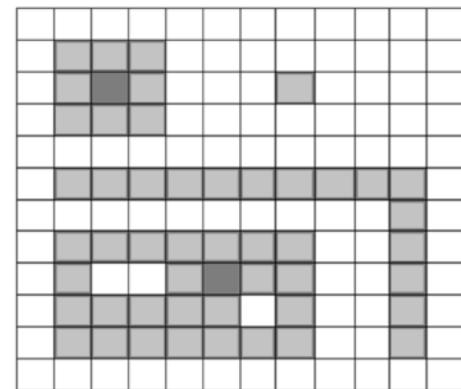
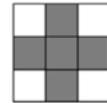


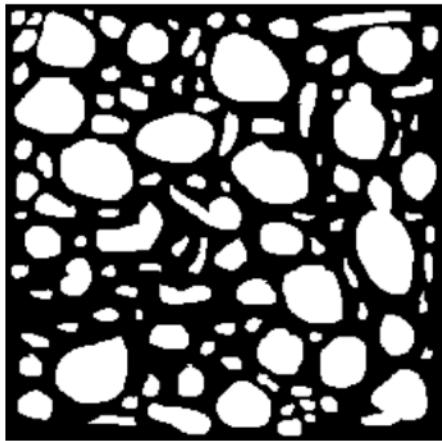
Tribute to Michele GOUIFFES

Eroding image with the 3 by 3 structuring element

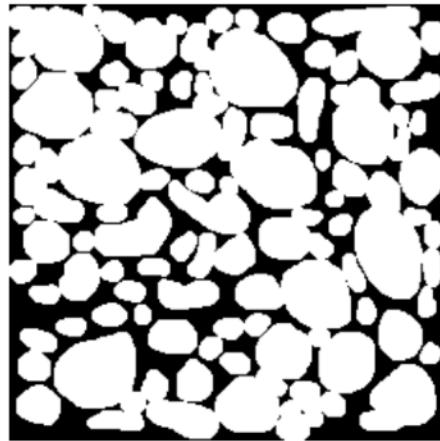


Solution

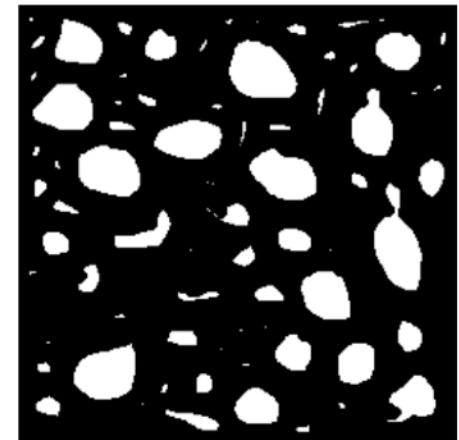




Originale



Dilatée



Erodée

- La dilatation fait disparaître les petits trous et les petits détroits, et fait grossir les objets.
- L'érosion fait disparaître les petits objets et les petits isthmes, et amincit les objets restants.

Ouverture/Fermeture

- L'ouverture constitue une érosion suivie d'une dilatation
- Cela permet de faire disparaître des petits artefacts
- Il faut que ceux-ci soient clairs sur un fond foncé
- La fermeture consiste d'une dilatation suivie d'une érosion
- Elle permet de fermer les petits trous dans une image



Mais aussi à base d'érosion/dilatation :

Boucher les trous

Calculer des cartographies de distances

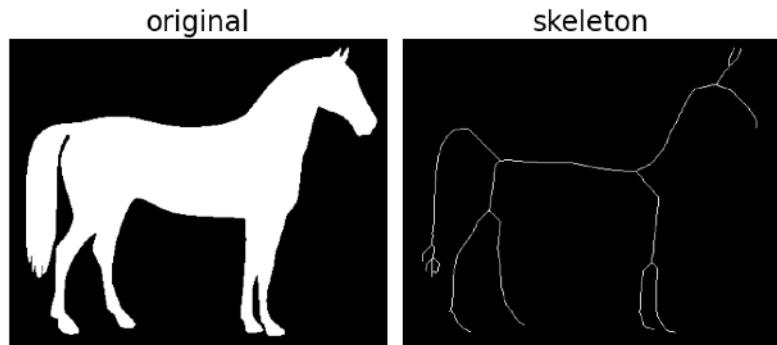
Composantes connexes

Enveloppe convexe

...

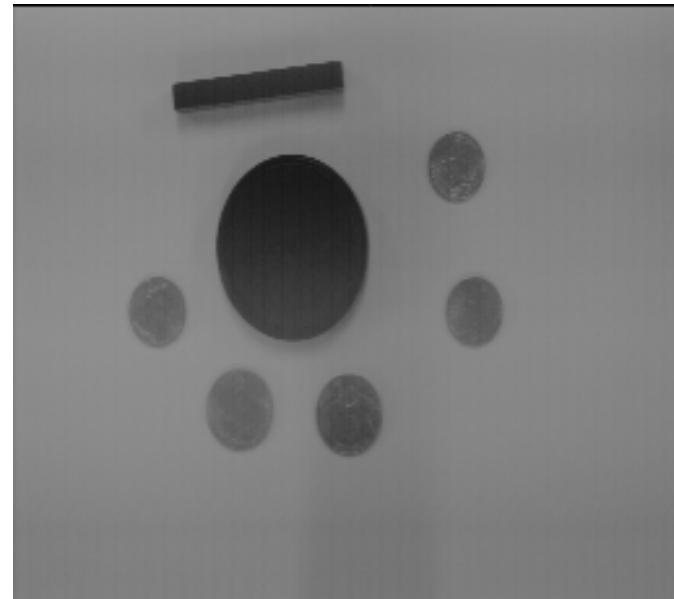
Squelettisation

Il existe d'autres opérateurs morphologiques comme la squelettisation qui consiste une succession d'erosions qui est itérée jusqu'à stabilisation en interdisant la séparation d'une région connexe et a suppression de pixels d'extremité.



HANDS ON n°8

- 1) Load morpho.bmp.
- 2) Binarize image to obtain black objects on white background
- 3) Duplicate this image
- 4) We want to erase small objects.
- 5) In menu Process/Binary/options set parameters
- 6) We have to proceed as :
 - Dilatation
 - Erosion with the same mask
 - 5 erosion with a mask at 4 neighbours



HANDS ON n°9

Exercice 1 :

1. Ouvrir BLOOD.BMP
2. Binariser (soit manuellement à l'aide de l'observation de l'histogramme soit par calcul (fct *graythresh* dans Octave))
3. Tester les opérateurs morphologiques *erode* et *dilate*. En particulier, vous essaierez de :
 - éliminer le bruit résiduel
 - boucher les cellules non bouchées... s'il y en a :)

Please feel free to try the function *imfill*

Exercice 2 :

1. Ouvrir TEXTE.BMP
2. Appliquer 1 opération de skeletization, Appliquer 5 opérations, Appliquer INF opérations de skeletization

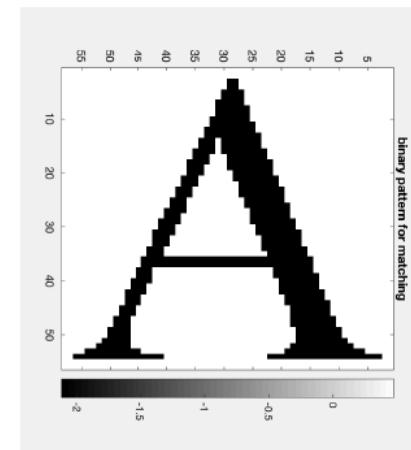
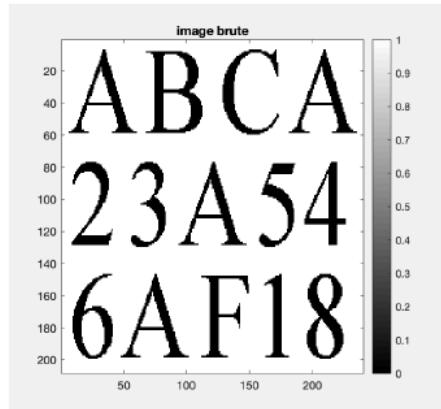
Exercice 3 :

1. Ouvrir LENA.bmp
2. Afficher image en niveau de gris
3. Calculer les gradients Dx et Dy de l'image
4. Calculer le module de l'image des gradients D
5. Binariser l'image D des gradients
6. Calculer les laplaciens et Lx et Ly puis calculer L(x,y)
7. Vérifier que le masque [0 -1 0;-1 4 -1;0 -1 0] donne la même chose ?
8. Tester la détection de contour avec function *edge* utilisant le filtre de Roberts. Comparer avec résultat de 5.
9. Tester les filtres Sobel, Prewitt, Canny, Log,...
10. Ajouter un bruit gaussien à votre image (moy = 0, var = 0.05) et tester les filtres « déivateurs + moyenneurs » tel que Sobel.
11. Ajouter un bruit impulsionnel à votre image (var = 0.01) et tester les filtres « déivateurs + moyenneurs » tel que Sobel.
12. Filtrer d'abord avec les méthodes que vous connaissez puis lancer une détection de contour.
13. Conclure

HANDS ON n°10

Détection de forme par corrélation.

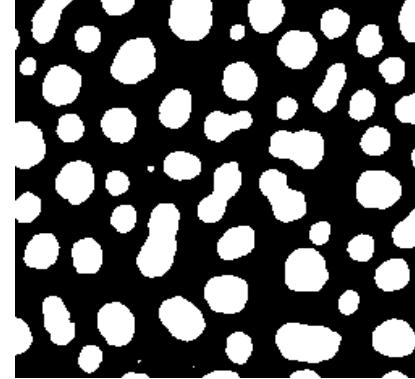
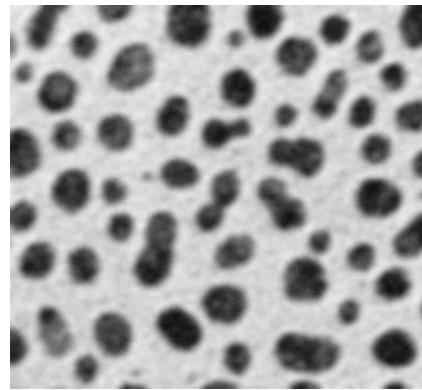
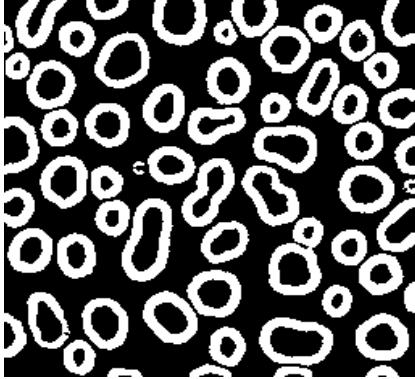
L'idée ici consiste à filter l'image par un filtre (ici une imagette) contenant le pattern / la forme que l'on cherche.



1. Ouvrir image « abcda.png »
2. Se définir par la méthode que vous voulez le pattern « A »
3. Centrer et normaliser l'imagette « A » et l'image « abcda »
4. Réaliser une opération de corrélation (i.e. filtrage de l'image par l'imagette) pour détecter la position de la lettre A dans l'image.
5. Faire la même chose avec une image bruitée type sel et poivre ($v=0.75$)

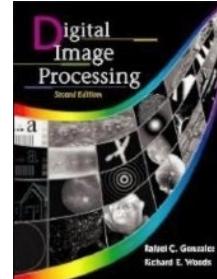
Chapitre 5

Représentation, description et analyse pour la reconnaissance de formes / shapes recognition (Haut-niveau)

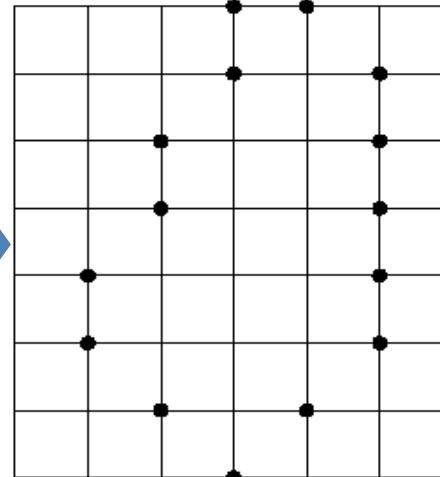
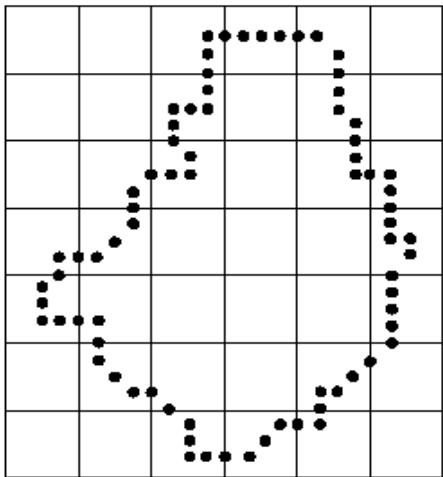


Methods to:

- **describe shapes (boundary or region)**
- **looking for invariance in translation, rotation, homothety**
- with features/attributes.
- These can be viewed as external (boundary) or internal (region) characteristics, respectively.

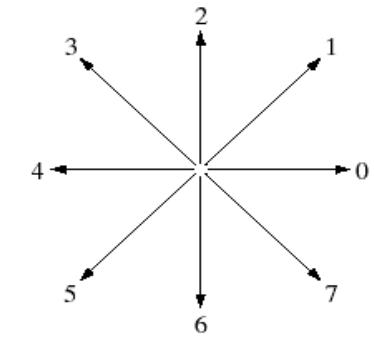
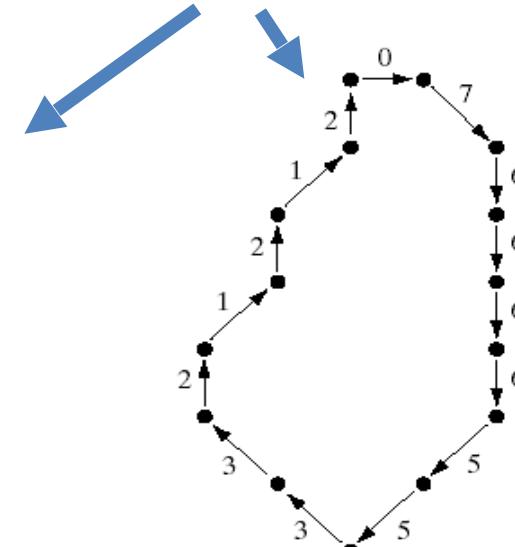
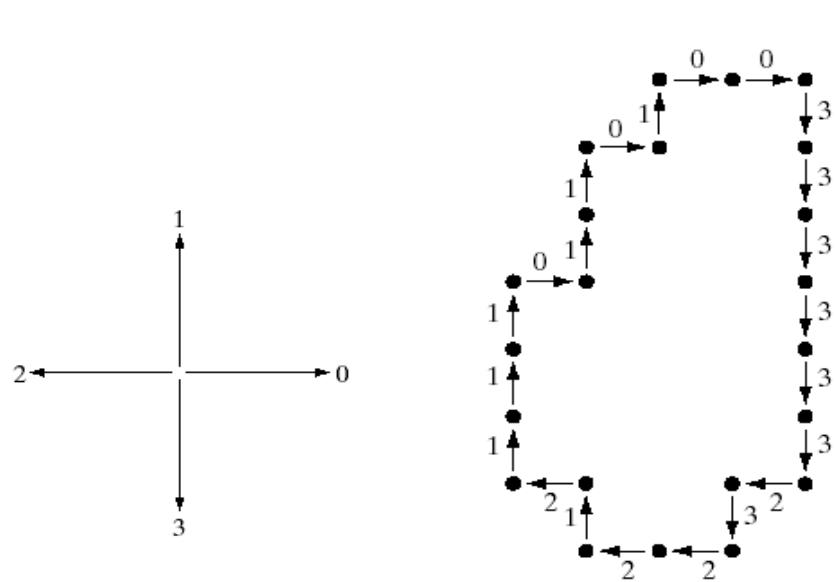


Chain codes: represent a boundary of a connected region.



a	b
c	d

FIGURE 11.2
(a) Digital boundary with resampling grid superimposed.
(b) Result of resampling.
(c) 4-directional chain code.
(d) 8-directional chain code.



Chain codes: represent a boundary of a connected region.

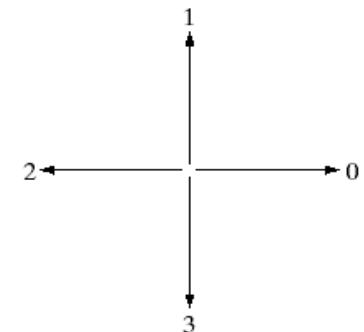
Chain codes can be based on either 4-connectedness or 8-connectedness.

The first difference of the chain code:

- This difference is obtained by counting the number of direction changes (in a counterclockwise direction)
- For example, the first difference of the 4-direction chain code 10103322 is 3133030.

Assuming the first difference code represent a closed path, rotation normalization can be achieved by circularly shifting the number of the code so that the list of numbers forms the smallest possible integer.

Size normalization can be achieved by adjusting the size of the resampling grid.



Polygonal approximation

Polygonal approximations: to represent a boundary by straight line segments, and a closed path becomes a polygon.

The number of straight line segments used determines the accuracy of the approximation.

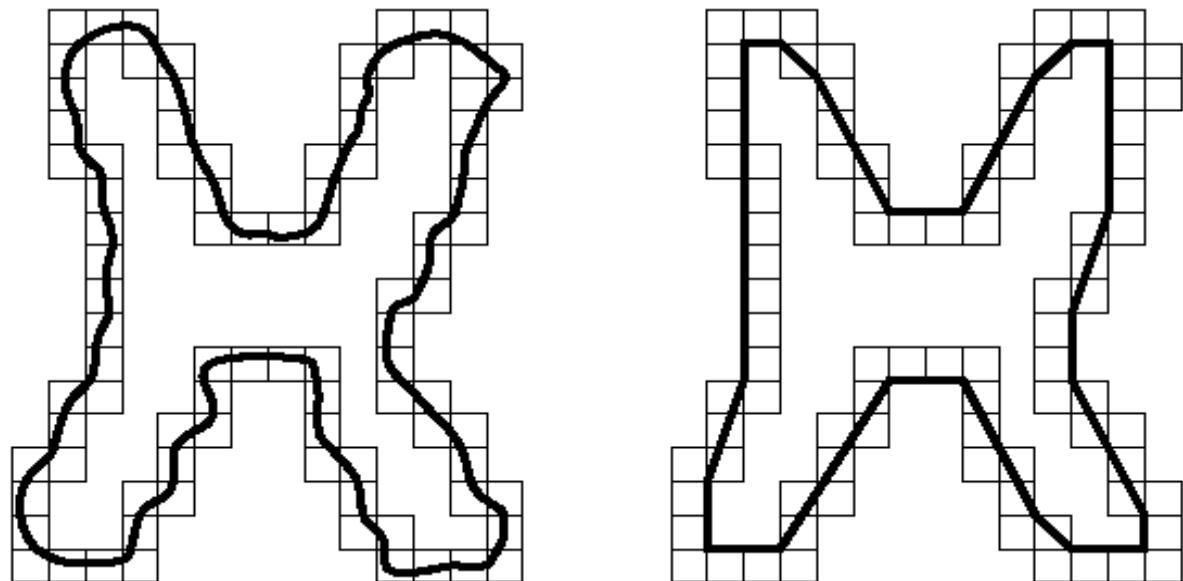
Only the minimum required number of sides necessary to preserve the needed shape information should be used (Minimum perimeter polygons).

A larger number of sides will only add noise to the model.

a b

FIGURE 11.3

- (a) Object boundary enclosed by cells.
(b) Minimum perimeter polygon.

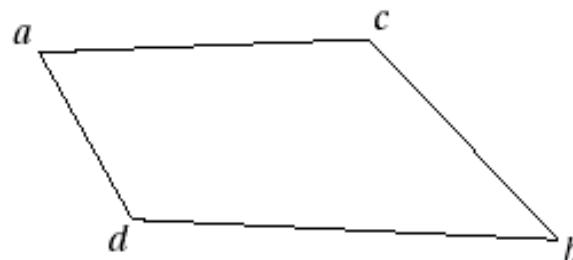
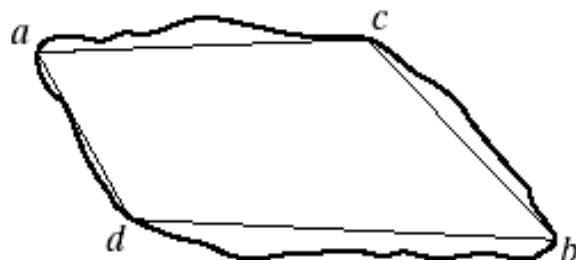
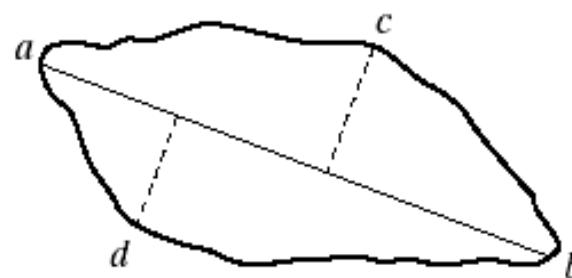
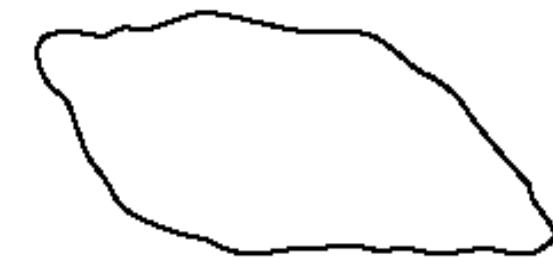


Polygonal approximation

Minimum perimeter polygons: (Merging and splitting)

Merging and splitting are often used together to ensure that vertices appear where they would naturally in the boundary.

A least squares criterion to a straight line is used to stop the processing.



a	b
c	d

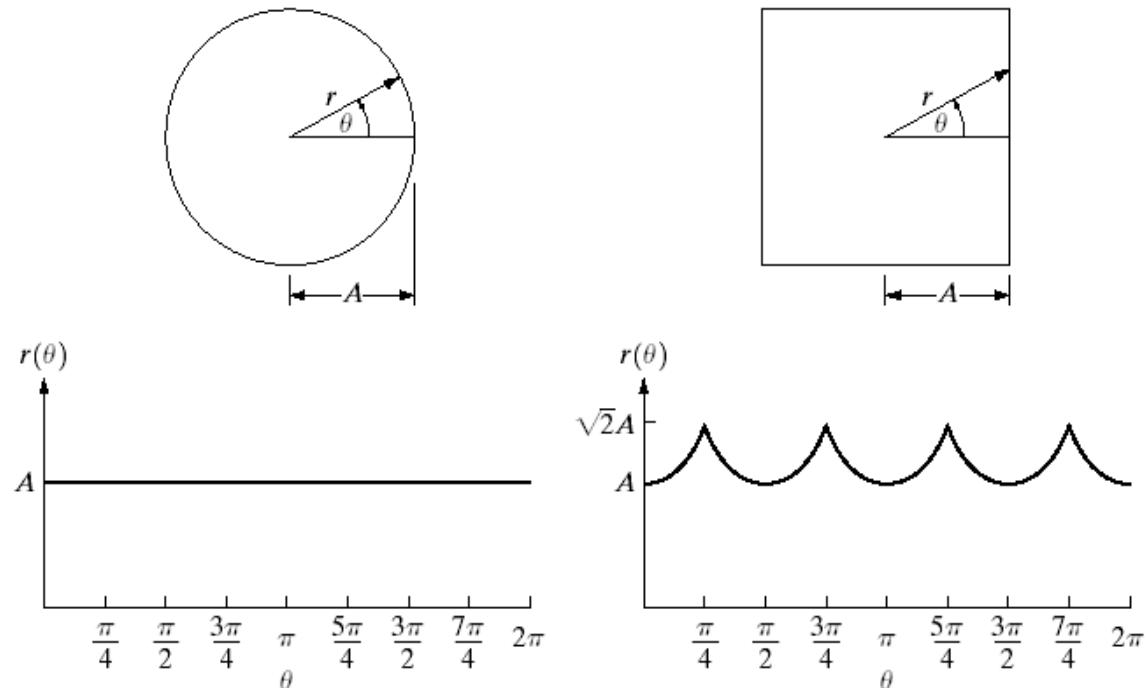
FIGURE 11.4
(a) Original boundary.
(b) Boundary divided into segments based on extreme points. (c) Joining of vertices.
(d) Resulting polygon.

Signature

The idea behind a signature is to convert a two dimensional boundary into a representative one dimensional function. Signatures are invariant to location, but will depend on rotation and scaling.

a b

FIGURE 11.5
Distance-versus-angle signatures.
In (a) $r(\theta)$ is constant. In (b),
the signature
consists of
repetitions of the
pattern
 $r(\theta) = A \sec \theta$ for
 $0 \leq \theta \leq \pi/4$ and
 $r(\theta) = A \csc \theta$ for
 $\pi/4 < \theta \leq \pi/2$.

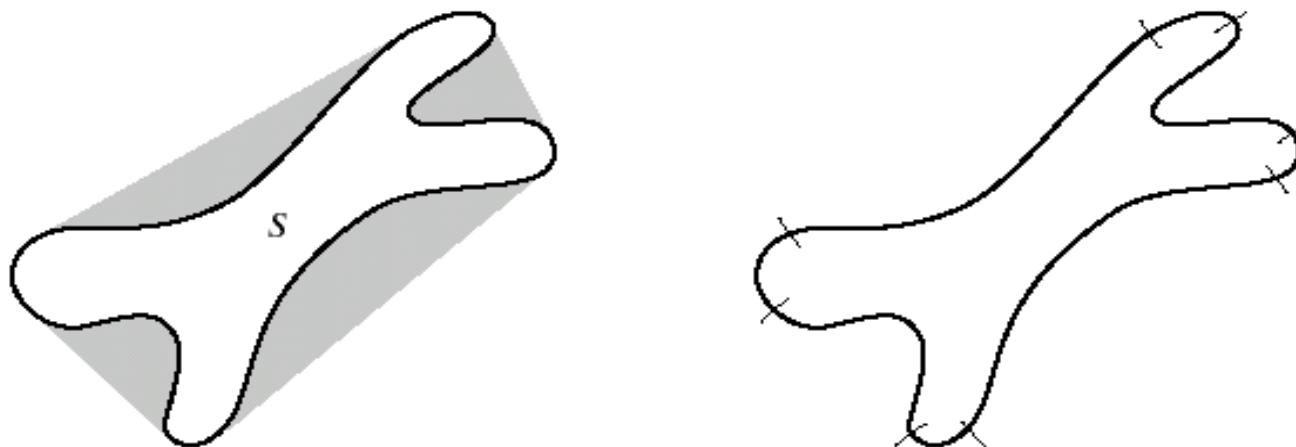


Starting at the point farthest from the reference point or using the major axis of the region can be used to decrease dependence on rotation.

Scale invariance can be achieved by either scaling the signature function to fixed amplitude or by dividing the function values by the std of the function.

Boundary segments

Boundary segments: decompose a boundary into segments. Use of the convex hull of the region enclosed by the boundary is a powerful tool for robust decomposition of the boundary.

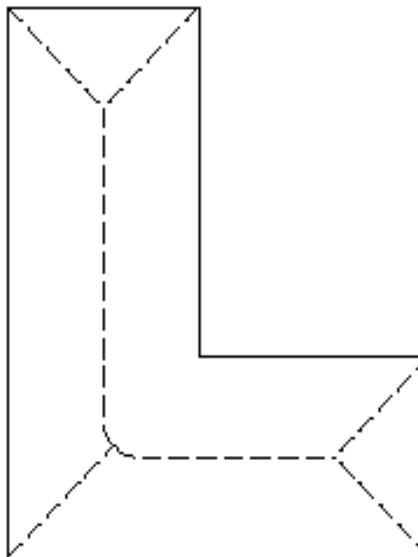
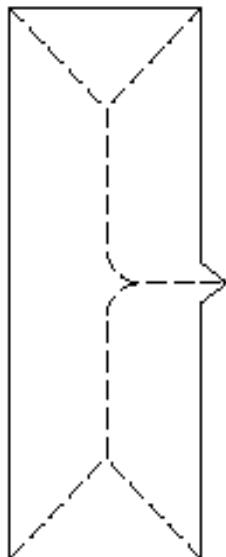
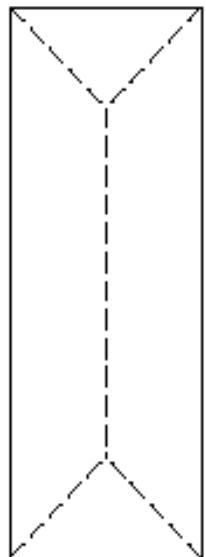


a b

FIGURE 11.6
(a) A region, S ,
and its convex
deficiency
(shaded).
(b) Partitioned
boundary.

Skeleton

Skeletons: produce a one pixel wide graph that has the same basic shape of the region, like a stick figure of a human. It can be used to analyze the geometric structure of a region which has bumps and “arms”.



a b c

FIGURE 11.7
Medial axes
(dashed) of three
simple regions.

Boundary descriptors

There are several simple geometric measures that can be useful for describing a boundary.

The length of a boundary: the number of pixels along a boundary gives a rough approximation of its length → Perimeter

Curvature: the rate of change of slope. To measure a curvature accurately at a point in a digital boundary is difficult. The difference between the slopes of adjacent boundary segments is used as a descriptor of curvature at the point of intersection of segments

Boundary descriptors: Fourier descriptors

This is a way of using the Fourier transform to analyze the shape of a boundary. The x-y coordinates of the boundary are treated as the real and imaginary parts of a complex number → $u_k = x_k + iy_k$

Then the list of coordinates is Fourier transformed using the FFT. The Fourier coefficients are called the Fourier descriptors.

The basic shape of the region is determined by the first several coefficients, which represent lower frequencies. Higher frequency terms provide information on the fine detail of the boundary.

- Avec les N points u_k , on peut en calculer la TFD :
$$f_l = \sum_{k=0}^{N-1} u_k e^{-j2\pi kl/N}$$
- Les f_l sont les descripteurs de Fourier du contour. Ce sont des nombres complexes.
- Par TFD inverse, on peut reconstruire le contour.

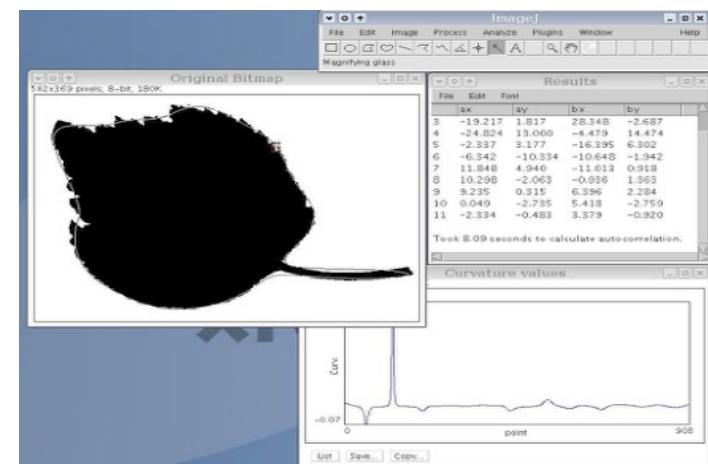
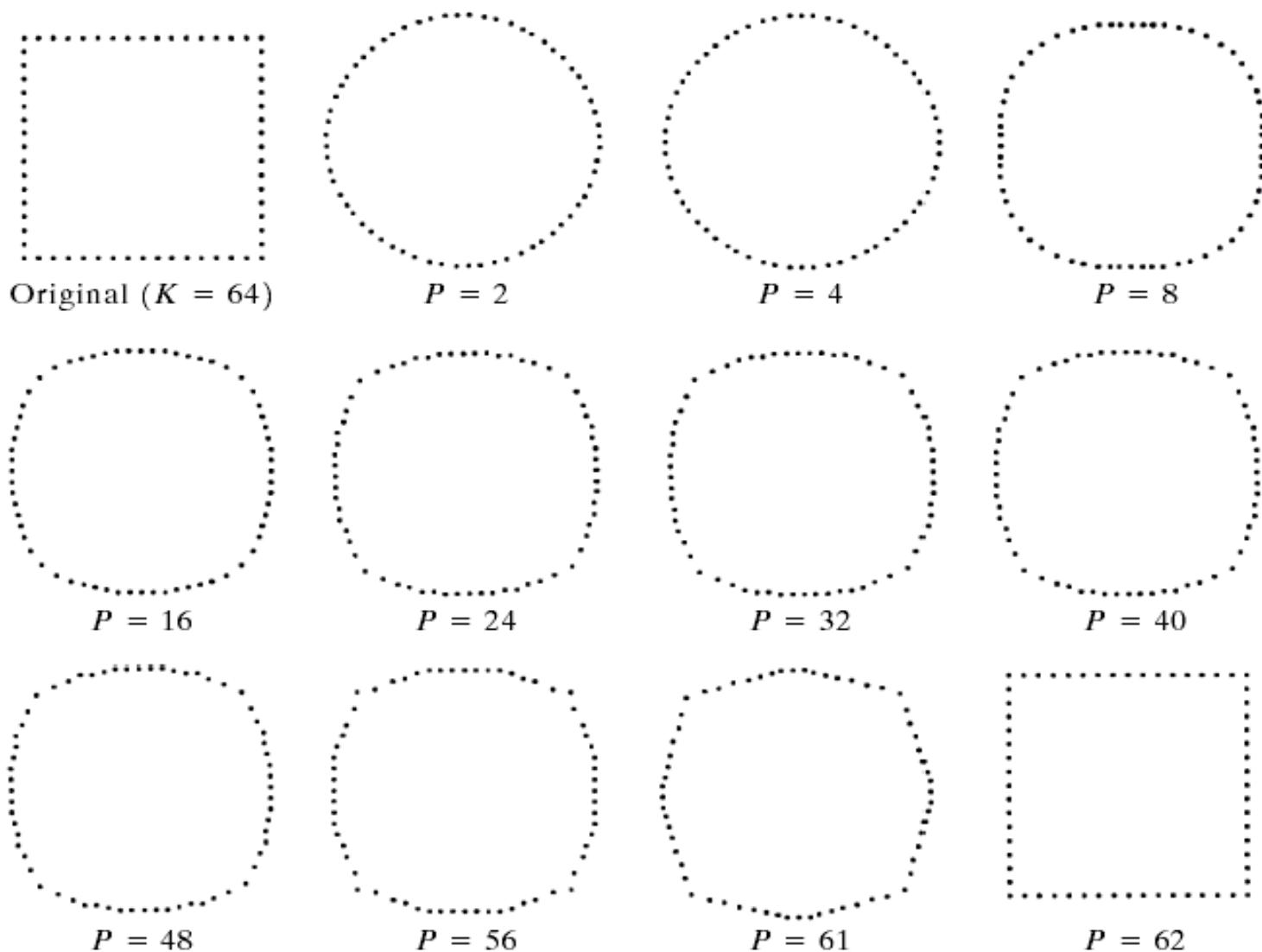


Image J → Plugin Fourier_.jar

Boundary descriptors: Fourier descriptors

FIGURE 11.14
Examples of reconstruction from Fourier descriptors. P is the number of Fourier coefficients used in the reconstruction of the boundary.



Region descriptors

Some simple descriptors

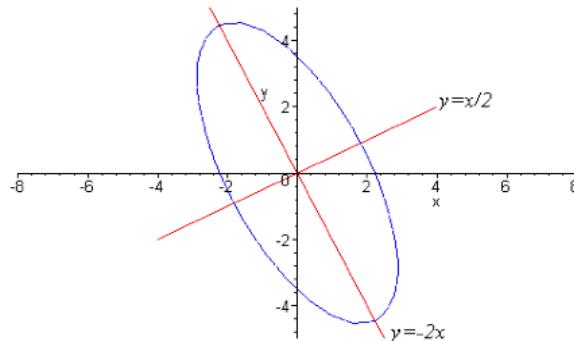
- The area of a region: the number of pixels in the region
- The perimeter of a region: the length of its boundary
- The compactness of a region: $(\text{perimeter})^2/\text{area}$
- The mean and median of the gray levels
- The minimum and maximum gray-level values
- The number of pixels with values above and below the mean

Region descriptors: eccentricity

Eccentricity is the measure of aspect ratio

It's ratio of length of major axis to minor axis (think ellipse for example)

Calculated by principal axes method (PCA) or minimum bounding rectangularbox



Minimum bounding rectangle (minimum bounding box):

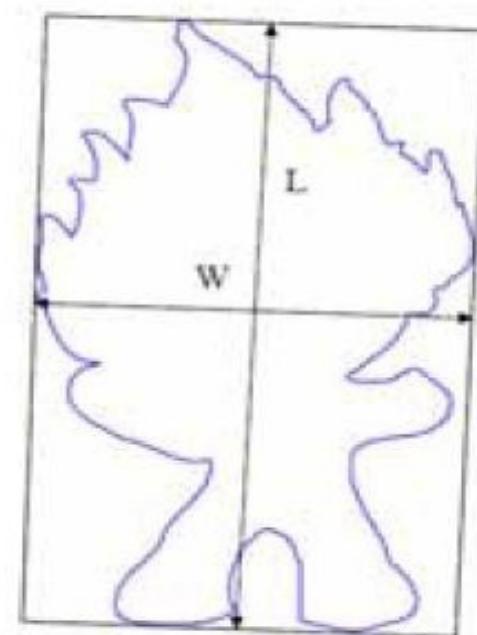
Smallest rectangle containing every pt. in the shape

- **Eccentricity:** $E=L/W$

L: length of bounding box W: width of bounding box

- **Elongation:** $Elo=1 - W/L$ Elo 2 [0,1]

Circle of square (symmetric): $Elo=0$ Shape w/ large aspect ratio: Elo close to 1



Region descriptors: circularity ratio

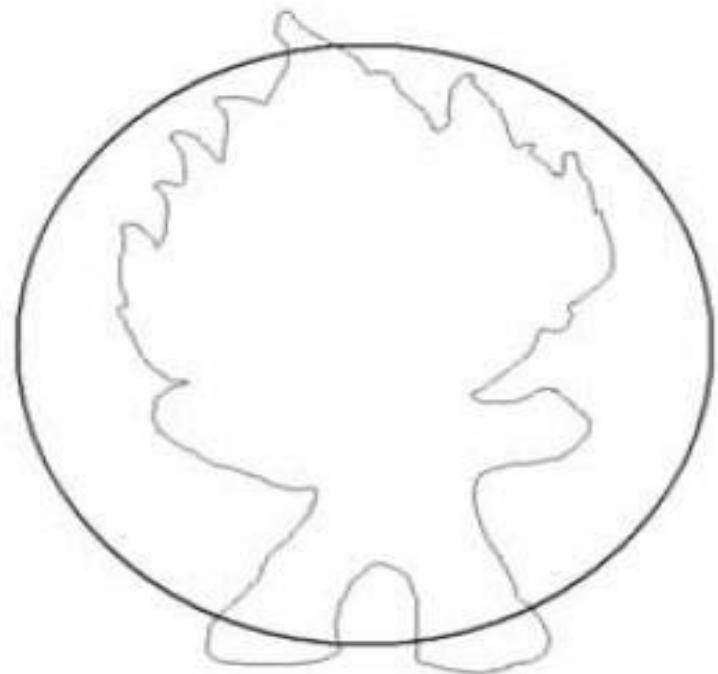
Circularity ratio: How similar to a circle is the shape
3 definitions:

Circularity ratio 1: $C_1 = A_s/A_c = (\text{Area of a shape})/(\text{Area of circle})$

where circle has the same perimeter

Circularity ratio 2: $C_2 = A_s/p^2$ ($p = \text{perim of shape}$)

Area to squared perimeter ratio.



Region descriptors: rectangularity

Rectangularity represents how rectangular a shape is, i.e. how much it fills its minimum bounding rectangle:

$$\text{Rectangularity} = A_S/A_R$$

where A_S is the area of a shape; A_R is the area of the minimum bounding rectangle.

What is rectangularity for a square? Circle? Ellipse?

Region descriptors: convexity

2.8 Convexity

Convexity is defined as the ratio of perimeters of the convex hull $\mathcal{O}_{Convexhull}$ over that of the original contour \mathcal{O} [7]:

$$Convexity = \frac{\mathcal{O}_{Convexhull}}{\mathcal{O}} \quad (13)$$

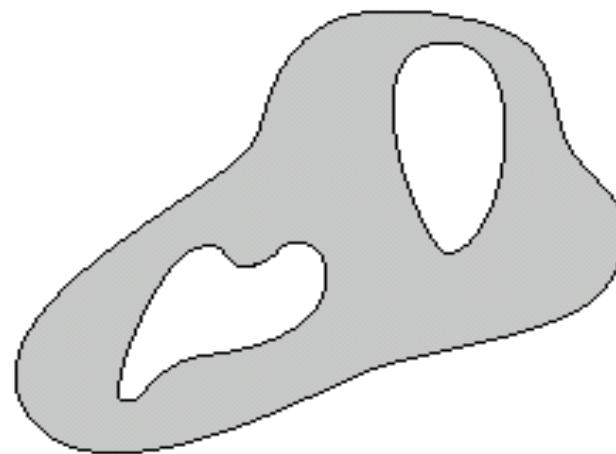
The region R^2 is a convex if and only if for any two points $P_1, P_2 \in R^2$, the entire line segment P_1P_2 is inside the region. The convex hull of a region is the smallest convex region including it. In Figure 6, the outline is the convex hull of the region.



Illustration of convex hull

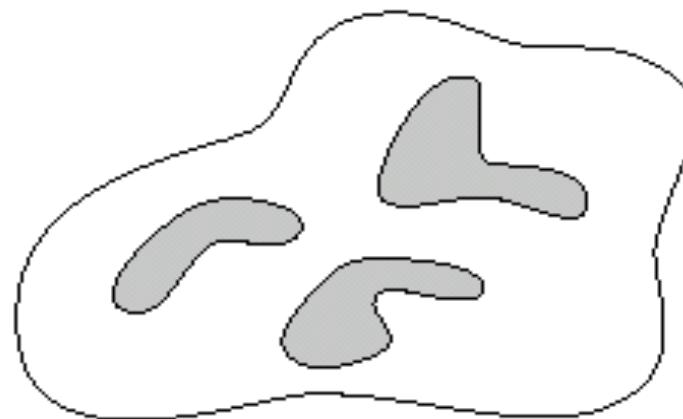
Region descriptors: topological descriptors

Topological property 1:
the number of holes
(H)



A region with two holes.

Topological property 2: the number
of
connected components (C)

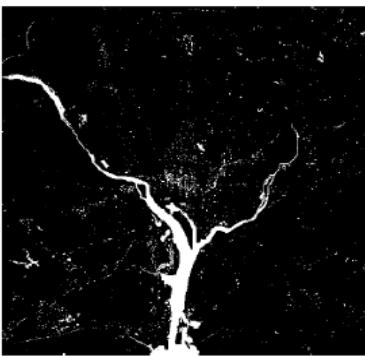
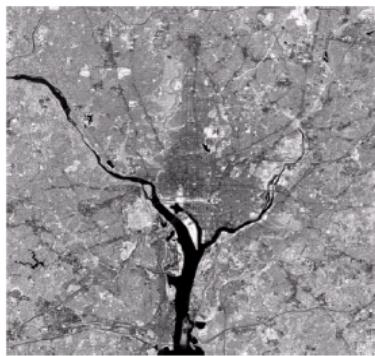


A region with three connected components.

Region descriptors: topological descriptors

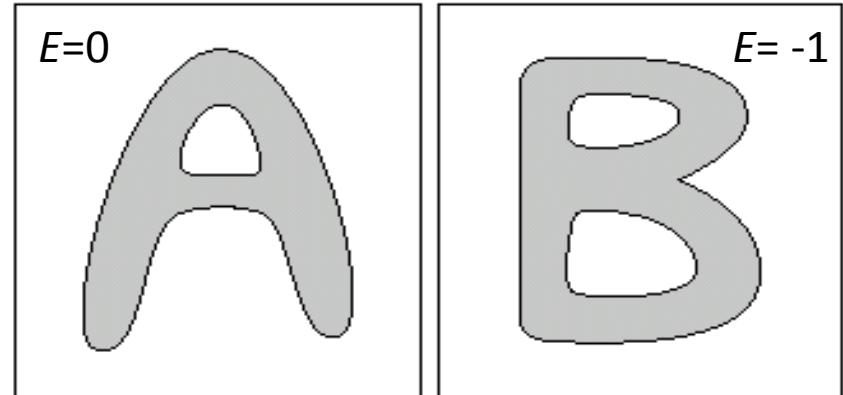
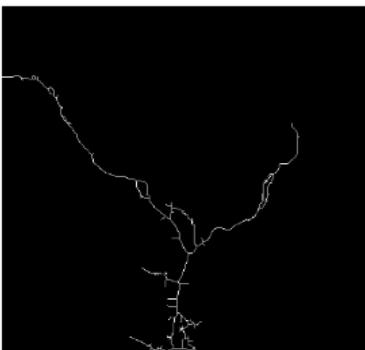
Topological property 3:

Euler number: the number of connected components subtract the number of holes $E = C - H$



a
b
c
d

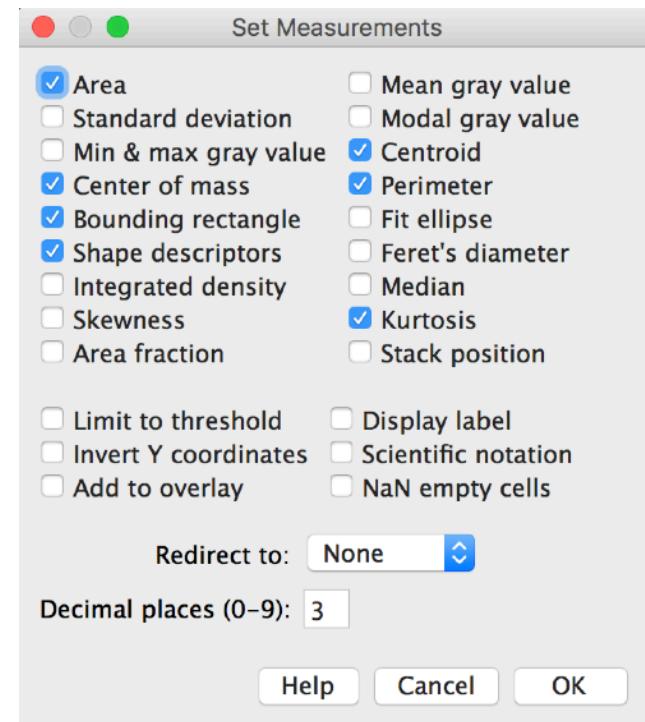
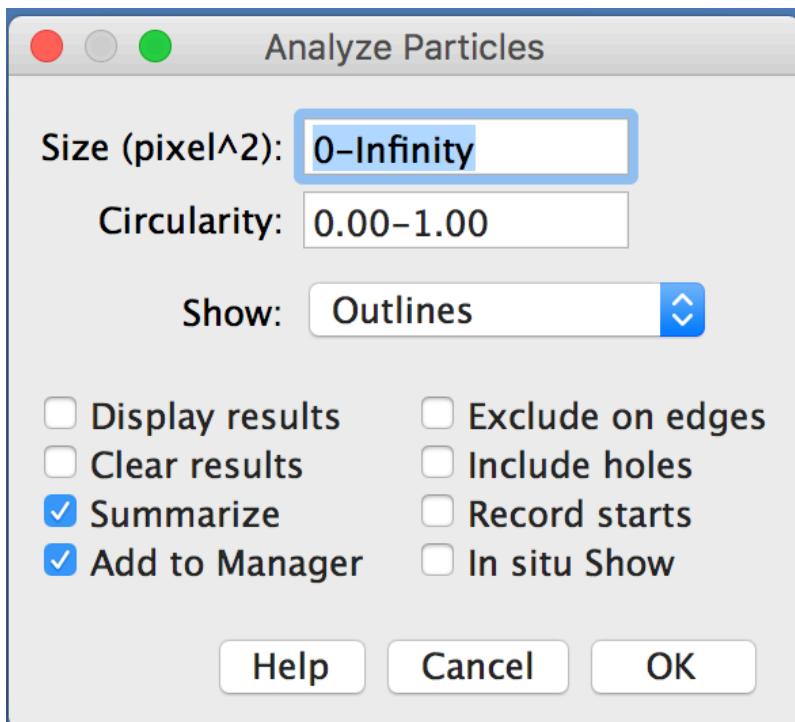
FIGURE 11.21
(a) Infrared image of the Washington, D.C. area.
(b) Thresholded image. (c) The largest connected component of (b). Skeleton of (c).



Topological property 4:
the largest connected component.

Simple descriptors with Fiji

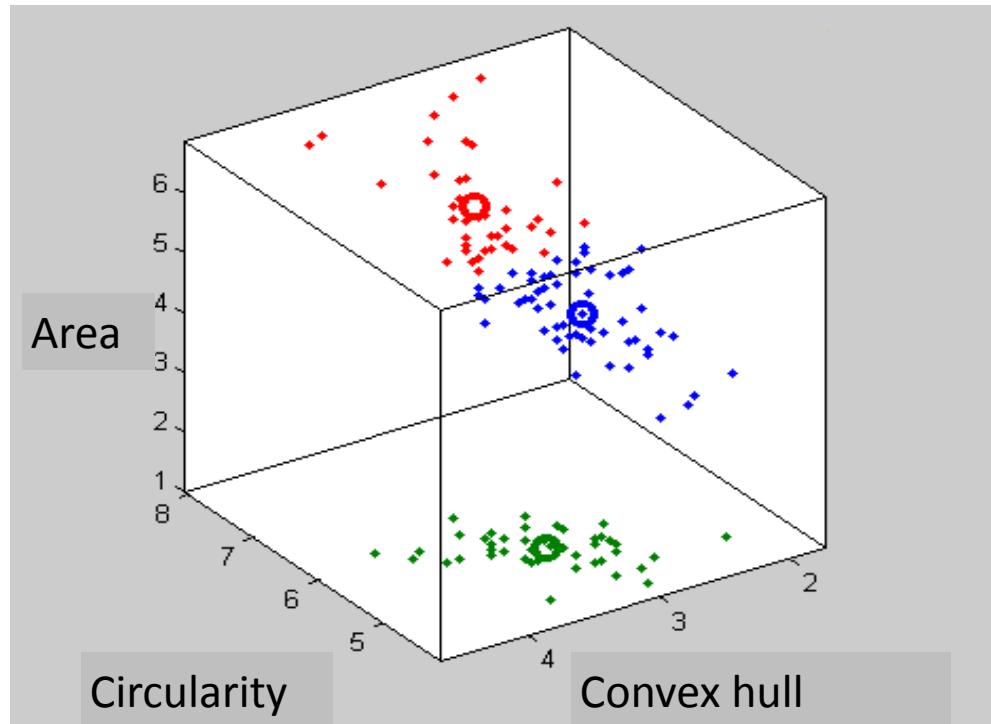
- Before any analysis he is necessary to set measures to perform. ImageJ has a wide range of settings in the menu Analysis> Set Measurements
- The function Analyze >Measure performs the measurements and displays the result in a table ("Results" window) that is possible to export by File>Save as or simply copy and paste.
- The "Particles Analysis" Analyze>Analyze Particles must be used sure a binary image or an image 8-bit thresholded by Image Feature> Adjust> Threshold.



Conclusion

The shapes recognition will consist in:

- choosing the best descriptors/features for a given application,
- and to use classifier on these descriptors/features to recognize objects in the descriptors/features space



Easy when small dimensions... But if dimensions increase, human capacity is overwhelmed → ML & DL

See for instance D. Rousseau course on ML & DL)

<https://www.youtube.com/watch?v=Rb-4Yu-qLR4> and other videos on the same channel.

HANDS ON n°11

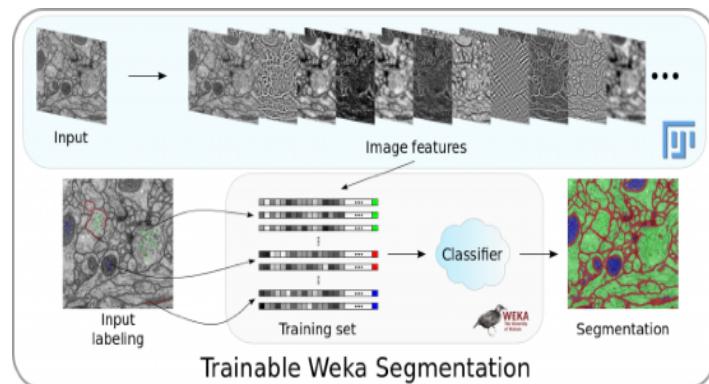
Attributs morphométriques

1. Ouvrir centimRGB.bmp
2. Calculer les descripteurs morphométriques suivants : aire, périmètre, centre de gravité, compacité pour toutes les pièces.

Axes principaux

1. Ouvrir Holmes_nb.bmp
2. Binariser.
3. Trouver le centroid, la longueur et l'orientation du major axis.
4. Tracer le contour.
5. Tracer l'axe sur le contour. Relever « à la main » les 2 intersections de l'axe avec le contour.

D'abord sans puis avec WEKA (random forest classifier)



1. Ouvrir seedsincoat.bmp.
2. Donner le nombre de graines (objet marron dans le filet beige...) afin d'en évaluer la surface.

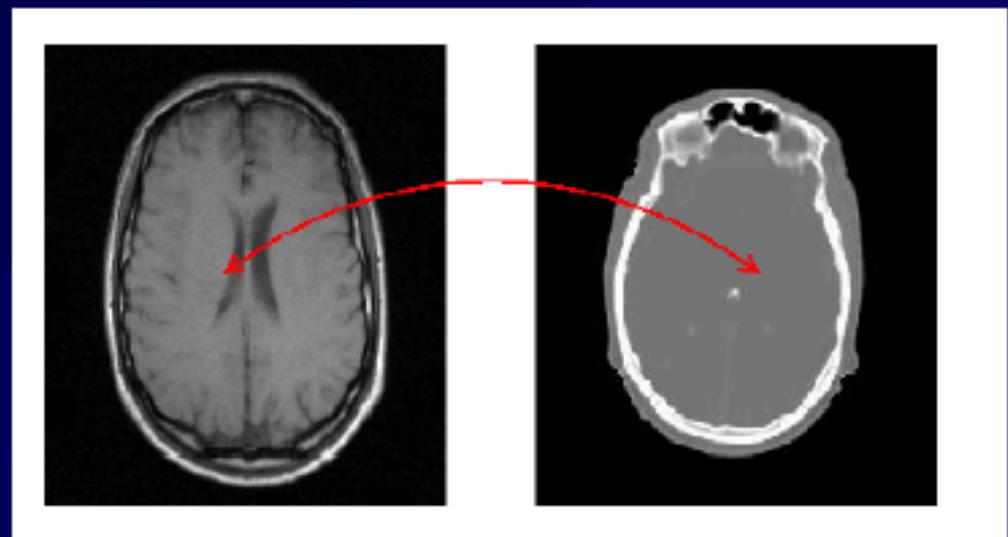
Chapitre 6 : Notions de recalage d'image

Définition

- Consiste à trouver une transformation spatiale permettant d'aligner une image (source ou **flottante**) sur une autre (cible ou **référence**).



Image registration
Image matching

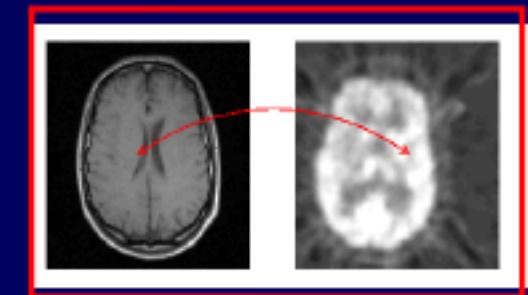
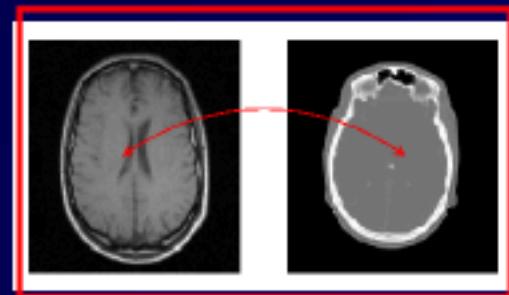
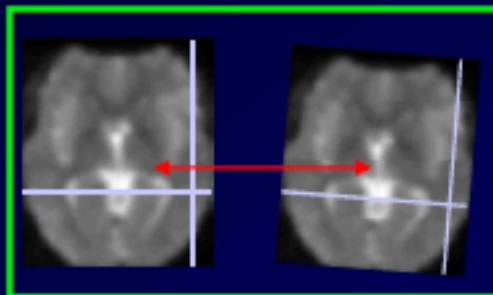


Recalage

Définition

- Consiste à trouver une transformation spatiale permettant d'aligner une image (source ou **flottante**) sur une autre (cible ou **référence**).

Recalage monomodal ou multimodal

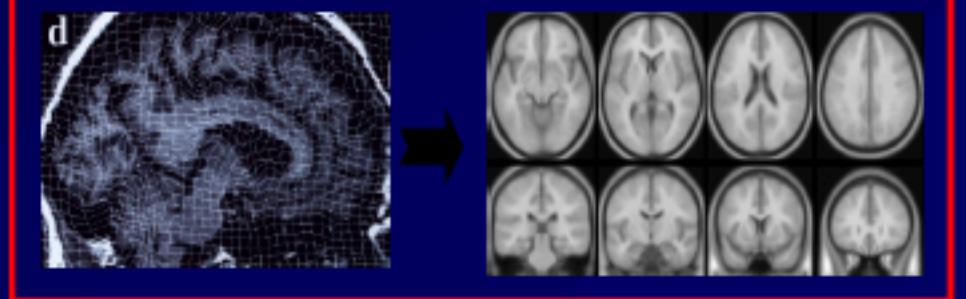
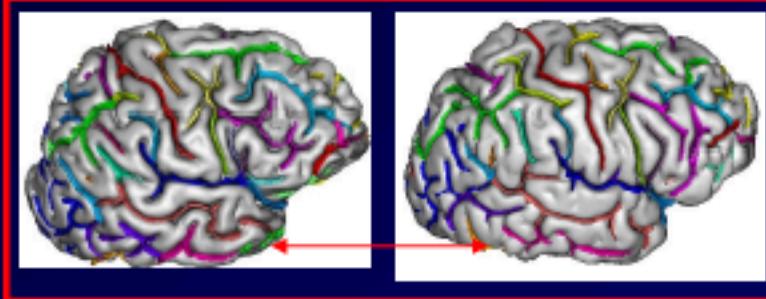
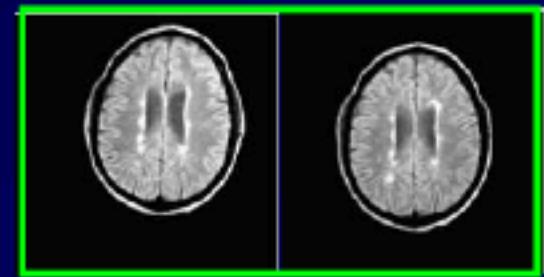


Recalage

Définition

- Consiste à trouver une transformation spatiale permettant d'aligner une image (source ou **flottante**) sur une autre (cible ou **référence**).

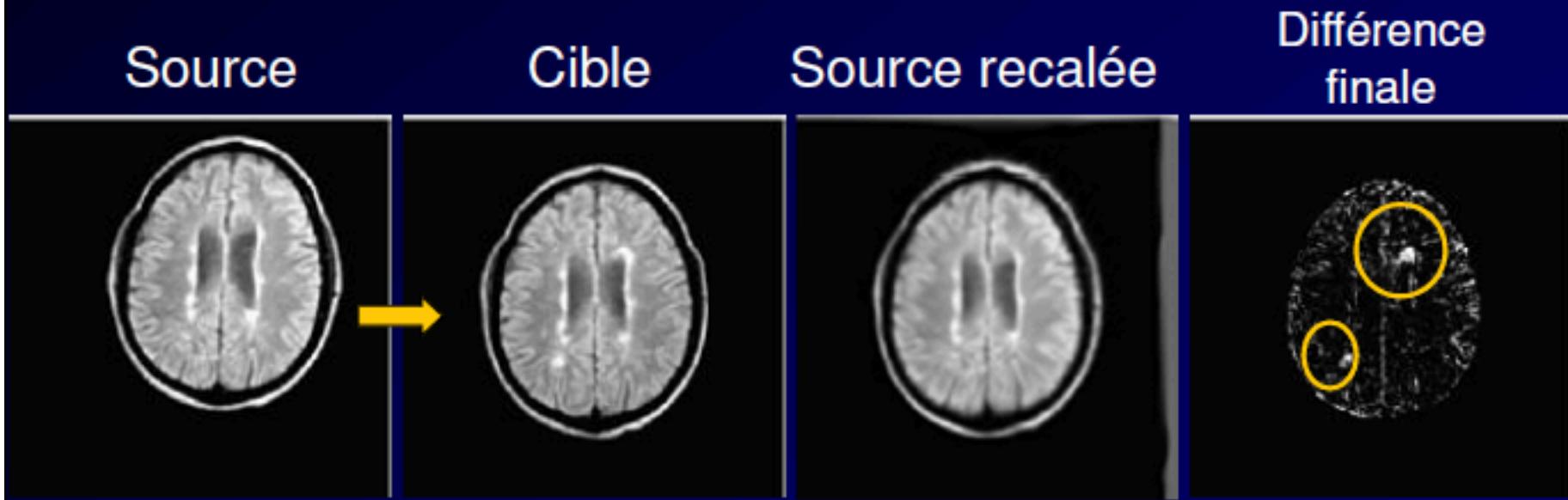
Recalage intra- ou inter-sujets



Recalage : exemples

1) Intra-patient, mono-modalité

- Exemple : évolution de lésions (images IRM d'un patient atteint de SEP à quelques mois d'intervalle)

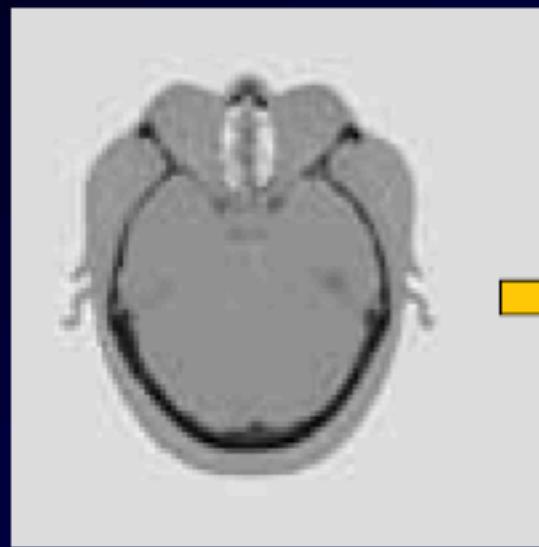


Source : Université Louis Pasteur, Strasbourg

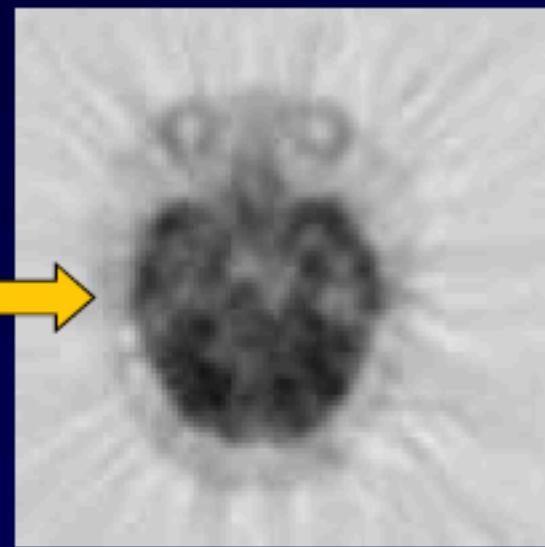
Recalage : exemples

2) Intra-patient, multi-modalité

- Exemple : fusion d'informations provenant de 2 modalités différentes



Source (scanner)



Cible (TEP)



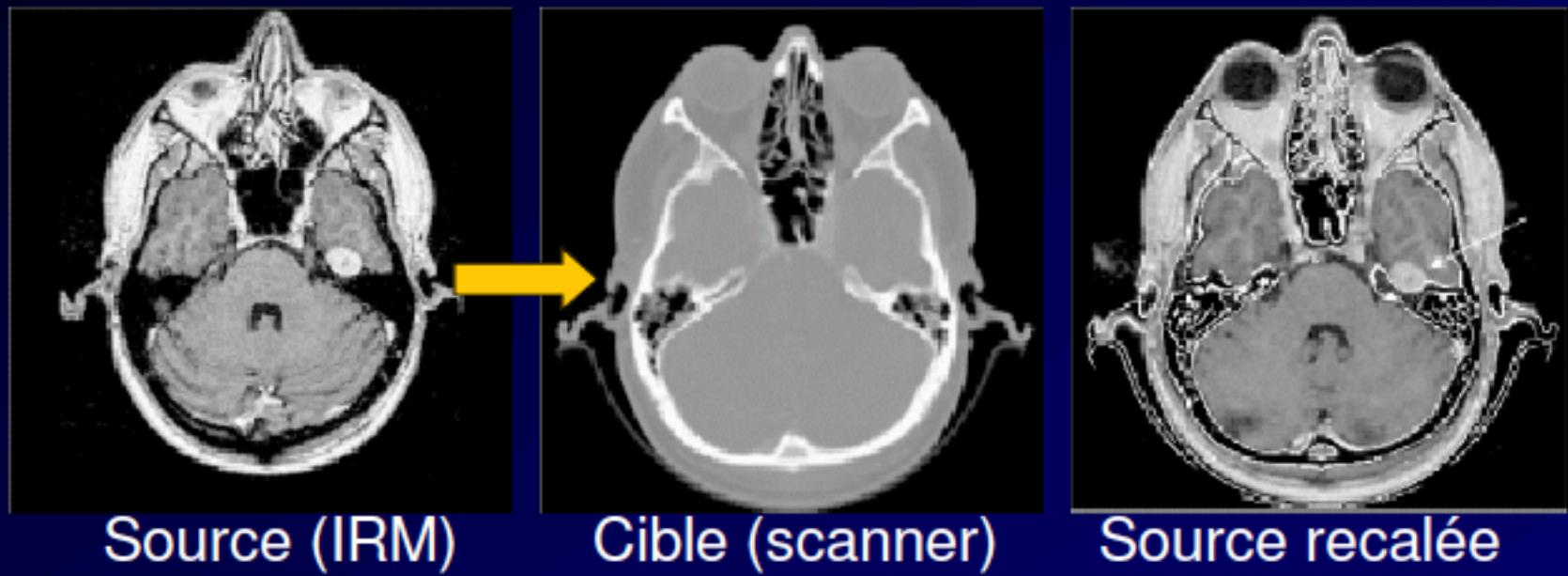
Source recalée

Source : EPFL

Recalage : exemples

2) Intra-patient, multi-modalité

- Exemple : fusion d'informations provenant de 2 modalités différentes



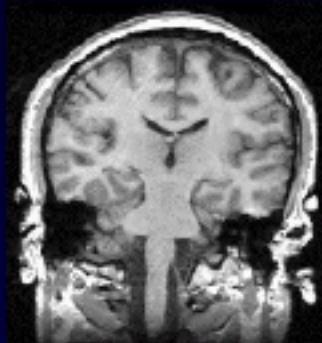
Source : Université de Hambourg

Recalage : exemples

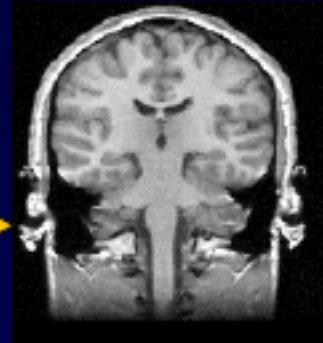
3) Inter-patient, intra-modalité

- Exemple : Segmentation à partir d'un atlas anatomique

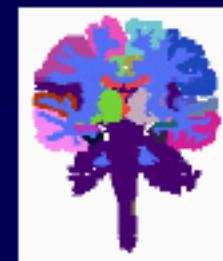
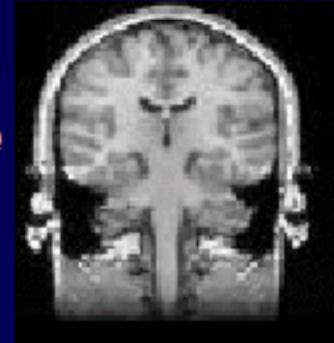
Source
(individu)



Cible
(atlas)



Source
recalée



Atlas segmenté
et étiqueté

Source : INRIA

Pourquoi faire du recalage?

- Le recalage vise à compenser les variations de mesure
 - variations géométriques (position du patient) et d'intensité
- Multiples sources d'information (Scanner, IRM, TEMP)
 - Difficulté d'interprétation des images
- **Aide au diagnostic**
- **Planification et suivi des traitements**

IMAGE DE REFERENCE

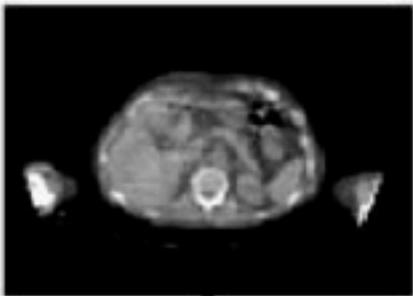
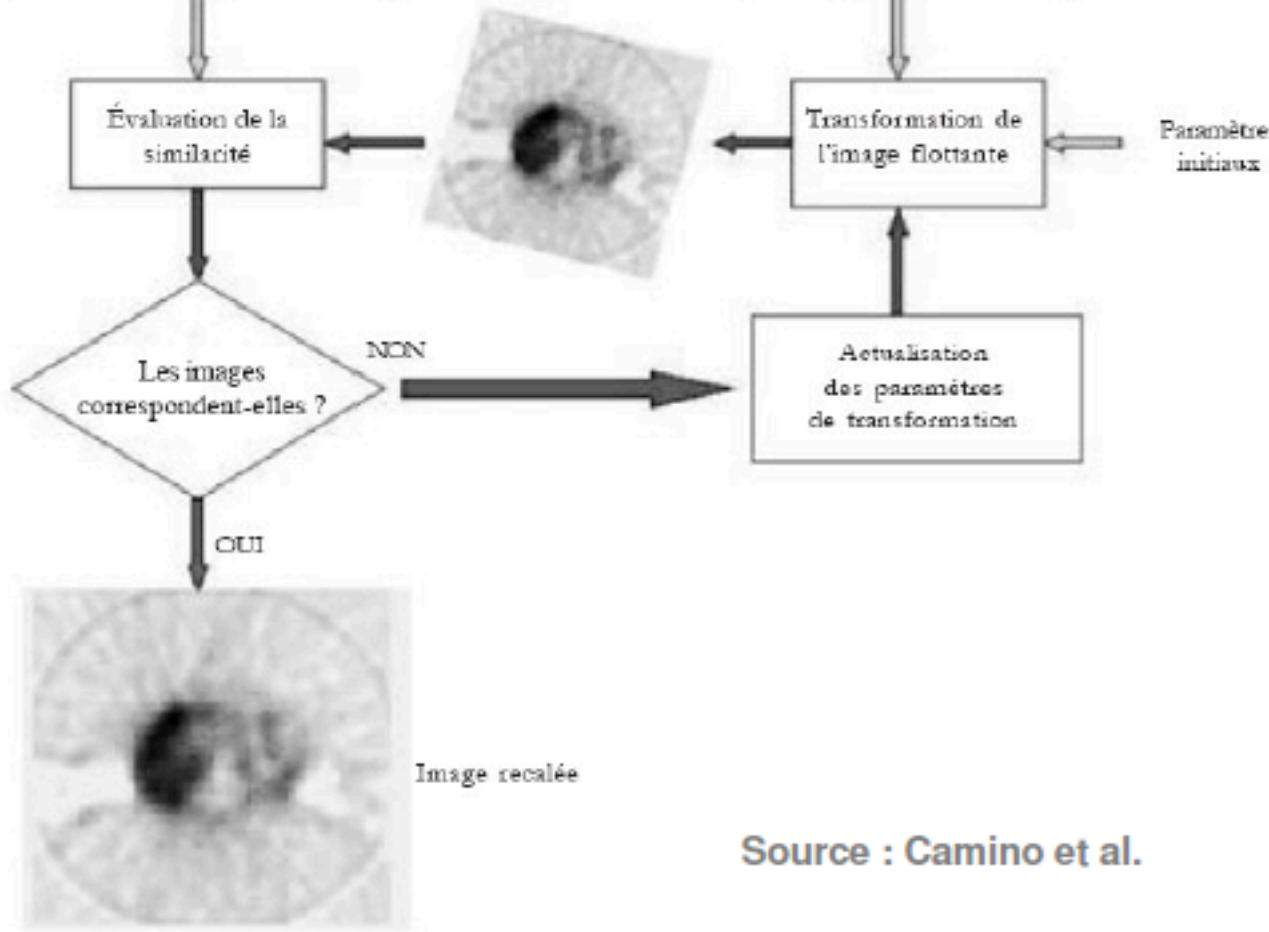
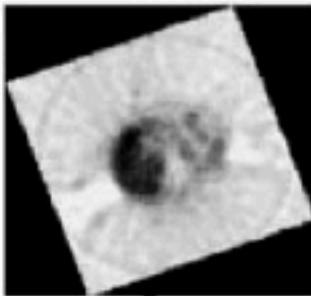


IMAGE FLOTTANTE



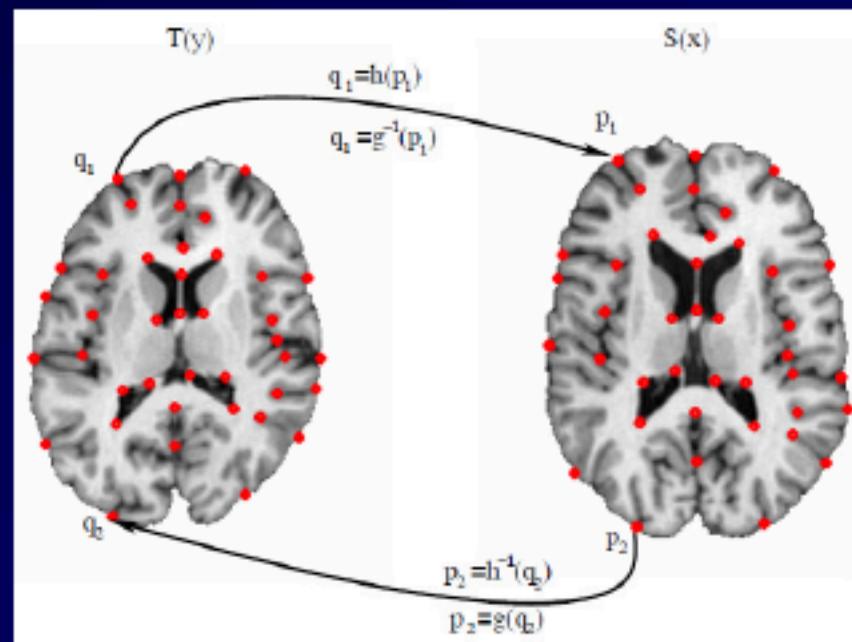
Méthode de recalage

- Structures (primitives) à mettre en correspondance
- Critère de similarité
- Transformation

Source : Camino et al.

Primitives géométriques

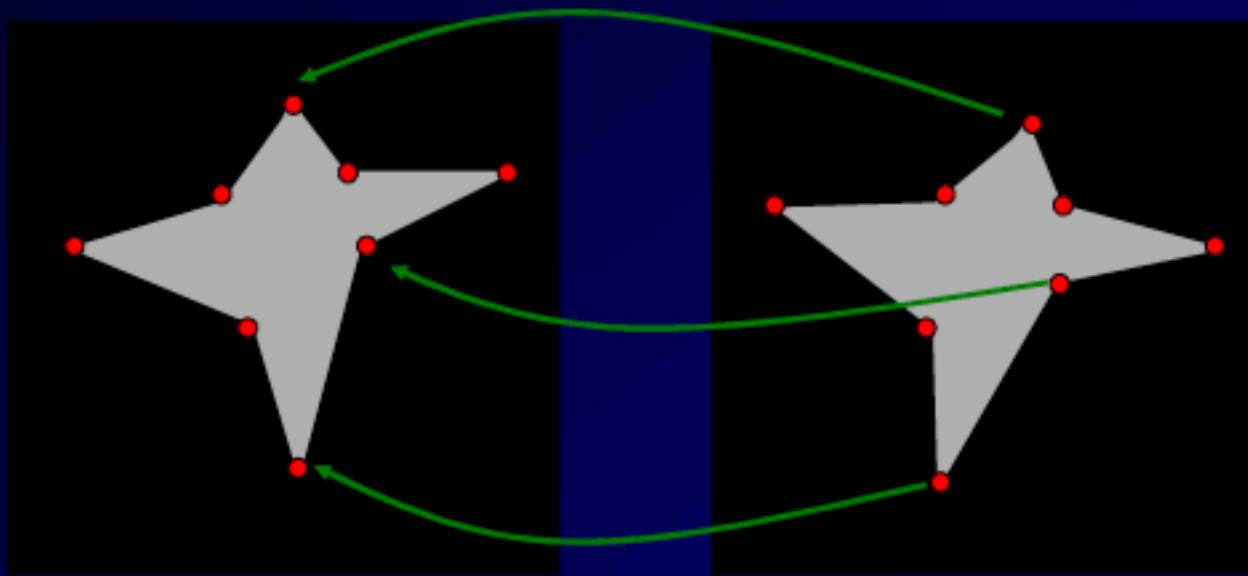
- Structures particulières dans l'image
 - Points, courbes, surfaces
 - Extraits automatiquement ou manuellement



Source : Johnson & Christensen, IPMI, 2001

Primitives géométriques

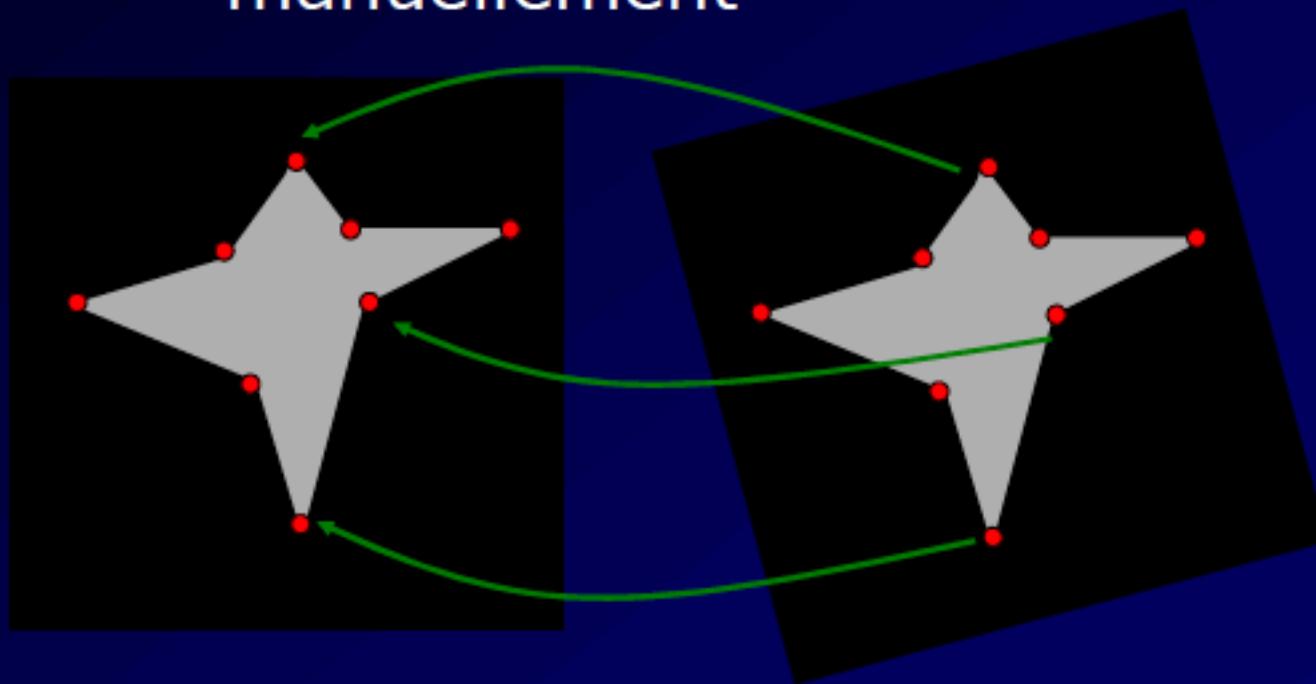
- Structures particulières dans l'image
 - Points, courbes, surfaces
 - Extraits automatiquement ou manuellement



Détection des primitives : ici points de forte courbure

Primitives géométriques

- Structures particulières dans l'image
 - Points, courbes, surfaces
 - Extraits automatiquement ou manuellement



Primitives intrinsèques

- Information pertinente présente dans les 2 jeux de données
 - Points
 - Courbes (contours)
 - Surfaces segmentées
 - Volumes
- Points anatomiques
 - identifiés manuellement par l'opérateur
 - isolés automatiquement

Primitives extrinsèques

- Repères externes, visibles dans les 2 modalités
 - fixées au patient ou à la table d'examen
 - Invasifs
 - Cadre stéréotaxique
 - Vis dans la boîte crânienne
 - Non invasifs
 - Cadre non vissé
 - Moule
 - Repères collés à la peau
- Avantages
 - Permet de recalier des données très différentes
- Inconvénients
 - Les marqueurs doivent être positionnés avant l'acquisition

IMAGE DE REFERENCE

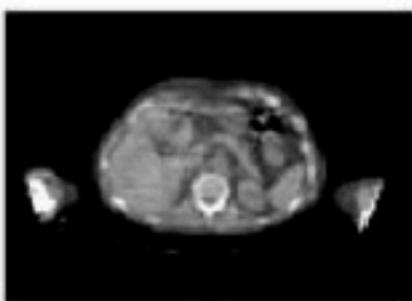
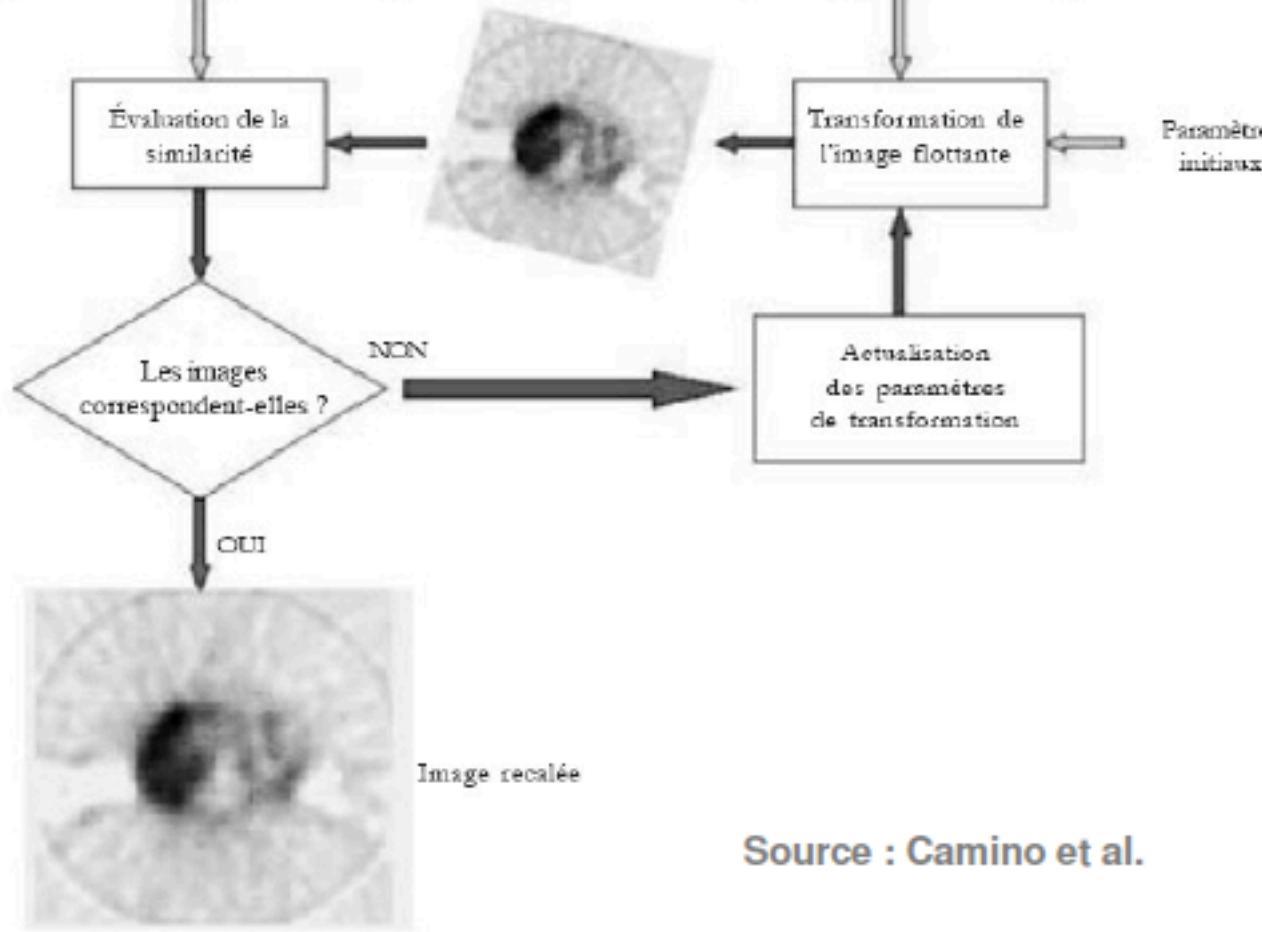
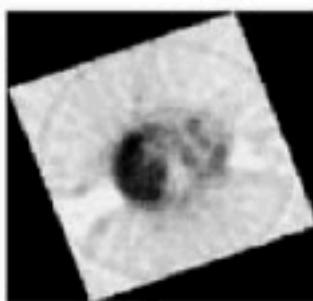


IMAGE FLOTTANTE



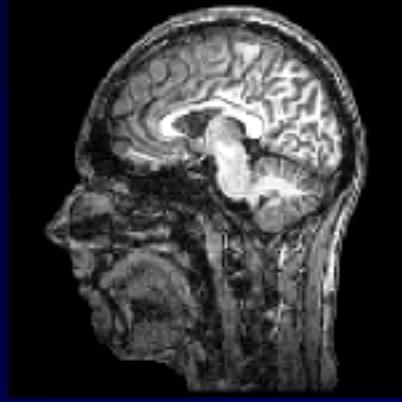
Méthode de recalage

- Structures (primitives) à mettre en correspondance
- Critère de similarité
- Transformation

Source : Camino et al.

Critère de similarité

Comment dire que 2 images sont similaires ?



Cas « facile »



Cas difficile !

De nombreux critères existent

- Minimiser le critère des moindres carrés
(SSD = sum of squared differences)

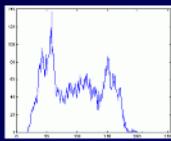
Monomodal

$$SSD(IoT, J) = \sum_{pixels\ x} (IoT(x) - J(x))^2$$

- Critères basés sur la théorie de l'information

- Aucune hypothèse n'est faite sur le rapport entre les intensités des images recalées

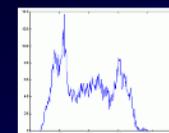
- Entropie d'une image $H(I) = - \sum_i p_i \log(p_i)$



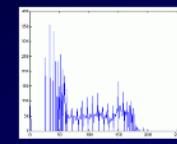
Entropie H=2

Multimodal

- Entropie d'une image $H(I) = - \sum_i p_i \log(p_i)$



Entropie H=2

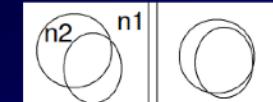


Entropie H=1.7

L'entropie est une mesure de l'incertitude

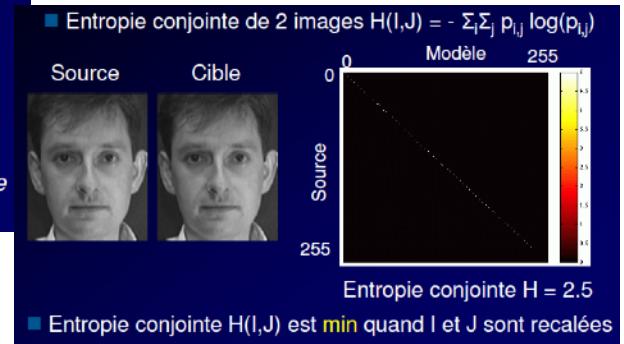
- Pour chaque NdG n présent dans A:
 - on note les coord des pixels de valeur n
 - On va chercher dans B les pixels de mêmes coordonnées
 - On calculer la moy, l'écart-type sur ces pixels

avant recalage



Multimodal

après recalage



■ Information mutuelle :

Multimodal

- $IM(I,J) = H(I) + H(J) - H(I,J)$
- On cherche à la maximiser

IMAGE DE REFERENCE

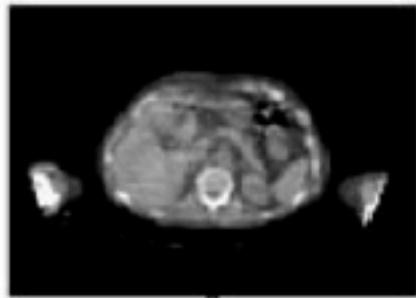
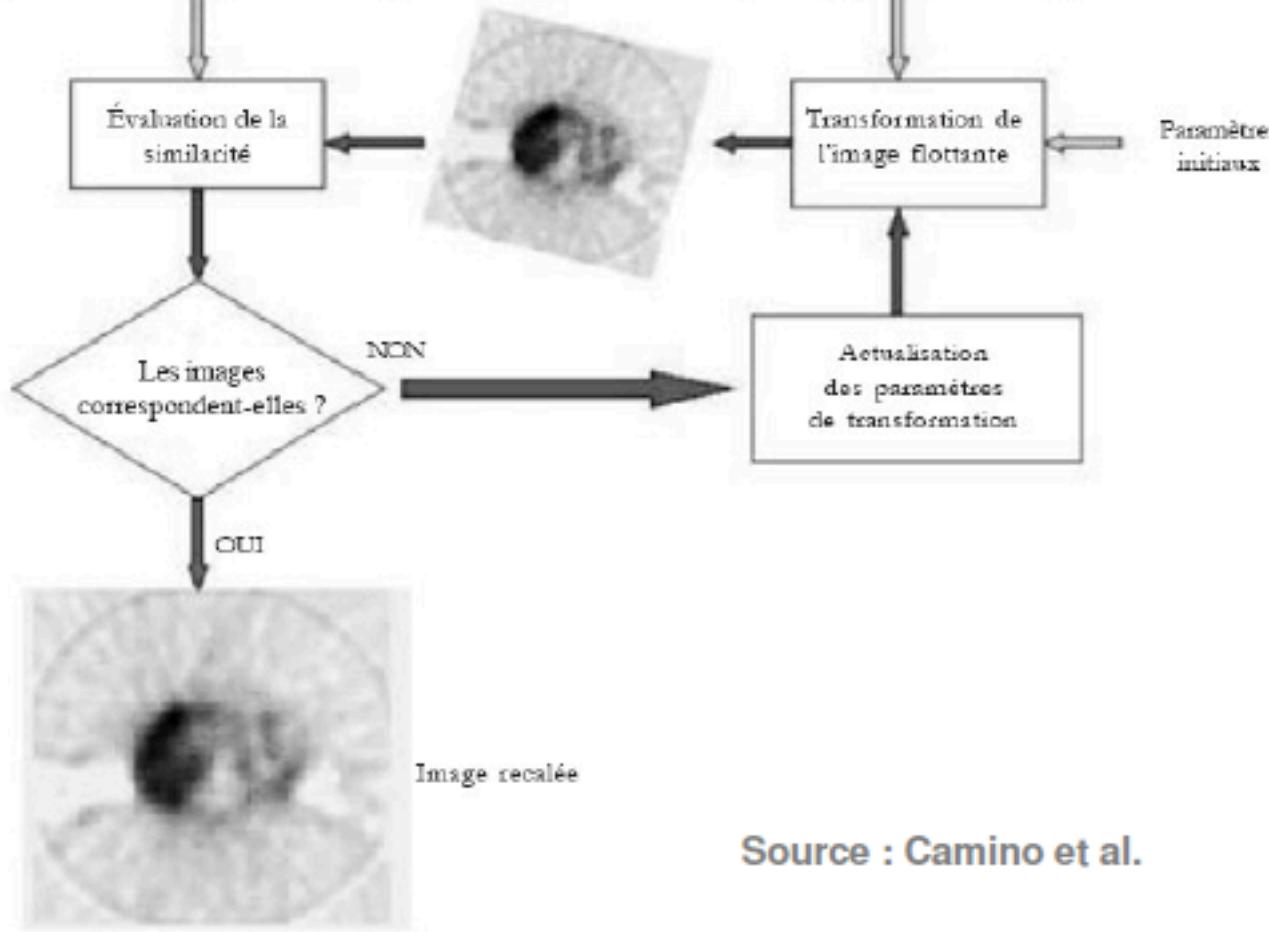
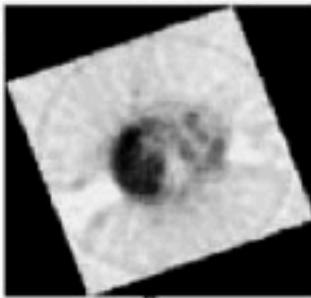


IMAGE FLOTTANTE



Méthode de recalage

- Structures (primitives) à mettre en correspondance
- Critère de similarité
- Transformation

Source : Camino et al.

Différents types de transformation

- Domaine de la transformation utilisée
- Globale
 - S'applique de manière identique à toute l'image
- Locale
 - Des sous-sections de l'image ont leurs propres transformations

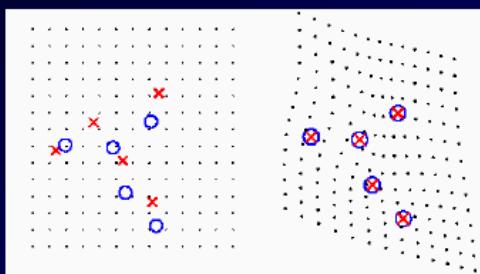
- Rigide
 - Translation – rotation
 - Conservation des angles et des distances
 - + pour recalage intra-patient / intra modalité



- Affine
 - Des droites parallèles restent parallèles après transformation

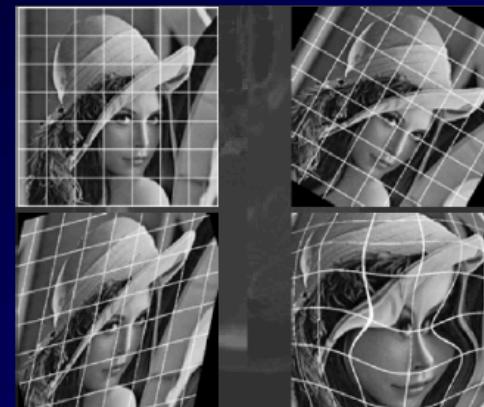


- Elastique
 - Toutes les déformations sont possibles



Différents types de transformation : résumé

- Recalage rigide ou affine
 - Erreur de positionnement
 - Pas de déformations de tissus mous
- Recalage non-rigide ou élastique
 - Petites déformations locales



Source : <http://www.inf.u-szeged.hu/~ssip/2007/lectures/AttilaTanacs-Registration.pdf>

IMAGE DE REFERENCE

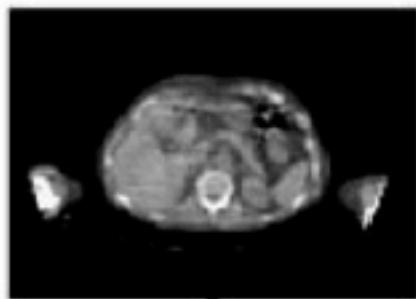
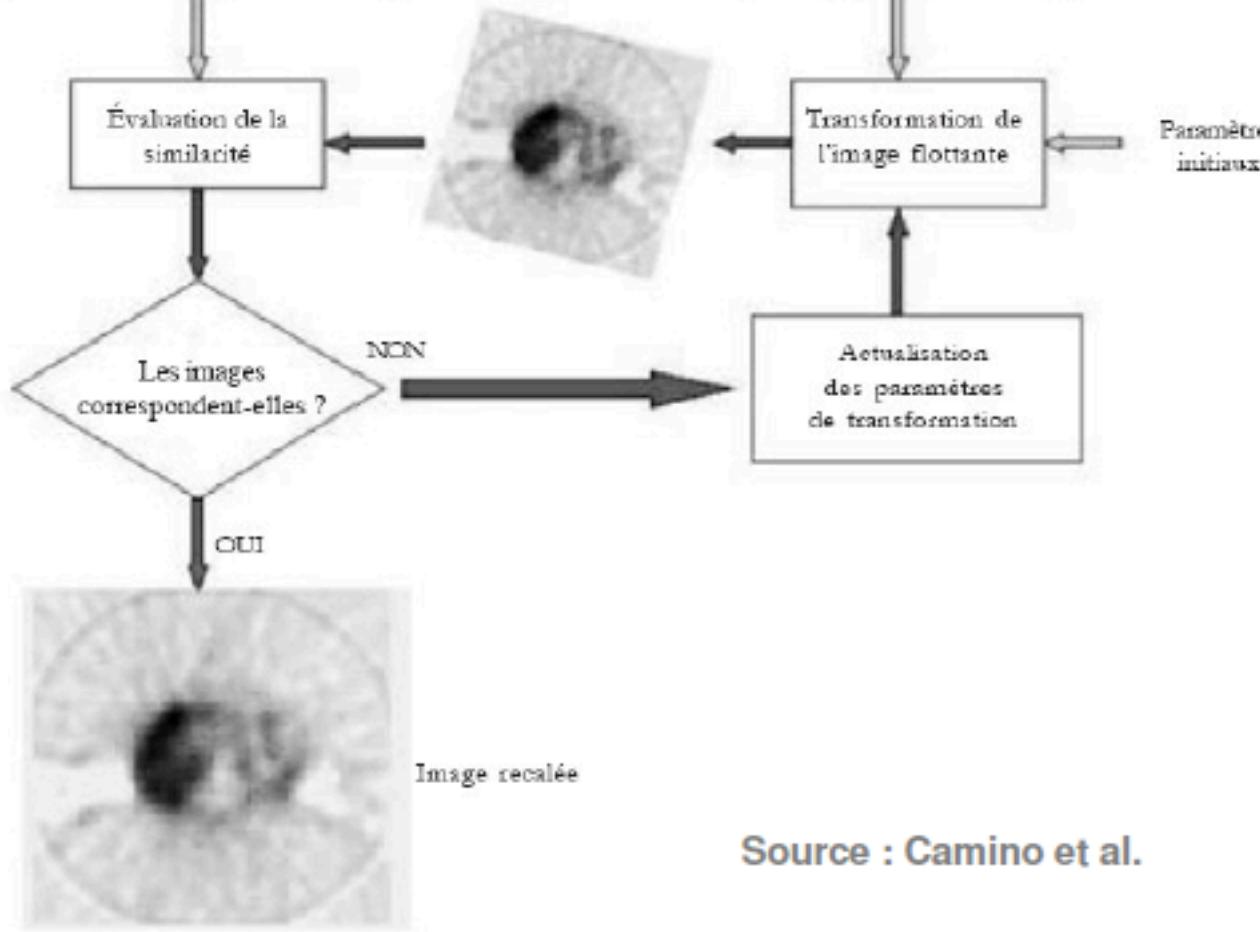
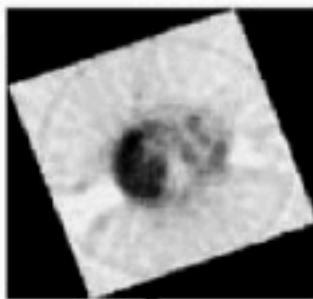


IMAGE FLOTTANTE



Méthode de recalage

- Structures (primitives) à mettre en correspondance
- Critère de similarité
- Transformation

Source : Camino et al.

- Mise en correspondance de points, courbes, etc positionnés manuellement ou automatiquement
- Très souvent recalage rigide

Recalage par chanfrein

- Recalage rigide nécessitant les contours initiaux



Image de référence

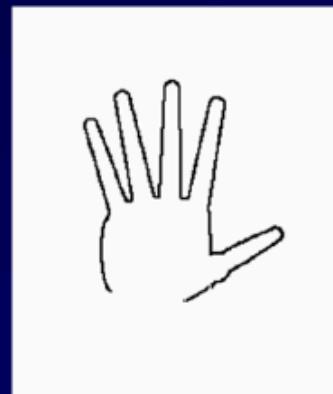


Image flottante

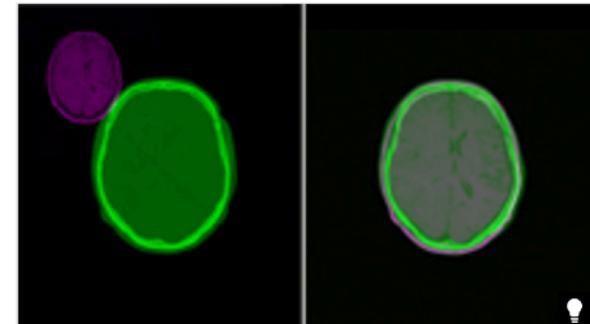
Que pouvons nous faire avec nos outils type Matlab, Octave ou IJ ?

Par ex. Matlab :

- imtransform
- cpselect
- cp2tform
- ...

Recalage d'images basé sur l'intensité

Le recalage d'images basé sur l'intensité aligne des pixels spécifiques dans des images en utilisant l'intensité relative. Cette technique de recalage est souvent bien adaptée à l'imagerie médicale lorsqu'il est nécessaire d'automatiser de grandes collections d'images.



Recaler des images médicales 3D multimodales
(Example)

Recalage d'images par points de contrôle

Le recalage d'images par points de contrôle nécessite la sélection manuelle des points de contrôle en deux images dans le but de les aligner. Cette méthode de recalage est la mieux adaptée pour les images qui ont des caractéristiques distinctes telles que les images satellites.



Recalage d'une photo aérienne sur une orthophoto
(Example)

Transformations géométriques

Les transformations géométriques sont utiles pour les tâches telles que la rotation d'images, la réduction de la résolution, la correction des distorsions géométriques et le recalage d'images. Image Processing Toolbox prend en charge des opérations simples comme le redimensionnement, la rotation et le rognage, ainsi que des transformations géométriques 2D plus complexes telles que les transformations affines et projectives.

La boîte à outils fournit un ensemble de fonction complet et flexible pour la création et l'application des transformations géométriques personnalisées et des méthodes d'interpolation pour les tableaux multidimensionnels.

