

Image Numérique

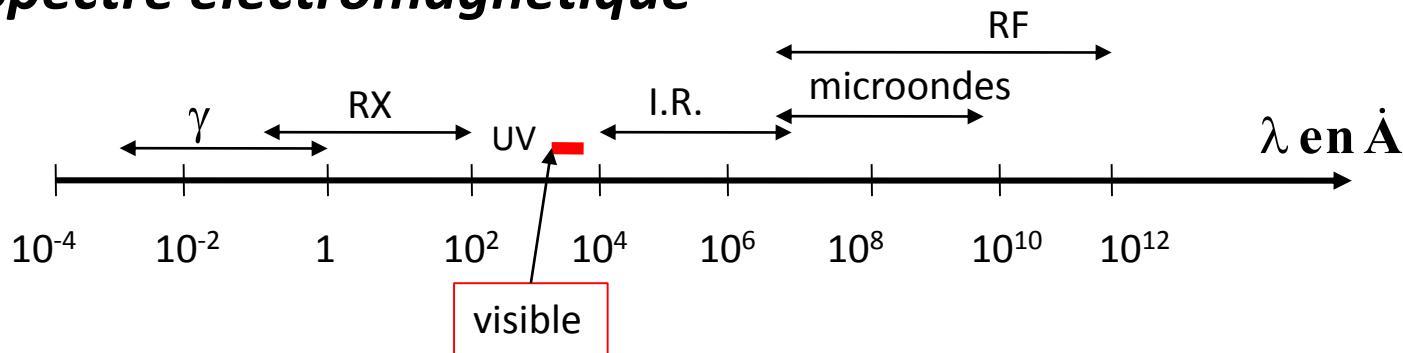
Définitions et outils



Nature des images

Une image se caractérise d'une part par la longueur d'onde du signal physique reçu et par son intensité enregistrée par le capteur. Le capteur est généralement de dimension 2, le résultat est donc un signal 2D analogique ou numérique.

Rappel Spectre électromagnétique



La plupart des images se font dans la partie visible du spectre.

Cependant il existe d'autres capteurs qui utilisent les autres domaines.

- Exemples : - Scanner (RX sous plusieurs angles + reconstruction)
- IRM (image à résonnance magnétique)
- Ultra-son (échographie, contrôle non-destructif)
- Sons Basse fréquence en image de prospection pour la surface de la terre (prospection pétrolière)
-



Signaux 2D - Acquisition

• Signaux continus multidimensionnels

Représentation unifiée : soit $f(\vec{q})$ le signal multidimensionnel de dimension D.

$\vec{q} = [q_1 q_2 \dots q_n]^T$ Vecteur unitaire de dimension D

Exemples : $f(t)$: grandeur en fonction du temps

$f(x)$: _____ d'une position

$f(x,y)$: _____ de 2 positions -> image $f(x,y) = \text{luminance}$

$f(x,y,t)$: _____ + temps -> video

etc.... (d'autres existent par exemple avec la longueur d 'onde)

Le signal f représente généralement une quantité physique (ex :luminance), on peut lui attribuer certaines propriétés permettant un traitement plus aisé...

Continuité : les signaux réels n 'ont généralement pas de discontinuité => différentiables

Domaine fini : la nature physique du signal et du capteur assure que le signal est limité sur un certain domaine.

Energie finie : le signal correspond à l 'amplitude ou l 'énergie d 'un phénomène physique. L 'énergie de tout système physique étant limitée, l 'intégration est possible et :

$$\int_{-\infty}^{\infty} |f(\vec{q})|^2 d^D q < \infty \Rightarrow \text{la TF existe et donc on peut utiliser tous ces outils}$$



- Signaux discrets multidimensionnels

Matrice régulière 2D

Les images numériques sont constituées par des tableaux 2D de points à valeurs discrètes. Un point sur une grille 2D est appelé pixel (ou pel : picture element), il se caractérise par ses coordonnées sur la grille 2D et on lui affecte une grandeur, la luminance présente en ce point.

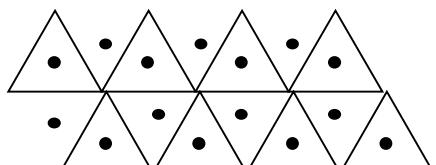
La conversion lumen->électricité->valeur numérique est assurée par le capteur.

Le capteur intègre la lumière reçue sur chaque cellule.

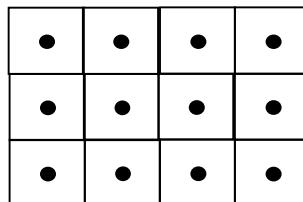
Le CAN convertit en valeur numérique.

Il existe plusieurs façons d'organisations des cellules dans les capteurs ,dont la principale :

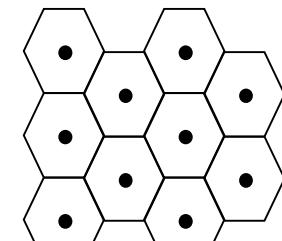
I 'espace 2D est partitionné en cellules équiréparties, pour des raisons de symétrie, on ne considère que les polygones réguliers, on obtient alors :



a) triangle



b) carré

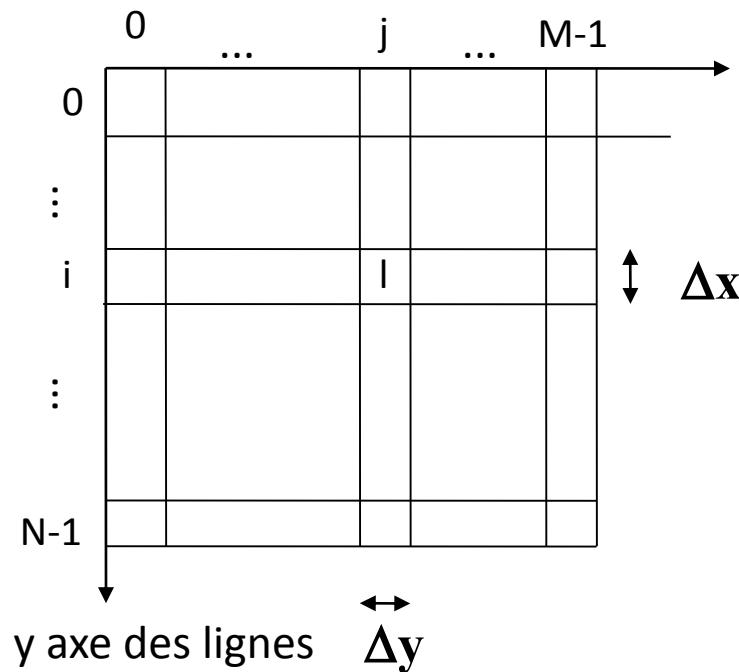


c) hexagone



Image numérique : 1 ou 3 tableaux 2D

Si les matrices ne sont pas forcément orthogonales avant la discréétisation, il est clair qu'ensuite il est beaucoup plus simple d'utiliser une représentation orthogonale, une image numérique est donc très souvent représentée de la manière suivante :



x axe des colonnes

Dimension en x : N
y : M

Une image est une application I :

$$I : [0, N-1] \times [0, M-1] \rightarrow [0, L]^k$$

$$I(i, j) = (i\Delta x, j\Delta y)^T = l \text{ =luminance du pixel de coordonnées (i,j)}$$

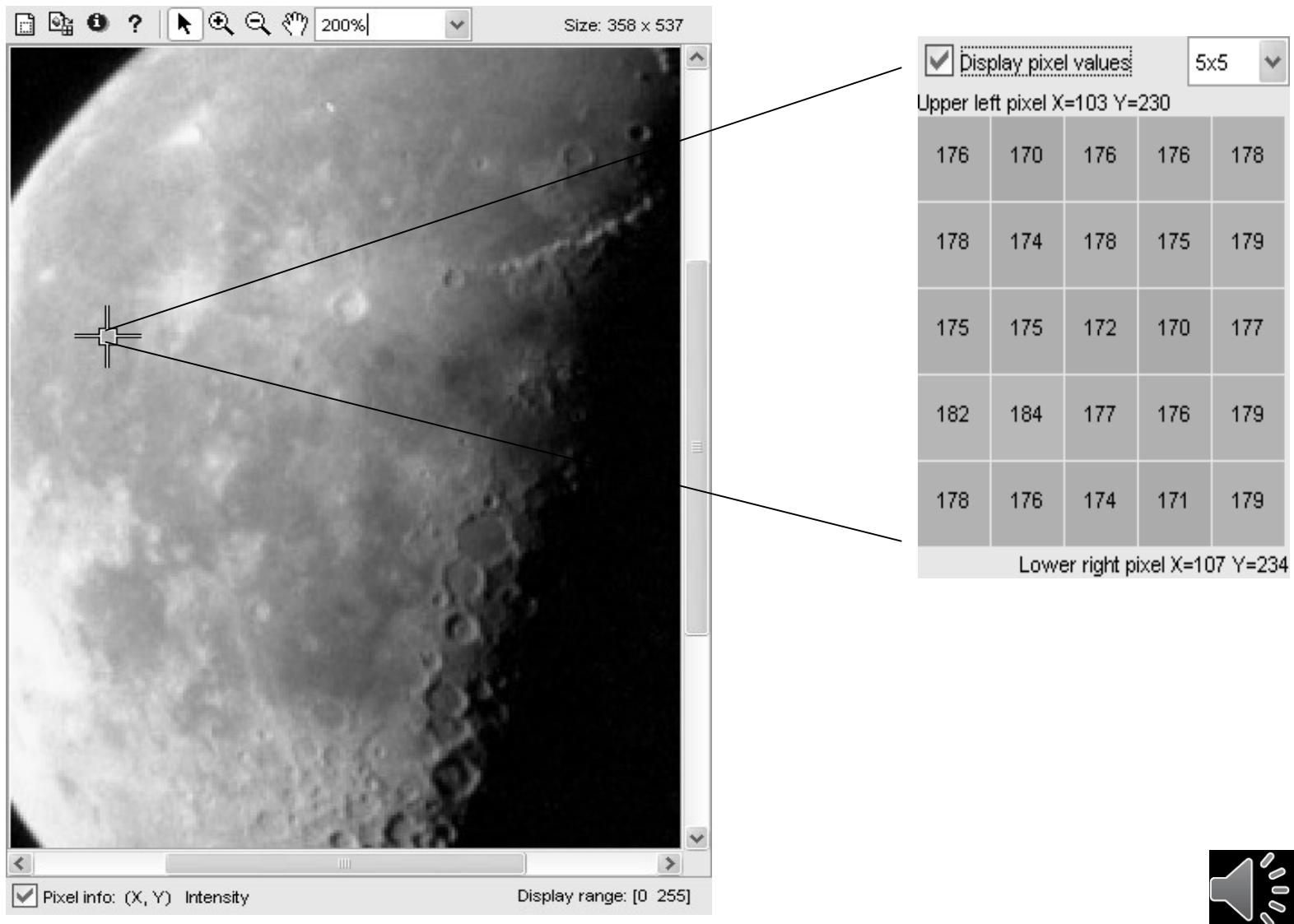


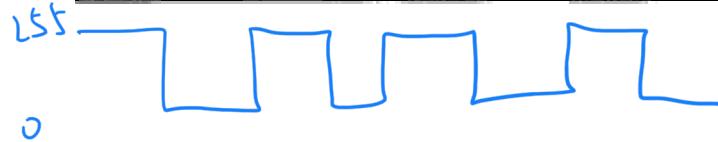
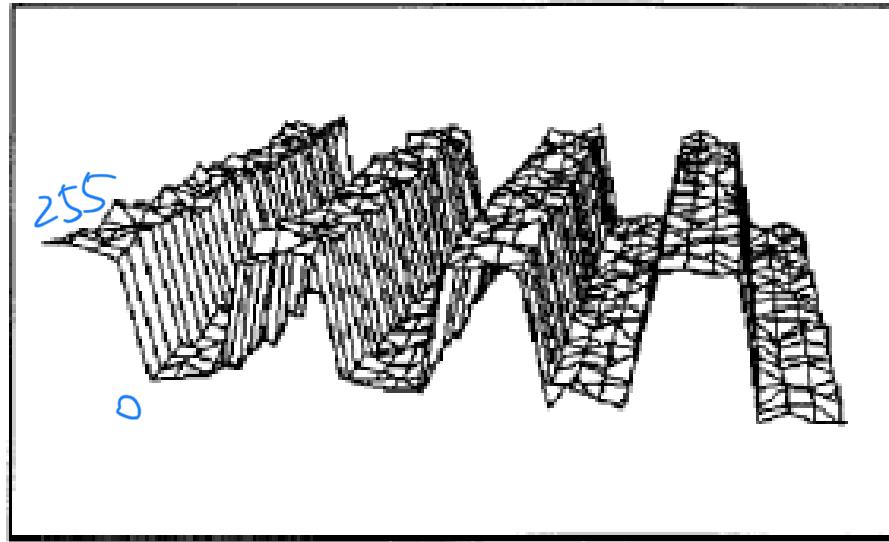
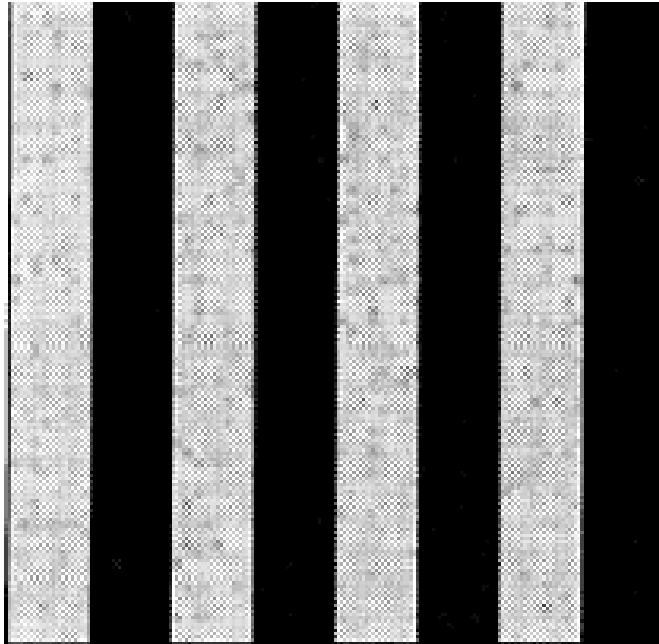
Les différents types d'images numériques

- Le nombre de bits affectés à la luminance l permet de définir les différents types d'images :
 - Images Noir et Blanc (1 bit)
 - Images niveaux de gris : l'intensité du pixel est codée sur n bits (en général 8 soit 256 niveaux de gris)
 - 0 ← noir
 - 255 ← blanc
 - Images couleurs : 2 représentations sont possibles
 - Codées sur 3 plans (RVB ou Ycb,Cr ou autre espace de couleur)
 - Indexées : table de couleur (Colormap)+un plan qui contient l'indice dans la table de couleurs (tableau 3*n)



Image niveau de gris

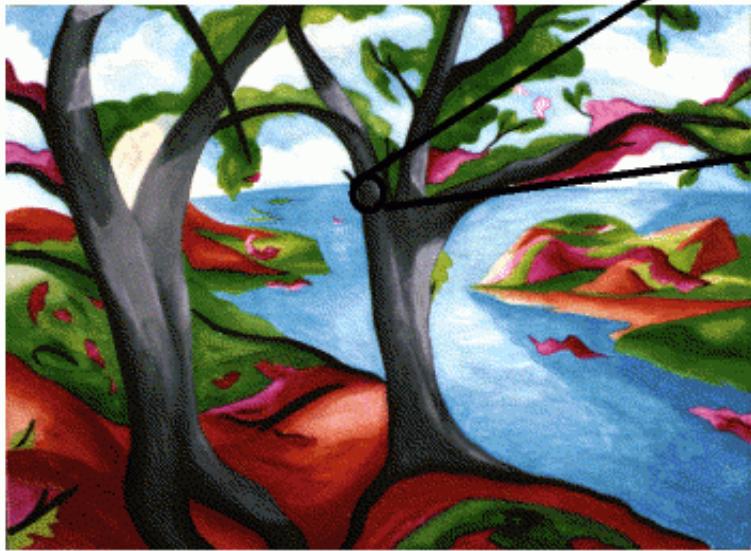




Représentation image et représentation 3d pour illustrer la valeur de l'intensité



Images couleurs indexées



12	21	40
14	17	21
5	8	15
1	15	18

0	0	0
0.0627	0.0627	0.0314
0.2902	0.0314	0
0	0	1.0000
0.2902	0.0627	0.0627
0.3882	0.0314	0.0941
0.4510	0.0627	0
0.2588	0.1608	0.0627
	:	

Colormap

Une ligne définit une couleur, le nombre de lignes définit le nombre de couleurs utilisées

Colormap : Un tableau de 3 composantes c1(pourcentage de rouge), c2 (vert), c3(bleu)



Image couleur 3 plans

3 matrices 2D,
exemple : R, V et
B



R(244,205)=245
V(244,205)=186
B(244,205)= 0



Relations entre signaux continus 2D et signaux discrets 2D

- Les outils : la Transformée de Fourier 2D, quelques rappels

Cas continu-continu : soit $f(x,y)$ une fonction continue 2D à variables réelles x et y , et $F(u,v)$ sa transformée de fourier.

$$\text{TF}[f(x,y)] = F(u,v) = \iint f(x,y) e^{-2i\pi(ux+vy)} dx dy$$

$$\text{TF}^{-1}[F(u,v)] = f(x,y) = \iint F(u,v) e^{+2i\pi(ux+vy)} du dv$$

u et v représentent les fréquences spatiales dans les deux directions.

Cas discret-continu : f et ses variables sont discrètes, sa transformée de Fourier peut être continue.

$$\text{TF}[f(x_i, y_j)] = F(u, v) = \sum_{x_i=-\infty}^{\infty} \sum_{y_j=-\infty}^{\infty} f(x_i, y_j) e^{-2i\pi(ux_i + vy_j)}$$

Cas discret-discret : f est discrète, sa transformée de Fourier aussi (cas réel, implémentation)

$$\text{TF}[f(x_i, y_j)] = F(u_k, v_l) = \frac{1}{MN} \sum_{x_i=0}^{M-1} \sum_{y_j=0}^{N-1} f(x_i, y_j) e^{-2i\pi(\frac{ux_i}{M} + \frac{vy_j}{N})}$$

où M est le nombre d'échantillons pris dans la direction x

N _____ y



$F(u,v)$ est complexe, on peut l'écrire :

$$F(u,v) = \operatorname{Re}(u,v) + i \operatorname{Im}(u,v) = |F(u,v)| e^{i\Phi(u,v)}$$

avec $|F(u,v)| = \sqrt{\operatorname{Re}^2(u,v) + \operatorname{Im}^2(u,v)}$

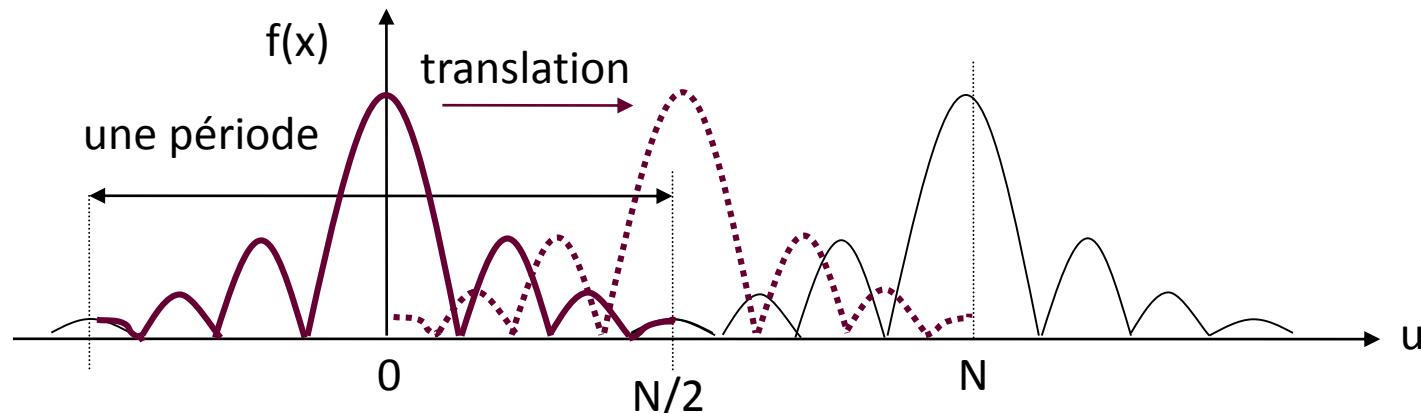
Propriétés de la TF2D

- séparabilité : on peut montrer qu'il est équivalent de faire successivement 2 TF 1D à la place d'une TF 2D
- Translation : Seule la phase est modifiée lors d'une translation de la fonction f
- Périodicité et conjugué symétrique : la TF d'un signal discret est périodisée

$$\underline{F(u,v) = F^*(-u,-v) \text{ ou } |F(u,v)| = |F(-u,-v)|}$$

- Convolution :

$$f(x,y) * g(x,y) \Leftrightarrow F(u,v) \cdot G(u,v)$$



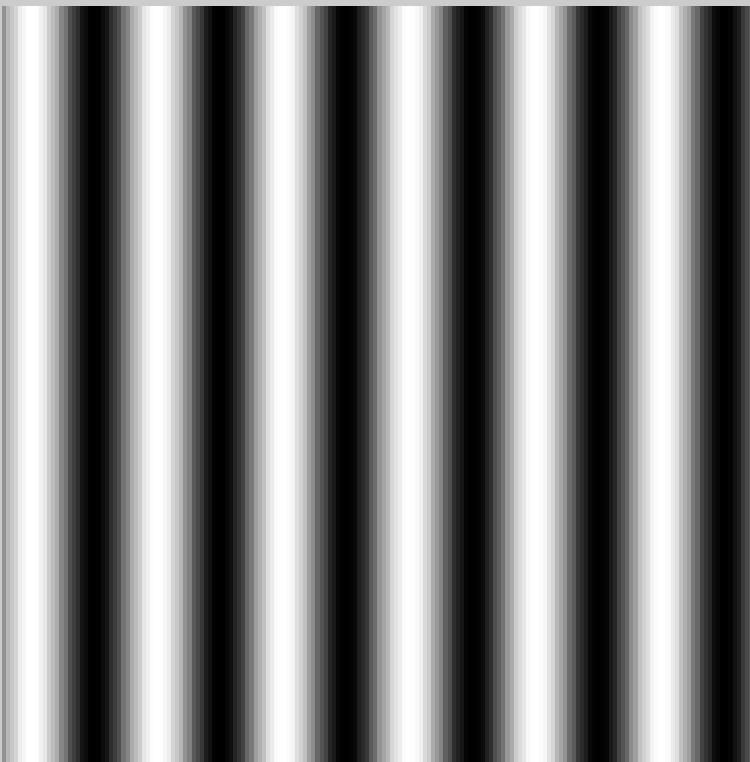
Transformée de Fourier

- Si besoin revoir les cours de l'année précédente sur la transformée de Fourier 1D et le traitement du signal numérique
- Quelques liens du web :
 - Wikipédia : [Transformée de Fourier](#), [TZ](#), [filtrage numérique](#)
 - [Rappel Traitement du signal](#)
 - [Formulaire](#)



Exemple : Transformée de fourier d'un sinus 2D

image originale



spectre

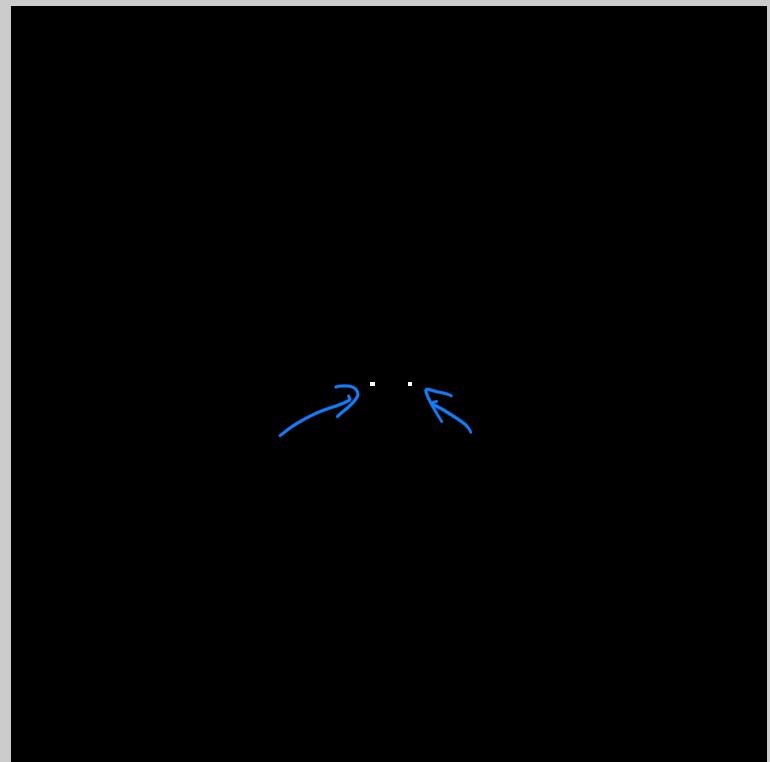
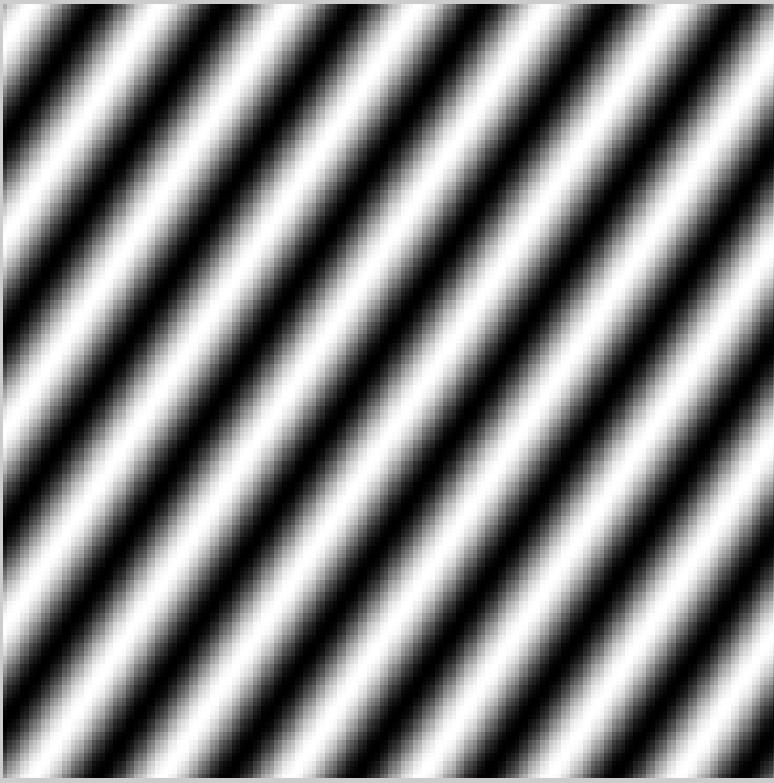
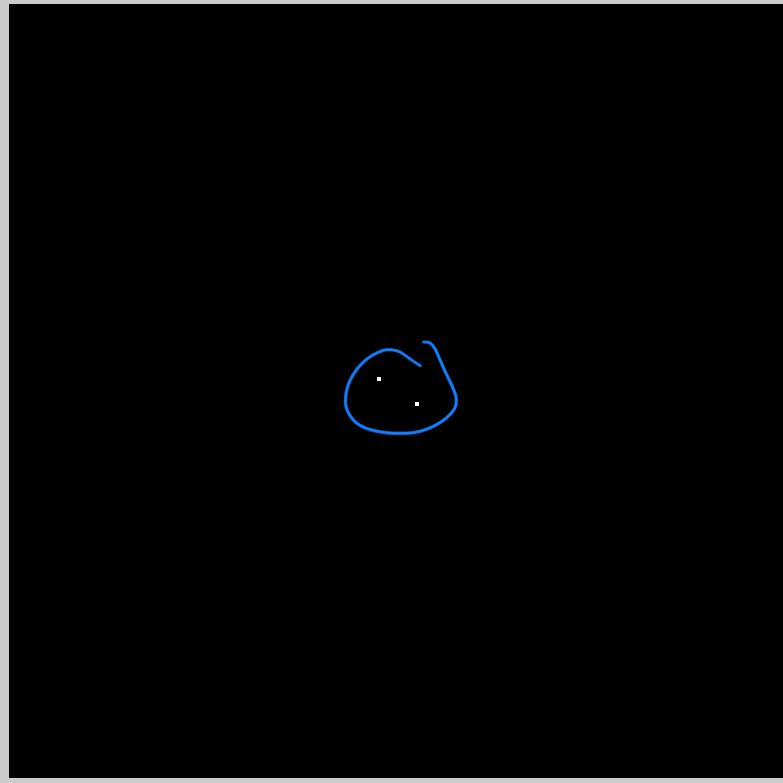


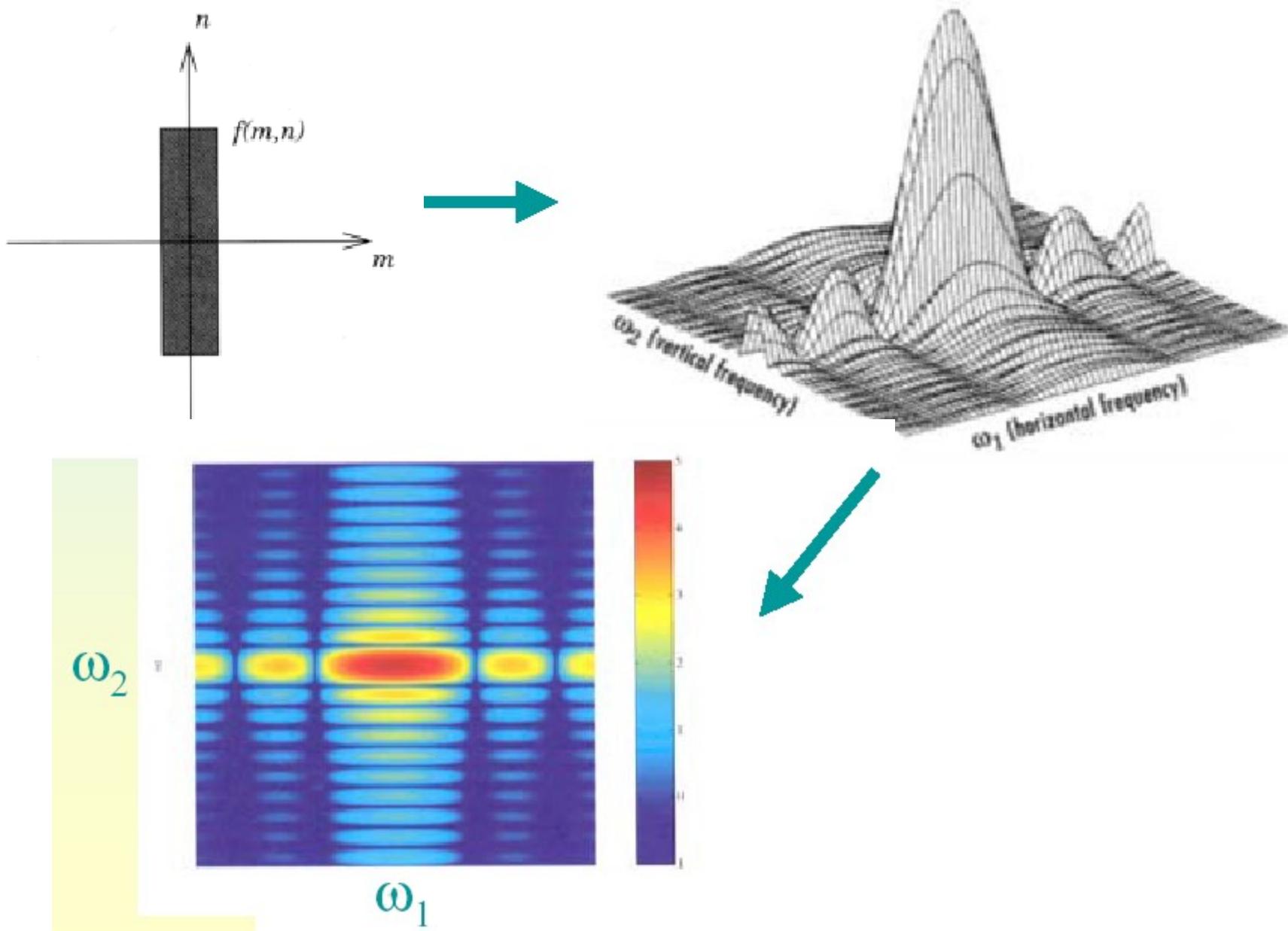
image originale



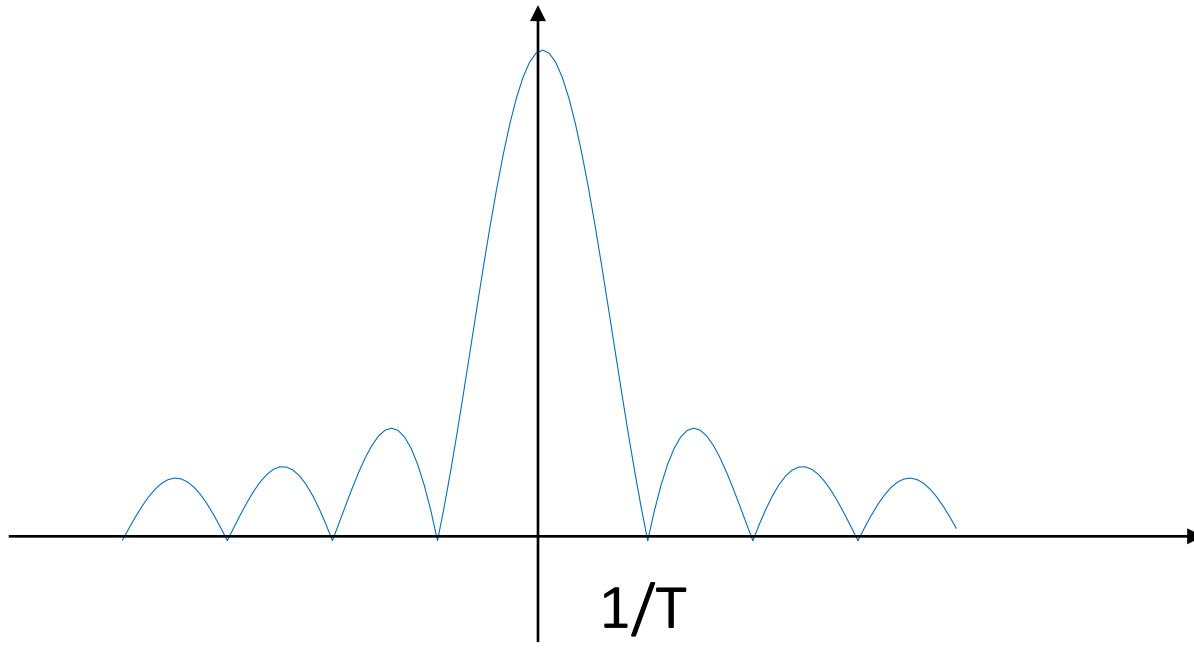
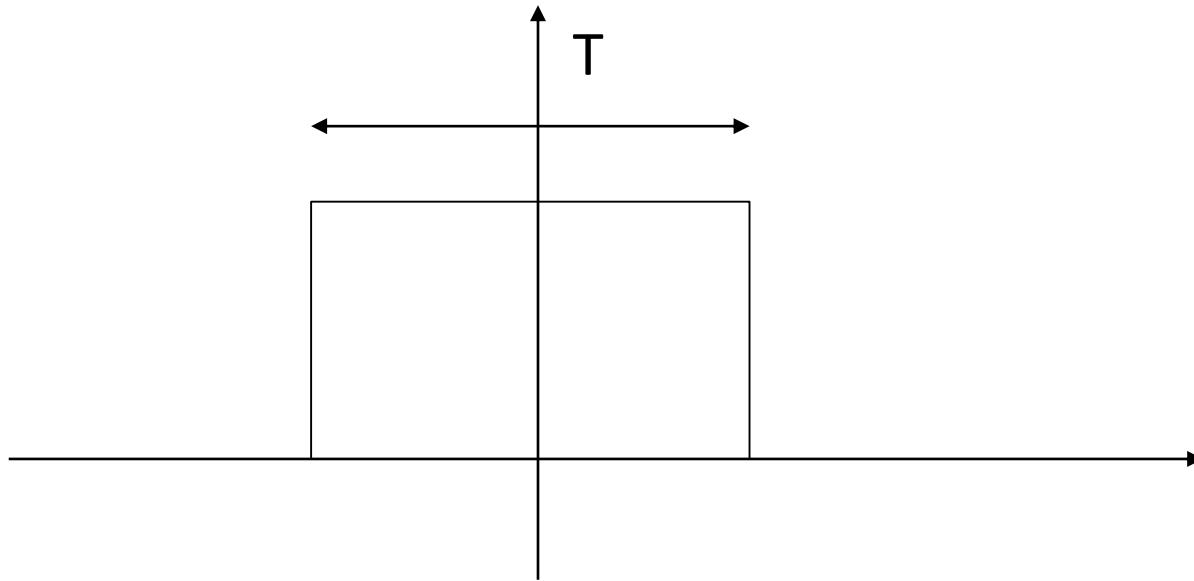
spectre



Exemple : Transformée de Fourier d'un rectangle



Rappel



Quelques exercices

$$F(u, v) = \sum_x \sum_y [f(x, y) \exp(-2\pi(iux + vy))] = \sum_x \left[\sum_y f(x, y) \exp(-2\pi(vy)) \right] \exp(-2\pi(iux)).$$
$$= \text{TF}_x [\text{TF}_y f(x, y)]$$

- Exercice 1

Montrez qu'il est équivalent de faire une transformée de Fourier 2D directe avec une transformée de Fourier 1D dans un sens suivie d'une transformée de Fourier dans l'autre sens.

- Exercices sur Matlab (voir les démos)



image originale



spectre

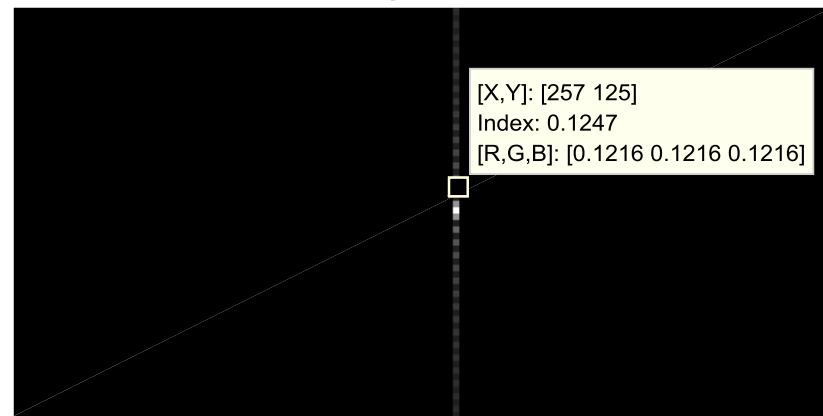
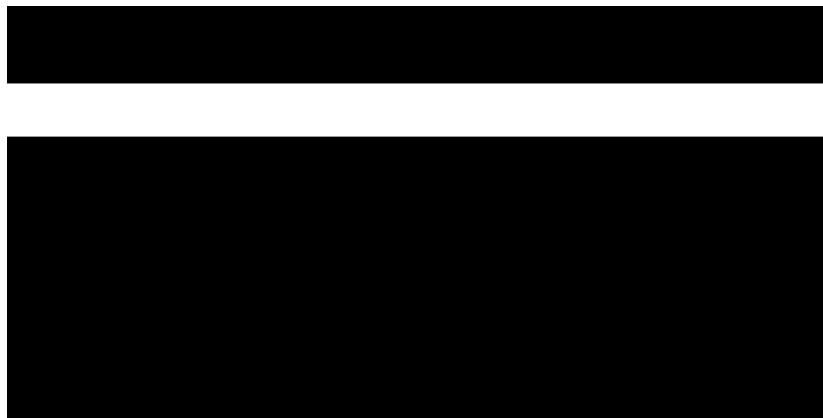
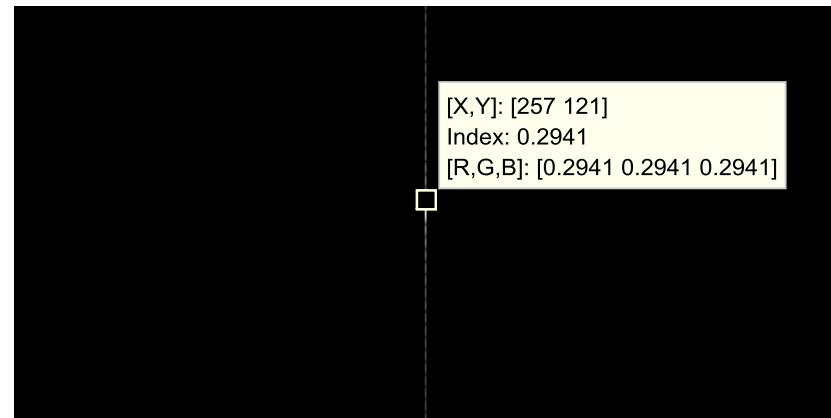


image originale

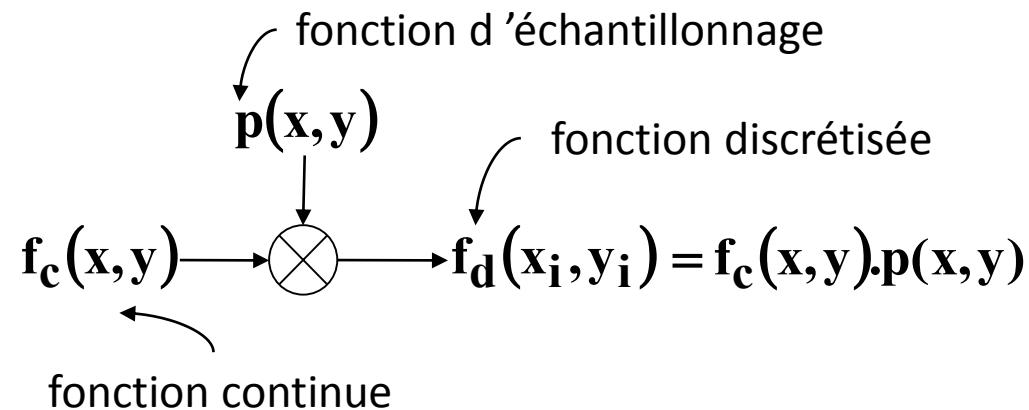


spectre

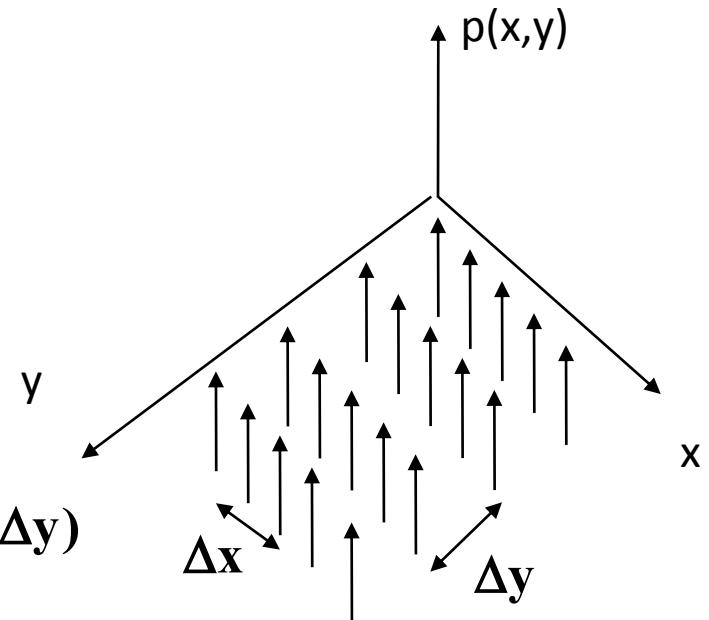


- Théorème d'échantillonnage

Une image à l'origine est généralement continue, elle devient numérique après échantillonnage
L'échantillonnage d'une fonction peut être modélisé par le système suivant :



$$p(x, y) = \delta_{\Delta x} \delta_{\Delta y}(x, y) = \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} \delta(x - i\Delta x) \delta(y - j\Delta y)$$



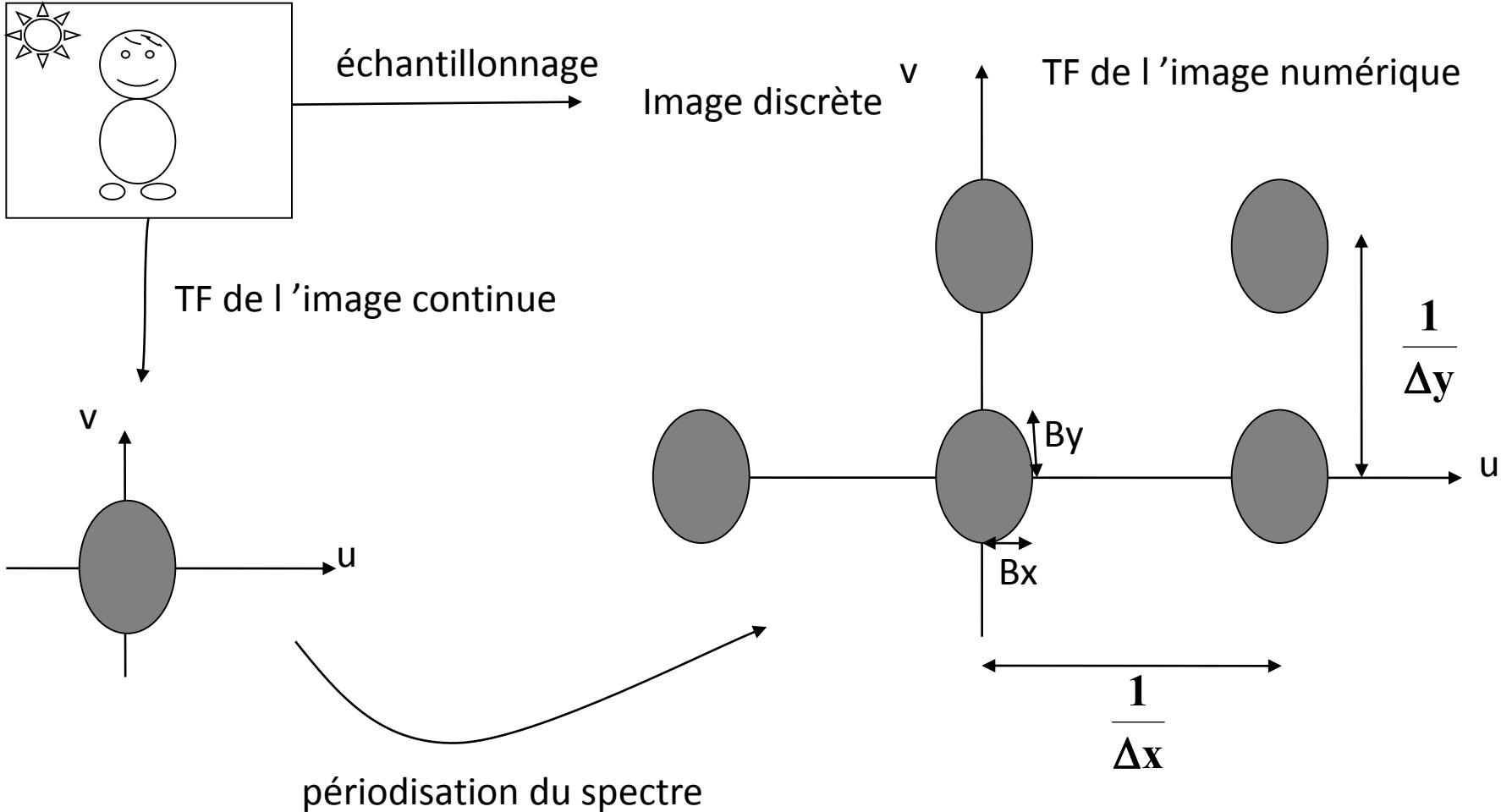
$F_c(u, v)$ existe (pas de pb en général car 1 image est un signal à énergie finie)

$P(u, v)$ existe et vaut : $P(u, v) = \frac{1}{\Delta x \Delta y} \delta_{\frac{1}{\Delta x}} \delta_{\frac{1}{\Delta y}}(u, v)$ (peigne de dirac)

donc $F_d(u, v) = F_c(u, v) * \frac{1}{\Delta x \Delta y} \delta_{\frac{1}{\Delta x}} \cdot \frac{1}{\Delta y}(u, v) \Rightarrow$

La TF est périodisée dans les 2 directions

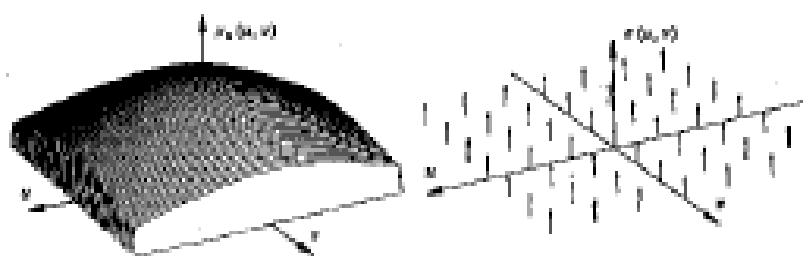




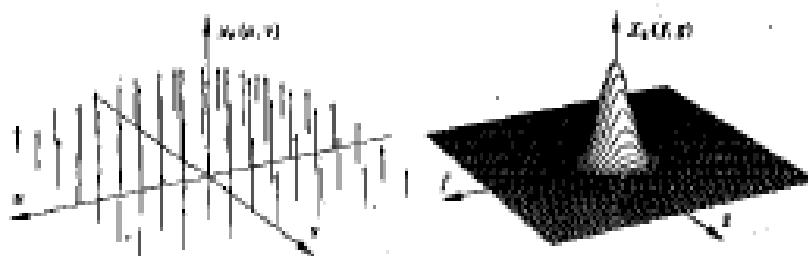
Shannon : $\frac{1}{\Delta x} = f_{e_x} \geq 2B_x$ et $\frac{1}{\Delta y} = f_{e_y} \geq 2B_y$



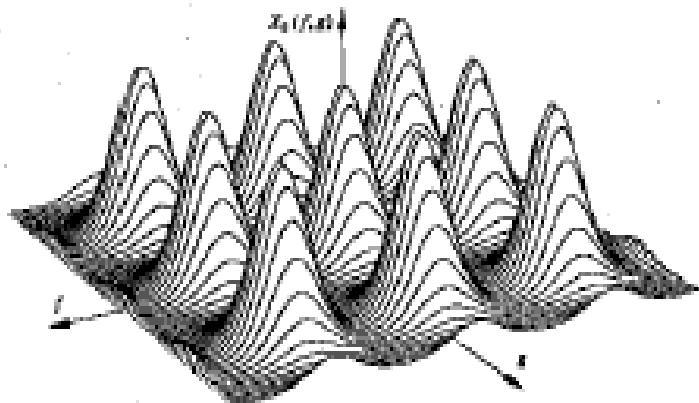
Signal analogique x_a



Signal échantillonné x_e



Spectre du signal échantillonné X_e



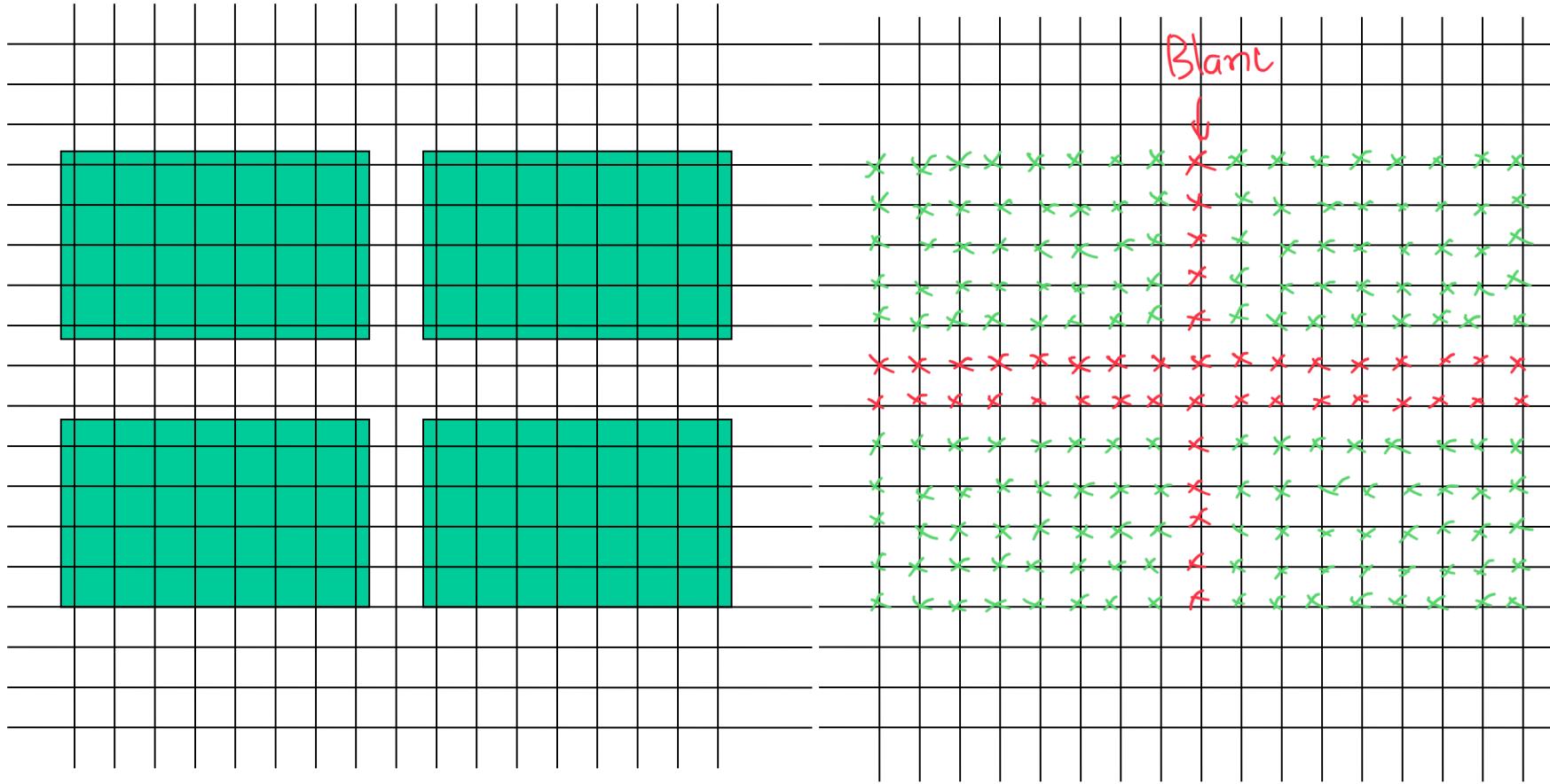
Grille d'échantillonnage e

Spectre du signal analogique X_a



Echantillonnage

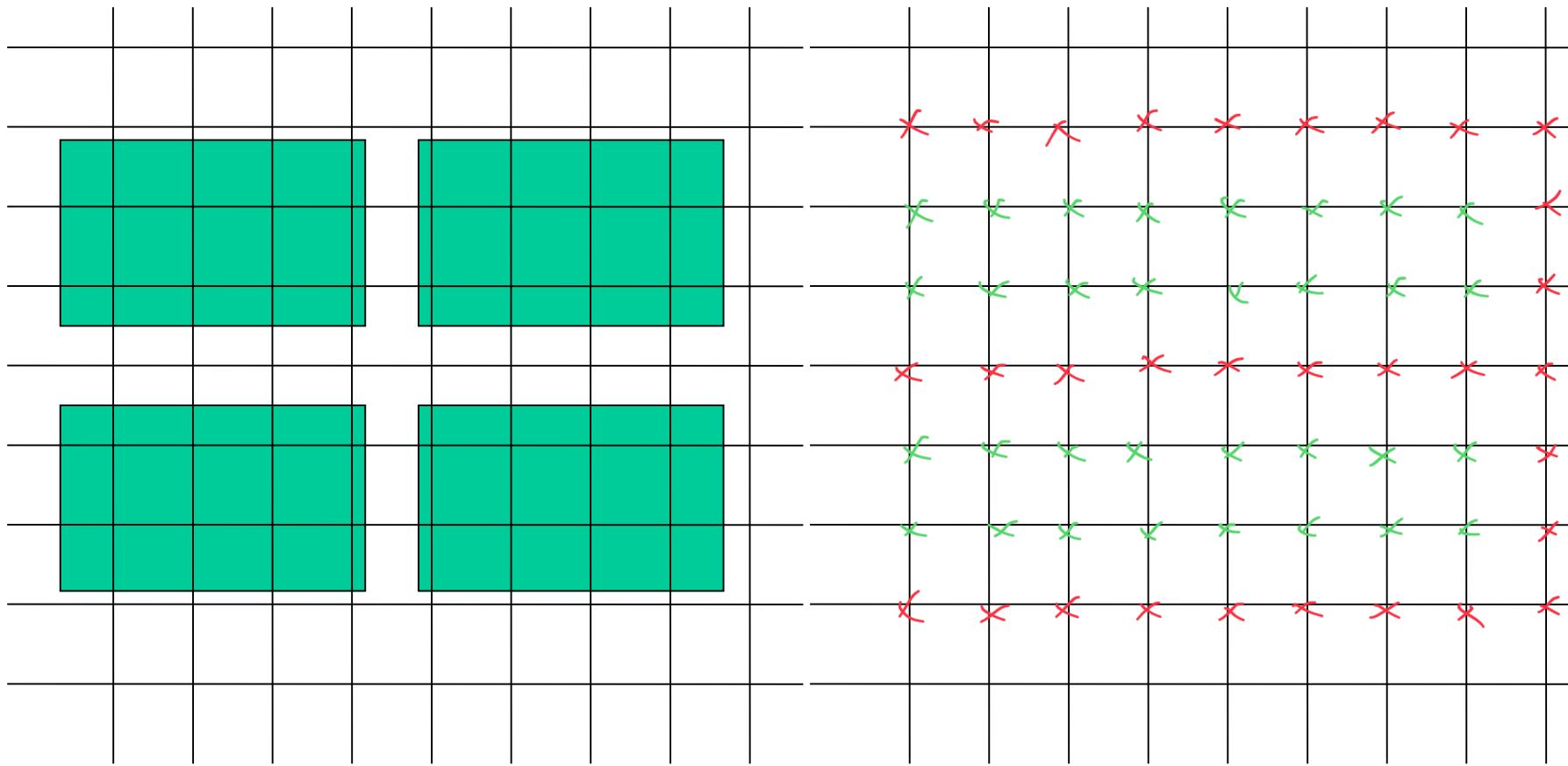
Exemple simple et pratique



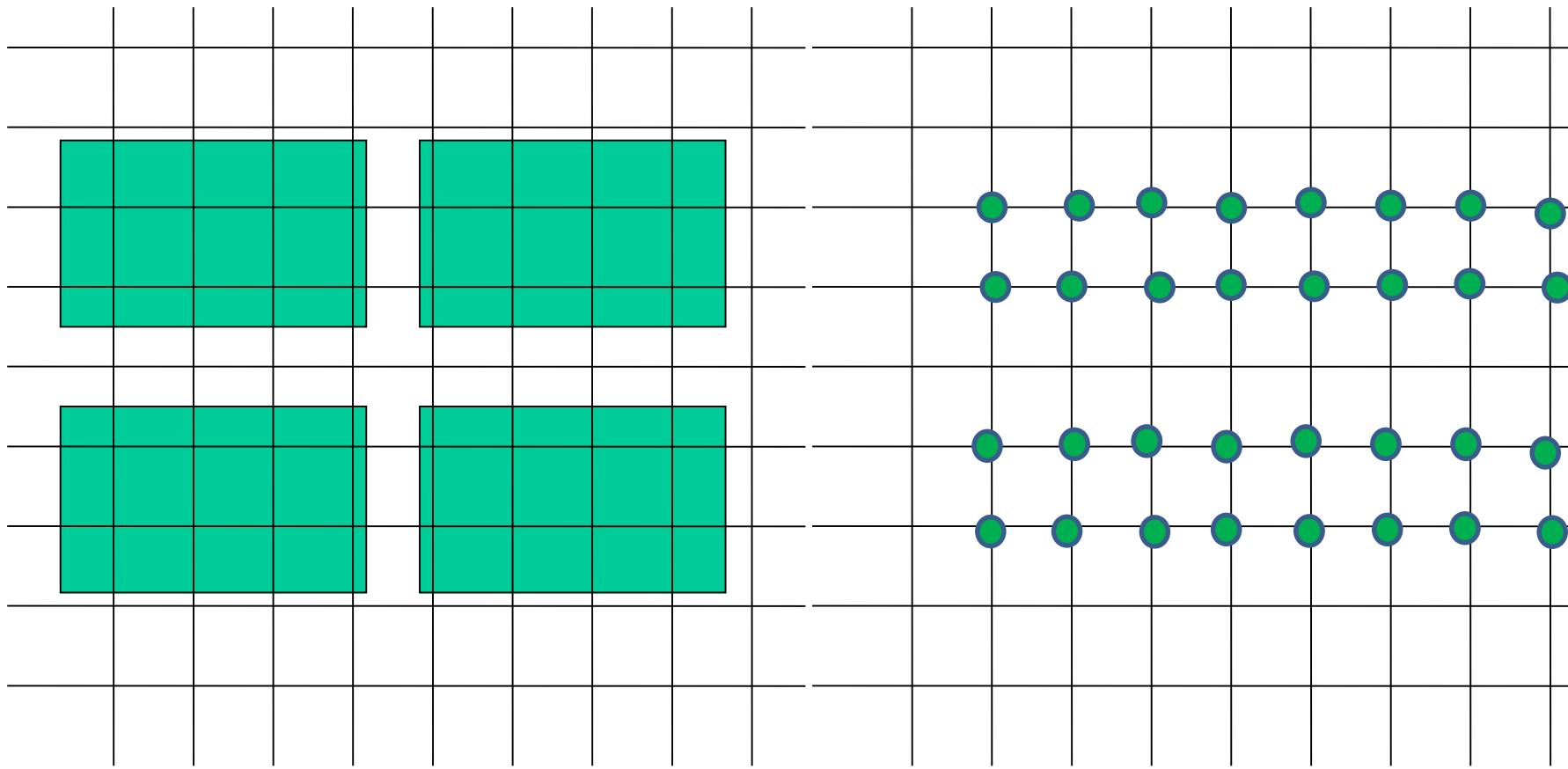
Points d'intersection des grilles : définissent la couleur du point disc



Echantillonnage

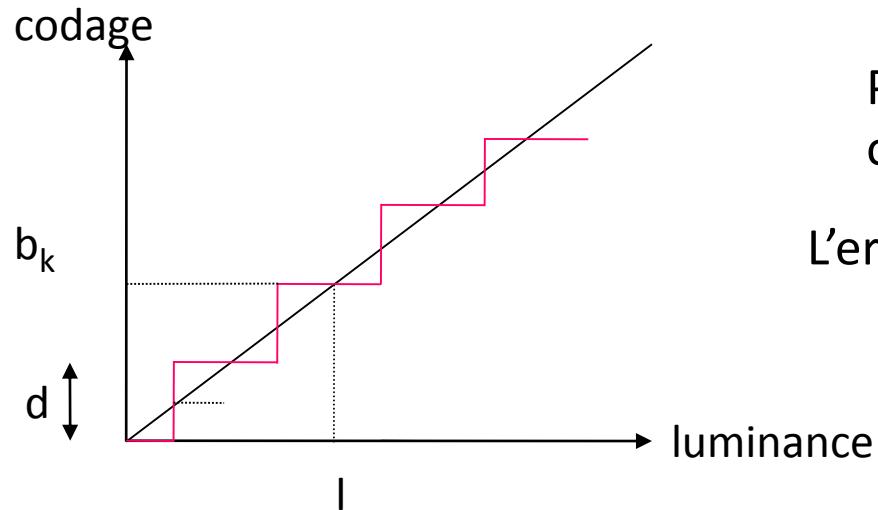


Echantillonnage



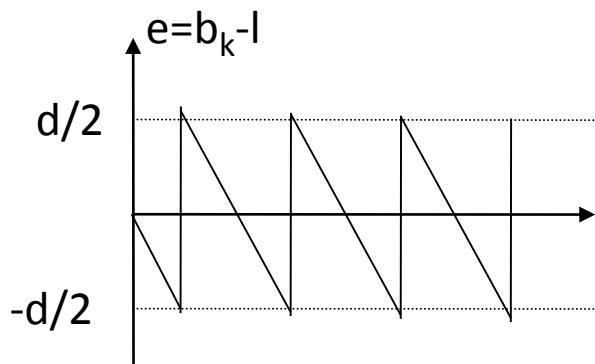
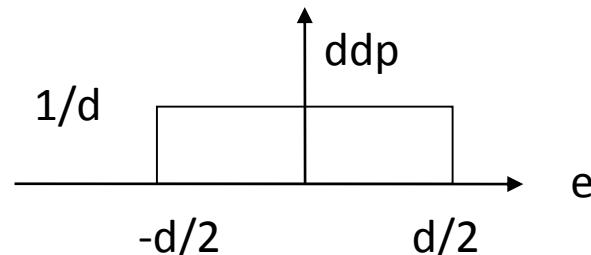
Quantification

Il s'agit de conversion classique analogique->numérique



d : pas de quantification
Pour une valeur I de luminance
on obtient un niveau b_k

L'erreur e est équiprobable entre $[-d/2; d/2]$



$$\bar{e} = 0$$

$$\bar{e}^2 = \frac{d^2}{12}$$

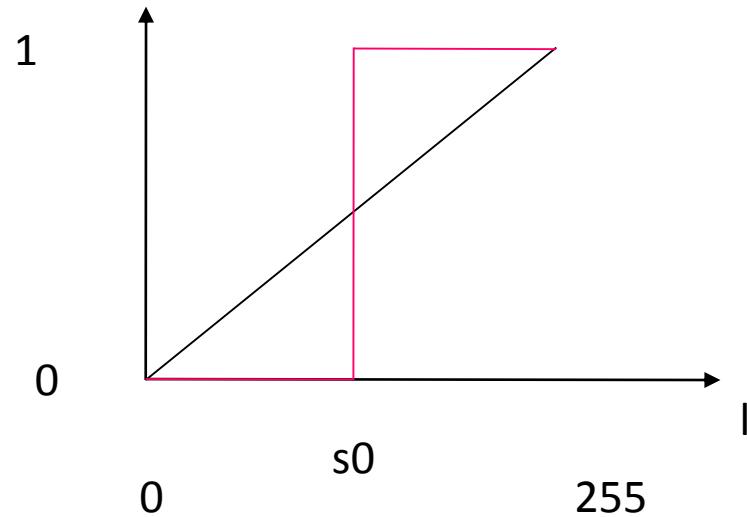
$$\text{RSB-} \rightarrow \frac{(2^n - 1)d}{\sqrt{12}} = \sqrt{12}(2^n - 1)$$



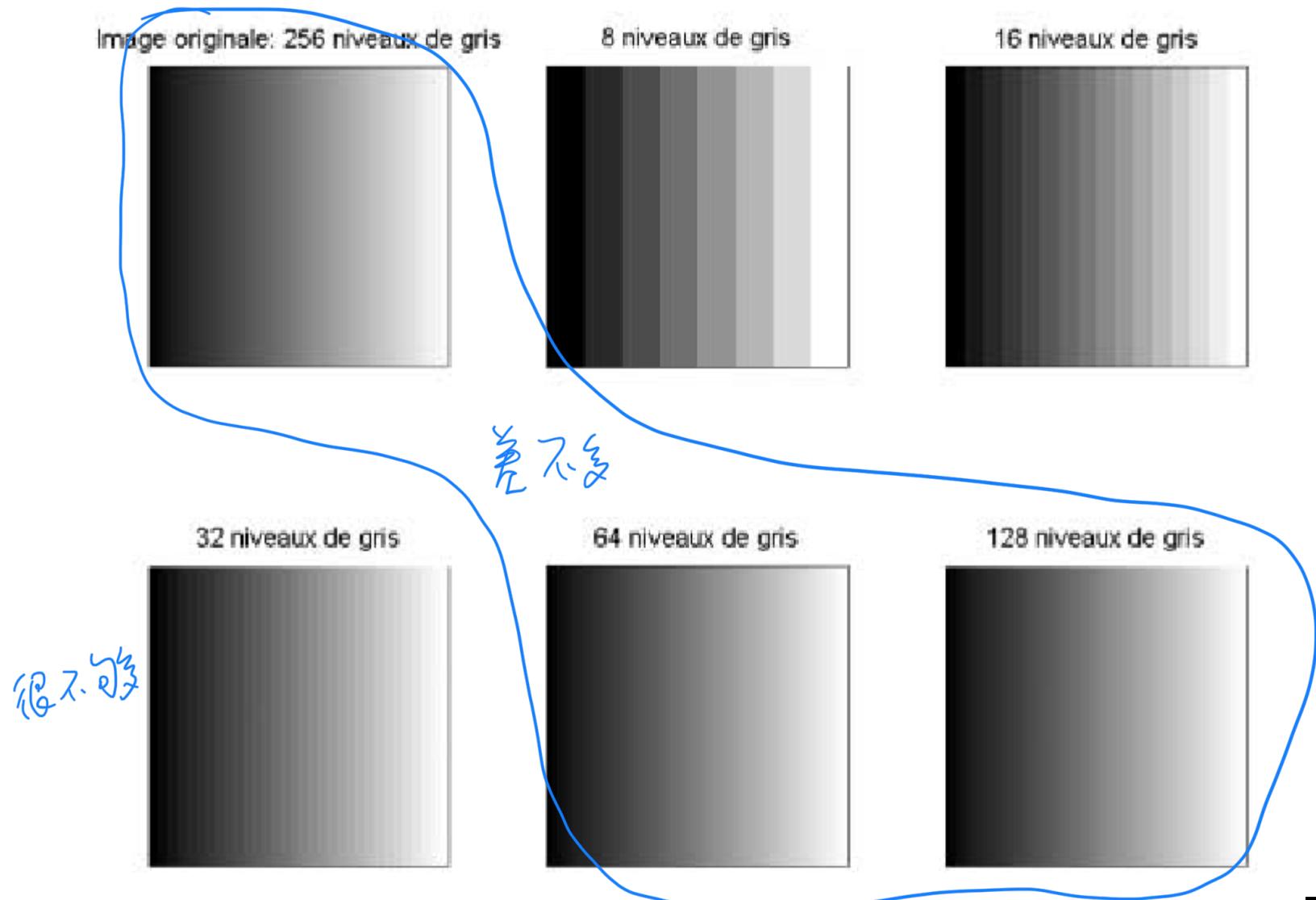
- Autres quantifications

Exemple : Seuillage

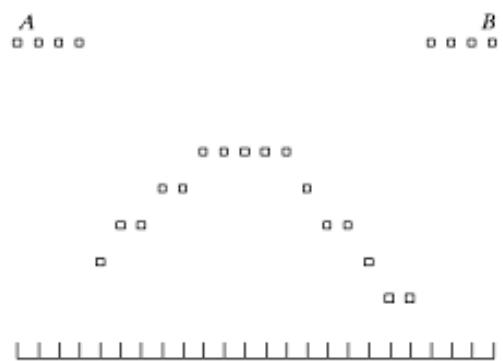
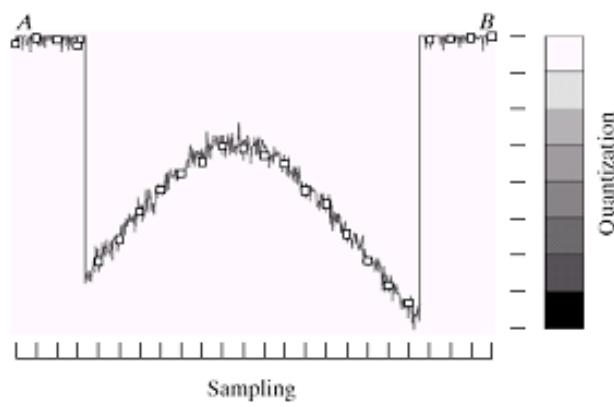
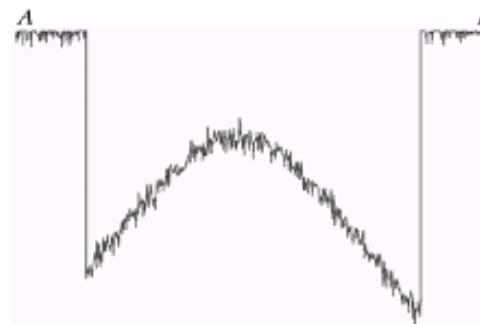
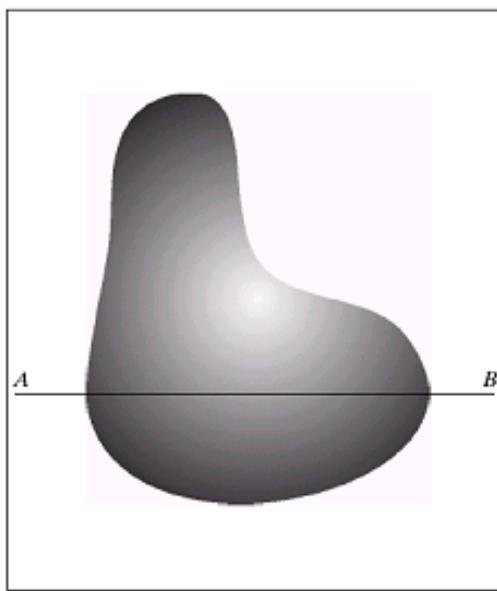
codage



Effet de la quantification



Échantillonnage et quantification : Résumé



Résolution des images

- Résolution spatiale
 - Le plus petit détail discernable
- Résolution de tons de gris
 - Le plus petit changement discernable
 - Normalement 8 bits (256 tons mais l'œil à partir de 64 niveaux ne discernent plus de différence)
- Une image a donc une résolution spatiale de $M \times N$ pixel et une résolution de tons de gris de K bit ou de L niveaux de gris ou tons

$$L = 2^k - 1$$



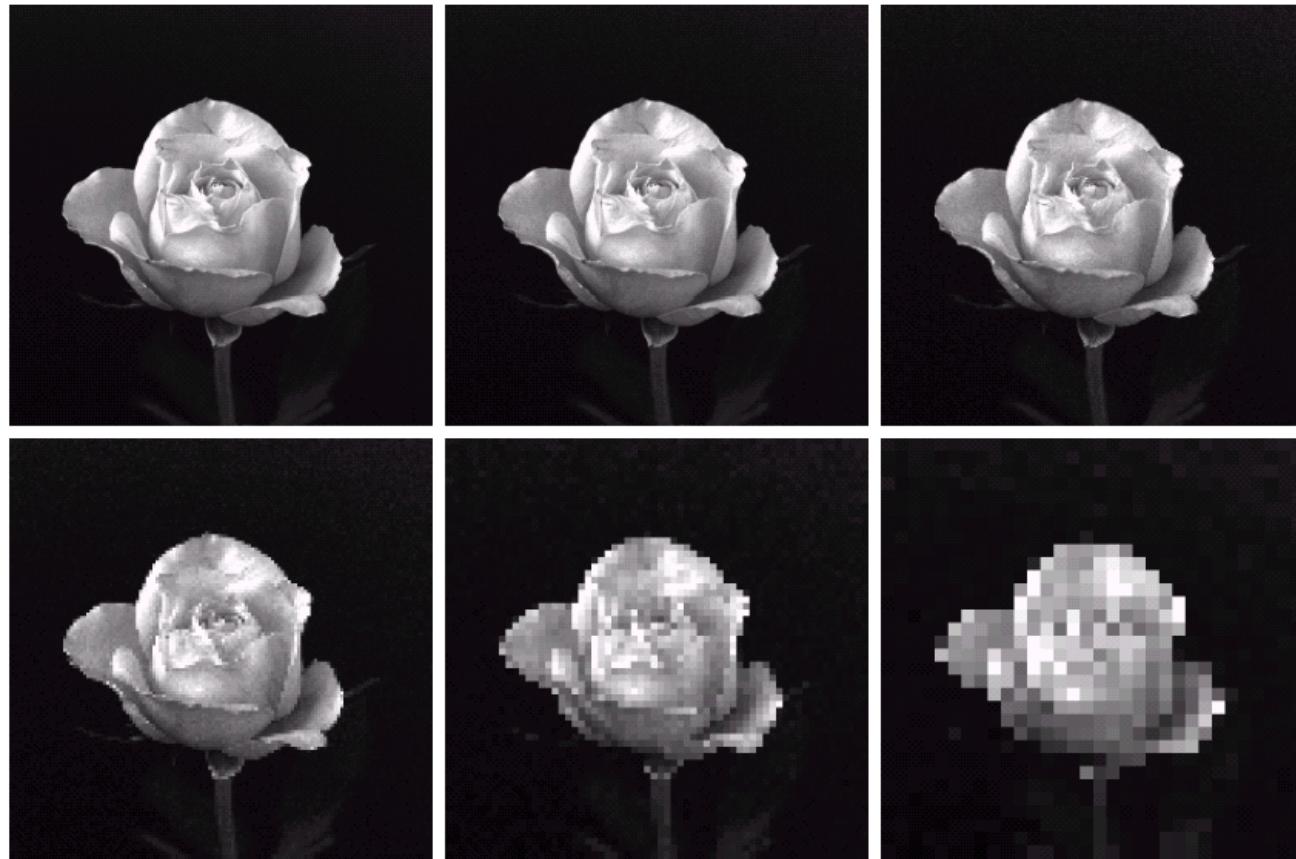
Résolution spatiale



FIGURE 2.19 A 1024×1024 , 8-bit image subsampled down to size 32×32 pixels. The number of allowable gray levels was kept at 256.



Résolution spatiale

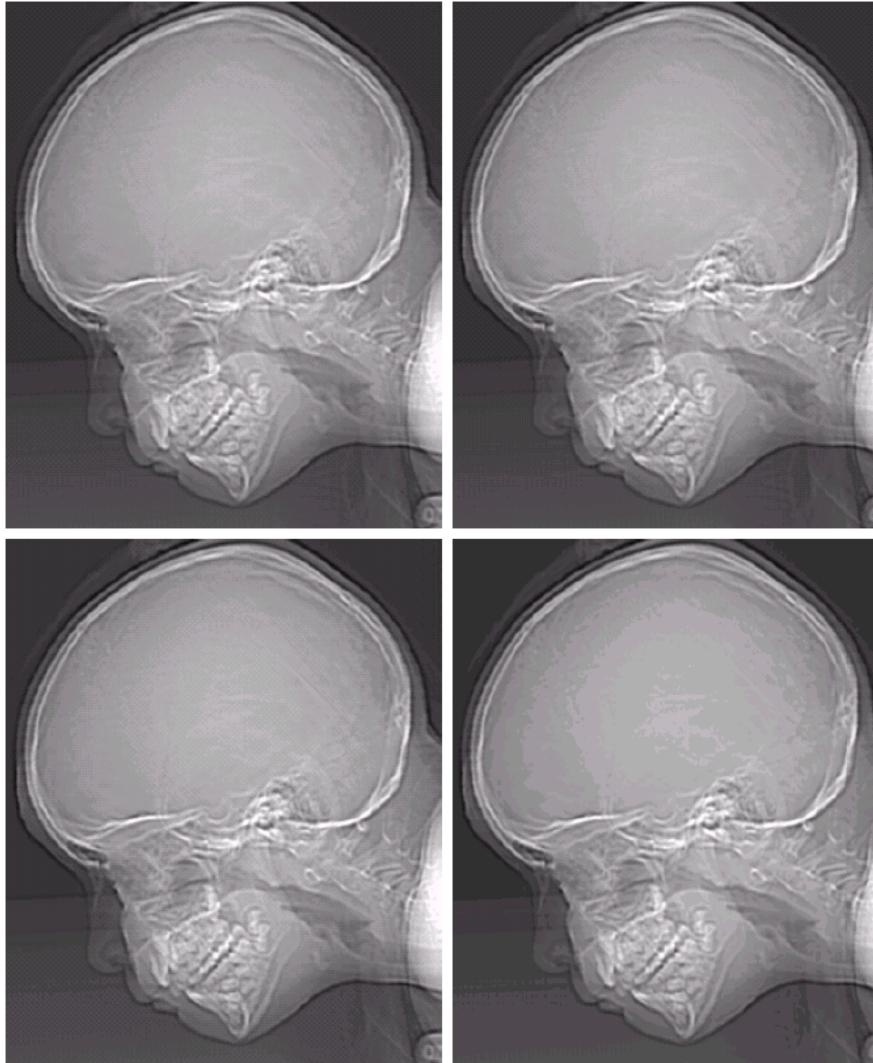


a	b	c
d	e	f

FIGURE 2.20 (a) 1024×1024 , 8-bit image. (b) 512×512 image resampled into 1024×1024 pixels by row and column duplication. (c) through (f) 256×256 , 128×128 , 64×64 , and 32×32 images resampled into 1024×1024 pixels.



Résolution de tons de gris



a b
c d

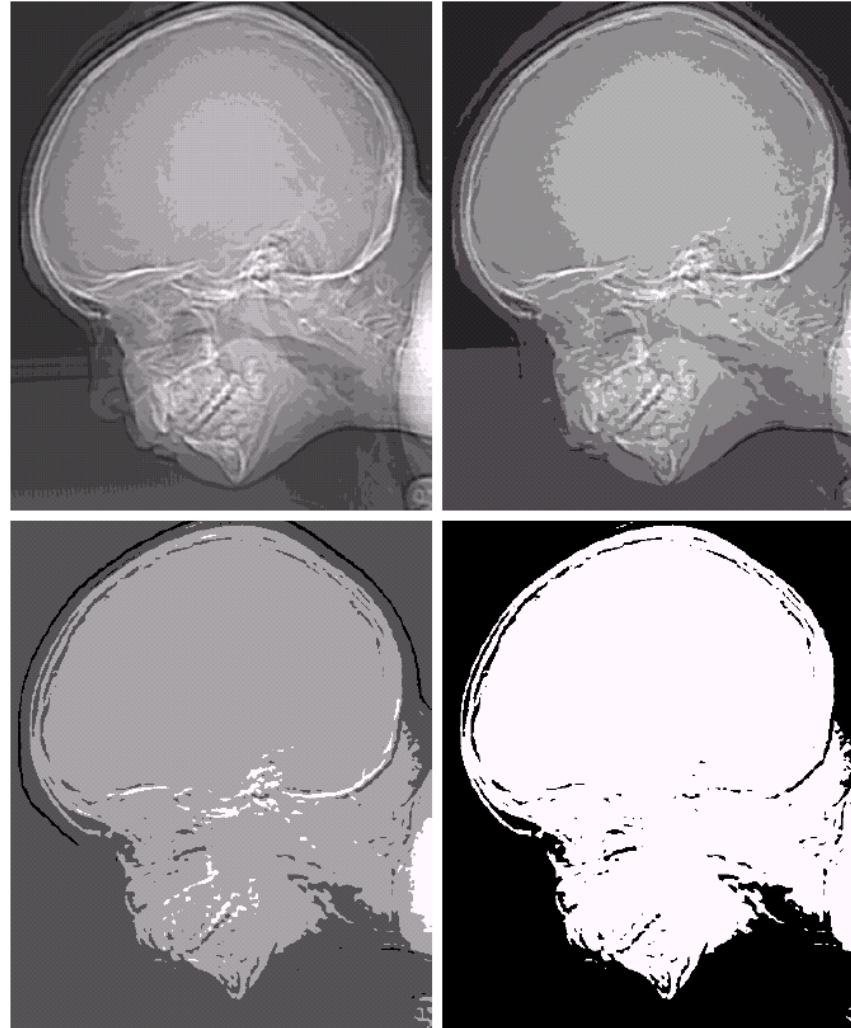
FIGURE 2.21
(a) 452×374 ,
256-level image.
(b)–(d) Image
displayed in 128,
64, and 32 gray
levels, while
keeping the
spatial resolution
constant.



Résolution de tons de gris

e f
g h

FIGURE 2.21
(Continued)
(e)–(h) Image displayed in 16, 8, 4, and 2 gray levels. (Original courtesy of Dr. David R. Pickens, Department of Radiology & Radiological Sciences, Vanderbilt University Medical Center.)



Résolution spatiale et tons de gris

- Courbes d'iso préférence



a b c

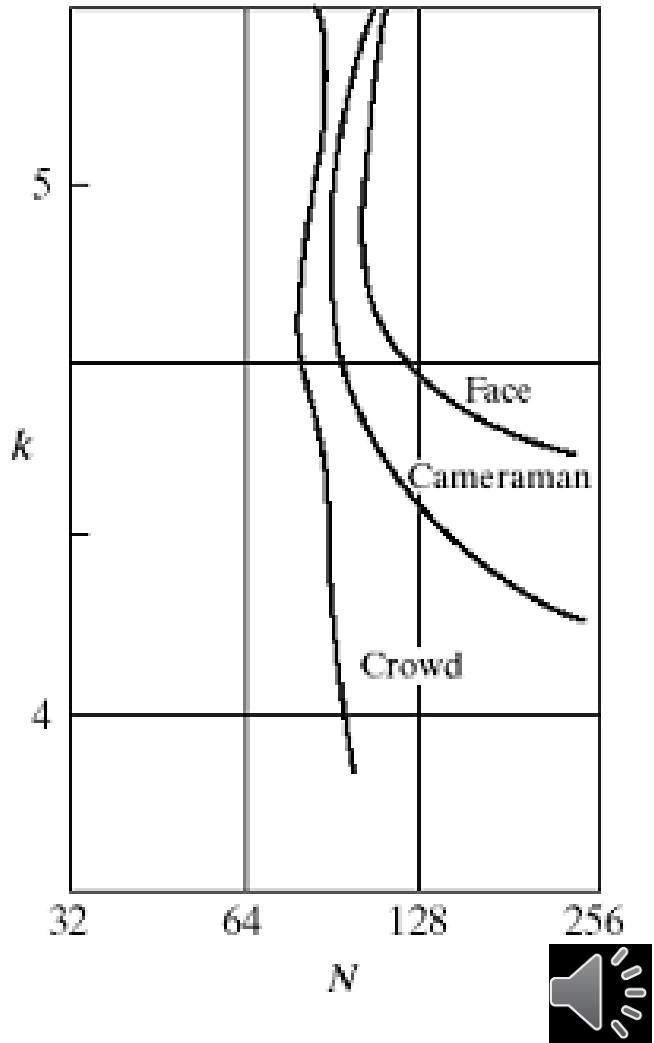
FIGURE 2.22 (a) Image with a low level of detail. (b) Image with a medium level of detail. (c) Image with a relatively large amount of detail. (Image (b) courtesy of the Massachusetts Institute of Technology.)

Ref : <http://iancat.tistory.com/64>



Résolution spatiale et tons de gris

- Courbes d'iso préférences
=Qualité subjective
- Moins de niveaux de gris nécessaires pour images plus détaillées
 - N: taille de l'image
 - Nbr de niveau de gris = 2^k



Changement de résolution

- Zoom !
- Interpolation du plus proche voisin
= RéPLICATION des Pixels
 - Copie chaque colonne et chaque rang



images zoomed from 128×128 , 64×64 , and 32×32 pixels to 1024×1024 pixels.



Changement de résolution

- Zoom !
- Interpolation bilinéaire
- Fonction bilinéaire de 4 pixels voisins

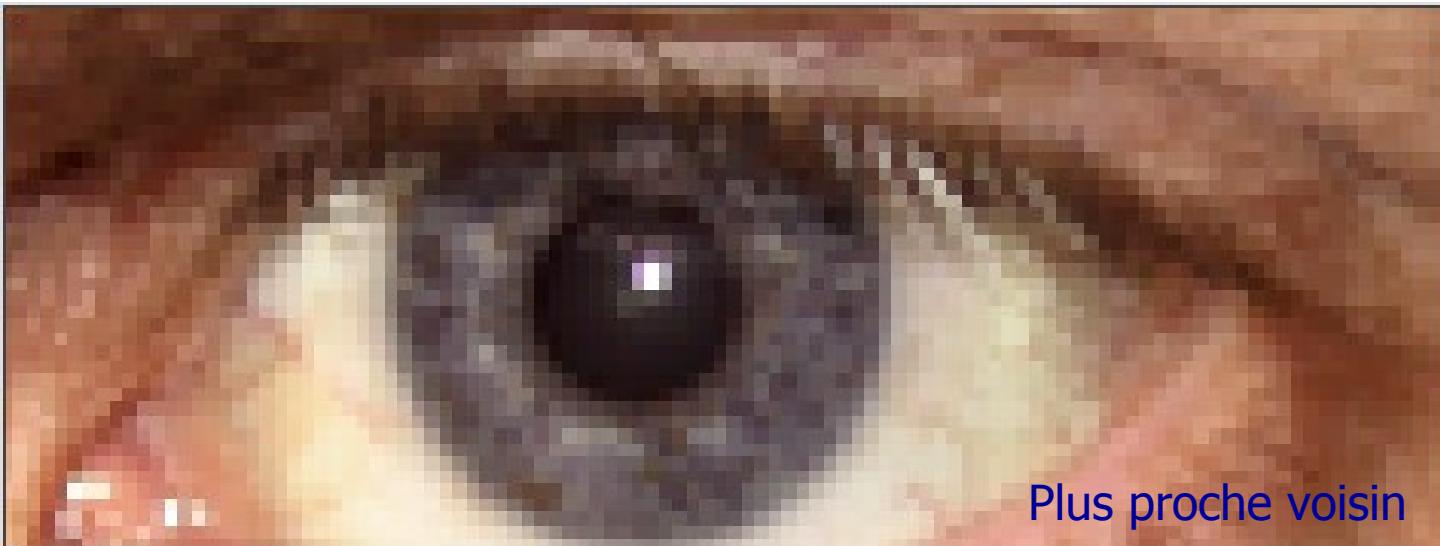


images zoomed from 128×128 , 64×64 , and 32×32 pixels to 1024×1024 pixels,





Changement de résolution



Plus proche voisin



Bilinéaire (4 voisins)





Changement de résolution



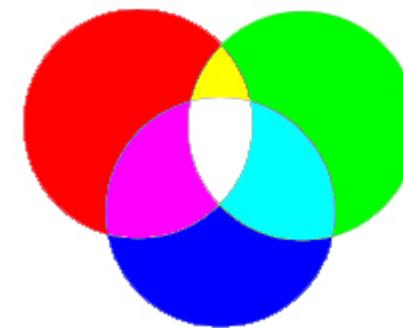
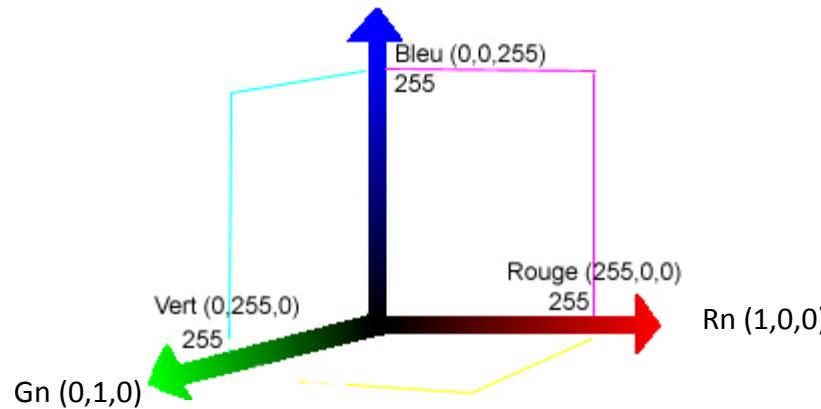
Représentation de la couleur

- Espace RGB ou RVB (rouge, vert, bleu) :

- Les caméras standard => en sortie trois canaux RVB => la couleur d'un point est reconstituée par synthèse additive de ces trois canaux.
- Normalisation des composantes :

$$R_n = R/(R+G+B), G_n = G/(R+G+B), B_n = B/(R+G+B)$$

- Représentation du Cube RnGnBn :



- Espace YUV

- Système standard de codage pour la télévision européenne.
- Les chrominances U et V s'obtiennent à partir des composantes R_n, B_n, G_n grâce à la transformation suivante :

$$\begin{bmatrix} \mathbf{Y} \\ \mathbf{U} \\ \mathbf{V} \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.148 & -0.289 & 0.437 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} \mathbf{R}_n \\ \mathbf{G}_n \\ \mathbf{B}_n \end{bmatrix}$$



- L'espace Lab
 - Il a été créé par la Commission Internationale de l'Eclairage (C.I.E.) pour définir une norme internationale en matière de vision des couleurs perçues par l'œil humain. C'est la seule mesure fiable des couleurs, elle se fait à l'aide d'un spectroréfractomètre. Pour l'obtenir, on fait une conversion de l'espace RGB vers l'espace XYZ normalisé par la C.I.E en 1931, puis de XYZ vers L.a.b. Le système L.a.b. possède le grand avantage d'être uniforme. Il est très utile dans le cas de mélange de pigments, par exemple pour l'industrie graphique, du textile ou le secteur de la peinture.

$$L = 116 * Y^{1/3} - 16$$

$$a = 500 * \left[X^{1/3} - Y^{1/3} \right]$$

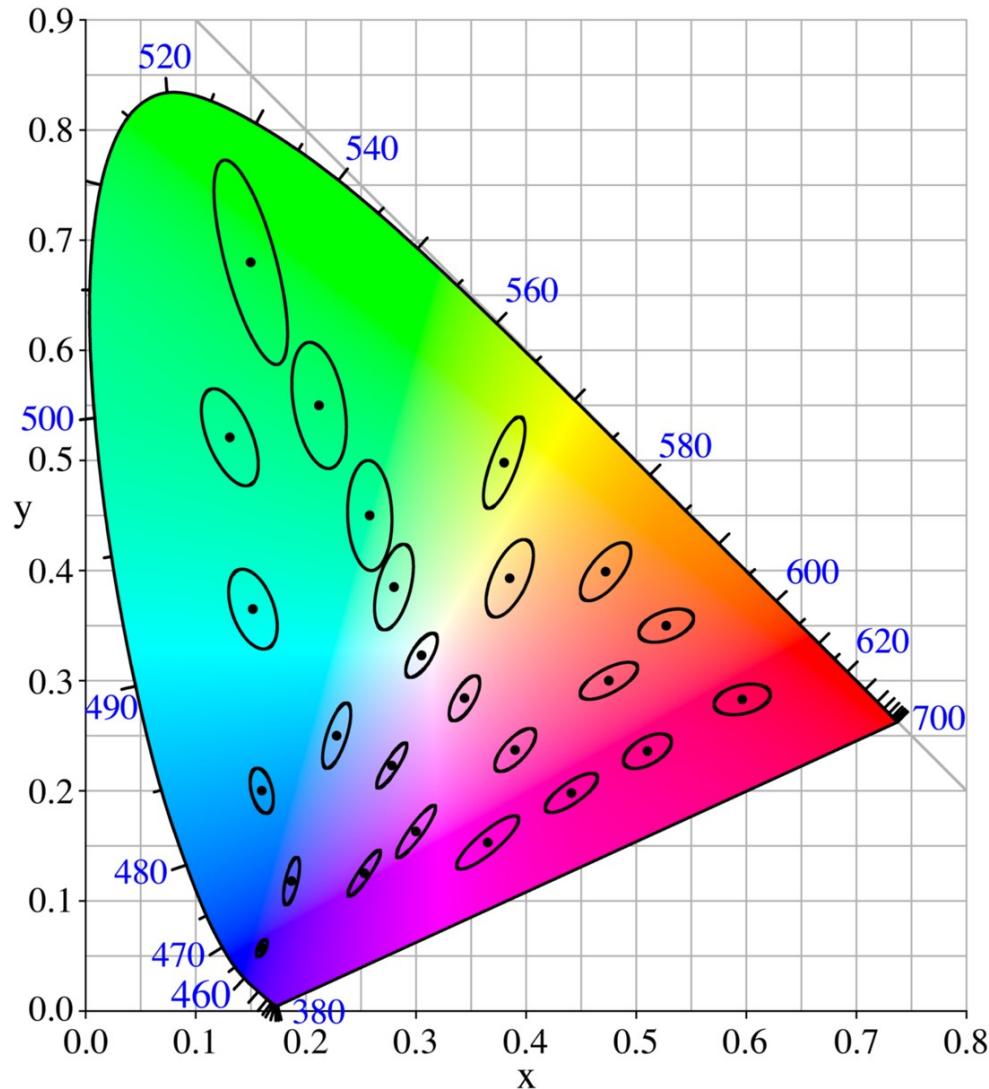
$$b = 200 * \left[Y^{1/3} - Z^{1/3} \right]$$

avec X,Y,Z tels que :

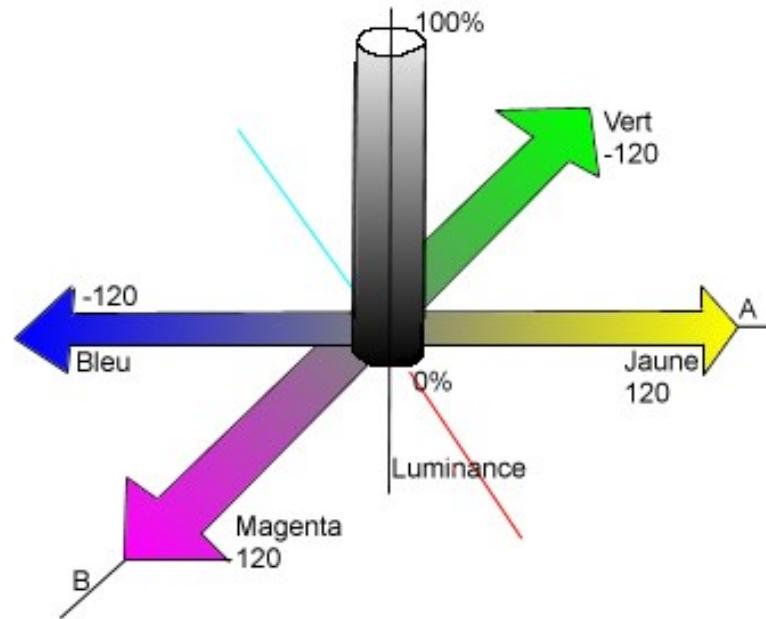
$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.490 & 0.310 & 0.200 \\ 0.117 & 0.813 & 0.011 \\ 0.000 & 0.010 & 0.990 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$



Ellipse de Macadam dans xyY

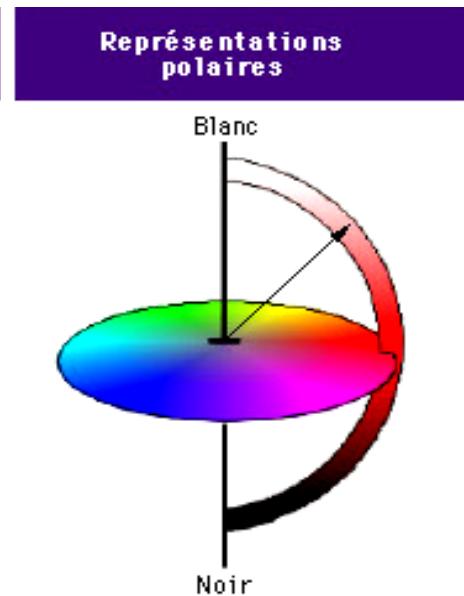
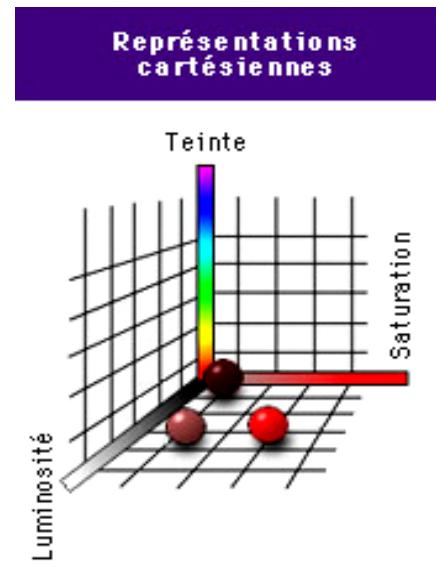
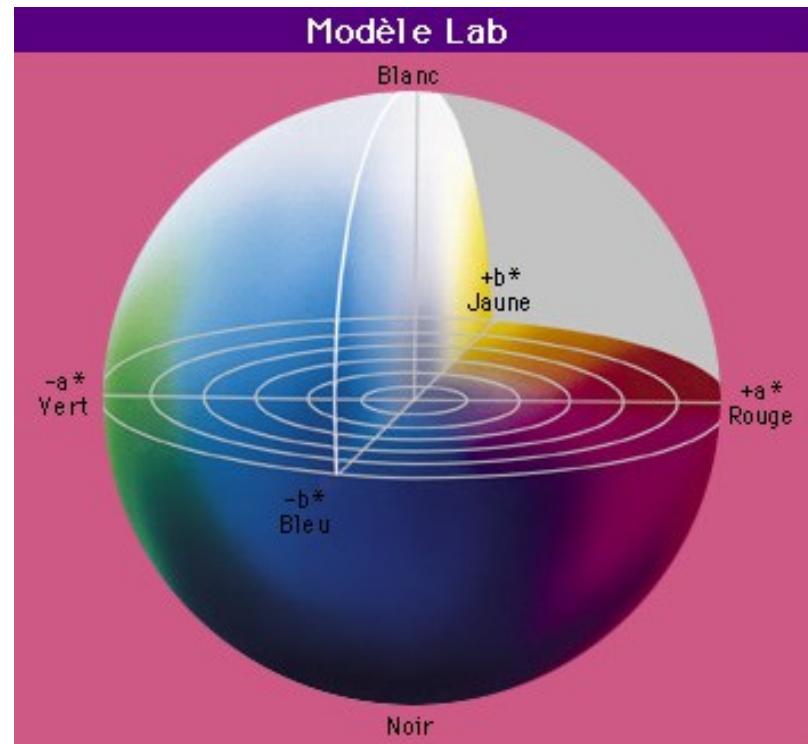

$$x = X/(X+Y+Z),$$
$$y = Y/(X+Y+Z),$$
$$z = Z/(X+Y+Z) = 1 - x - y$$


Les couleurs : LAB



- ⇒ Luminance
- ⇒ A : mélange de vert et de magenta
- ⇒ B : mélange de bleu et de jaune





- Luminosité et teinte mesurées de manière angulaire
- Saturation mesurée de manière cartésienne

Lab utilise un système mixte de repérage des points de couleur :
une représentation cartésienne et une représentation polaire



- Espace HSI

L'espace HSI (H=hue=Teinte, S=Saturation, I=Intensité) est relié à l'espace RGB-CIE par des relations non linéaires à savoir :

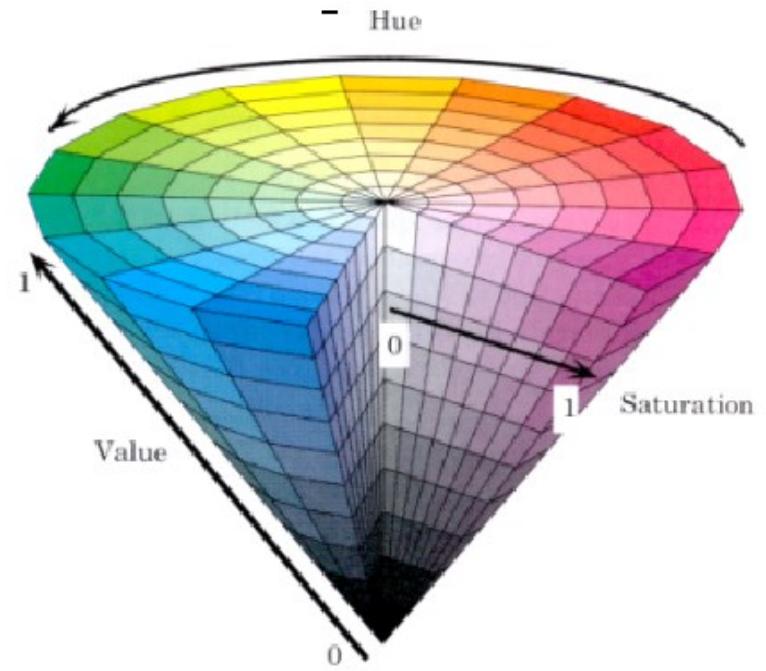
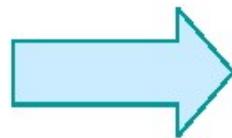
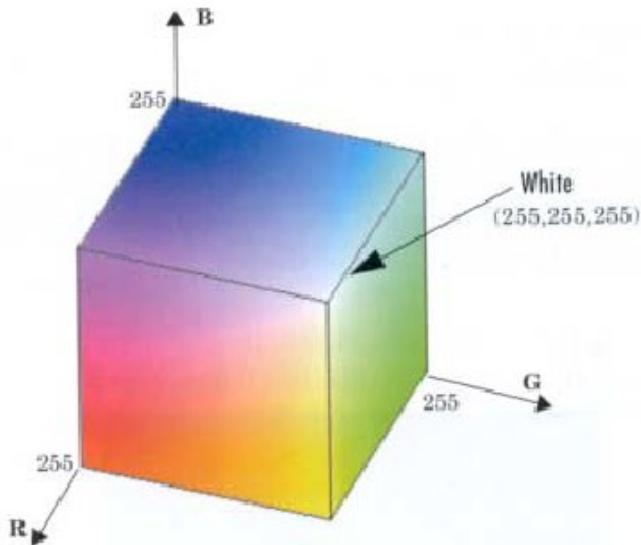
$$I = \frac{1}{3} * (R + G + B)$$

$$S = 1 - \frac{3 * \min(R, G, B)}{R + G + B}$$

$$H = \begin{cases} \cos^{-1} \left\{ \frac{[(R - G) + (R - B)]}{2 * \sqrt{[(R - G)^2 + (R - B)(G - B)]^{1/2}}} \right\} & \text{Si } G > B \\ 2\pi - \cos^{-1} \left\{ \frac{[(R - G) + (R - B)]}{2 * \sqrt{[(R - G)^2 + (R - B)(G - B)]^{1/2}}} \right\} & \text{Si } B > G \end{cases}$$

On remarque que la saturation S n'est pas définie quand I=0 et que la teinte H n'est pas définie si S=0.





Système RGB

Système HSV



- Espace TSL

L'espace TSL est une variante de l'espace HSI avec des coefficients de conversion différents. Le modèle TSL (Teinte, Saturation et Luminosité) est également utilisé par de nombreux programmes de traitement électronique de l'image.

On passe de l'espace RGB vers l'espace TSL selon les relations suivantes :

$$L = 0.299 * R + 0.587 * G + 0.114 * B$$

$$S = \left[9/5 * (r'^2 + g'^2) \right]^{1/2}$$

$$r' = r - 1/3, g' = g - 1/3, r = \frac{R}{R + G + B} \text{ and, } g = \frac{G}{R + G + B}$$

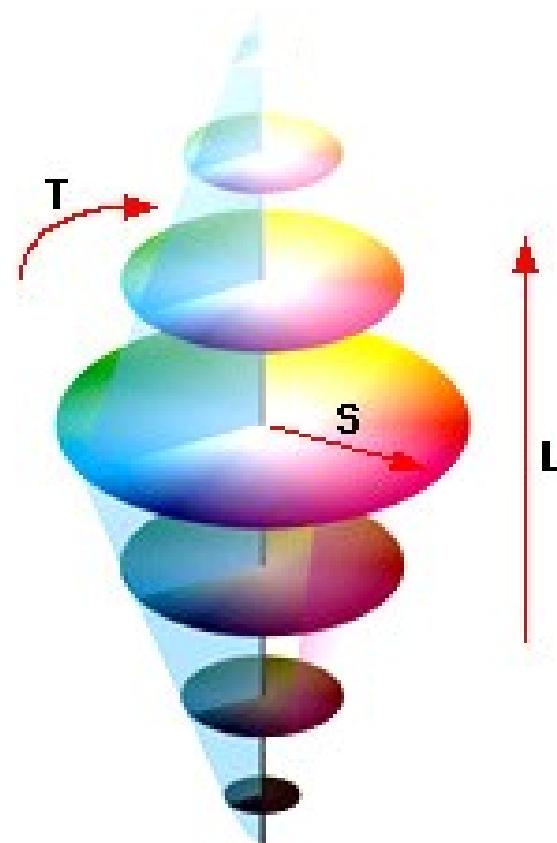
$$T = \begin{cases} \arctan \frac{r'}{\frac{g'}{2\pi} + \frac{1}{4}}, & g' > 0 \\ \arctan \frac{r'}{\frac{g'}{2\pi} + \frac{3}{4}}, & g' < 0 \\ 0, & g' = 0 \end{cases}$$





Teintes →

Teinte , saturation et luminosité



Les images 3D

- Quelques notions de base
- Acquisition : Images et vidéos 3D
- Les différents display 3D
- Retrouver les niveaux de profondeur d'une scène



Acquisition - stéréovision

- Principe de la vision humaine
- Acquérir en 3D pour voir en 3D: pas besoin de calcul de profondeur (depth map) ni de notion de géométrie de scènes, mais on peut avoir certains besoins pour synchroniser les deux images.
- La depth map utile pour convertir du 2D en 3D
- Acquérir pour avoir une vision 3D de la scène au sens mathématique : calcul de profondeur (depth map), scène 3D obtenue avec les paramètres des caméras, des références géométriques de la scène.

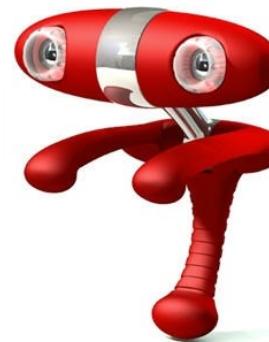


Dispositifs d'acquisition simple

Dès 2010



FUJIFILM FinePix REAL 3D W3



Webcam 3D Minoru



65mm (35mm camera equiv.)

* When setting the aspect ratio at 16:9 with DMC-GH2.

LUMIX G 12.5mm / F12

(H-FT012)



Dispositifs d'affichage du 3D

- Ecran avec lunettes :
 - Anaglyphe (lunettes R-B)
 - Lunettes polarisées (cinema)
 - Lunettes à Shutter (pour TV 3D actuelle)
- Ecran autostéréodisplay
 - Principe de réseaux lenticulaires
 - Principe avec barrière parallaxe





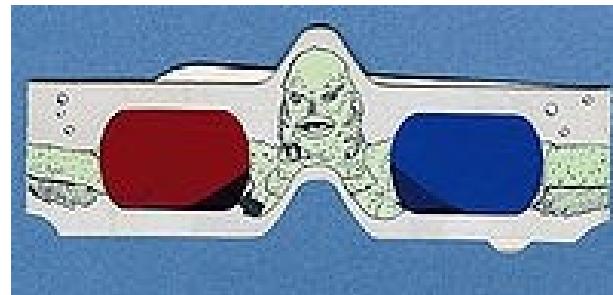
Public Library, Stereoscopic Looking Room, Chicago, by Phillips, 1923



L'anaglyphe

- Principe de base : l'image est composée de deux images : l'image de gauche est passée par le filtre rouge, l'image de droite est passée par le filtre bleue.

- $\text{lf}(:,:,1)=\text{lg}(:,:,1);$
- $\text{lf}(:,:,2)=\text{id}(:,:,2);$
- $\text{lf}(:,:,3)=\text{id}(:,:,3);$





Moniteurs ou TV 3D avec lunettes

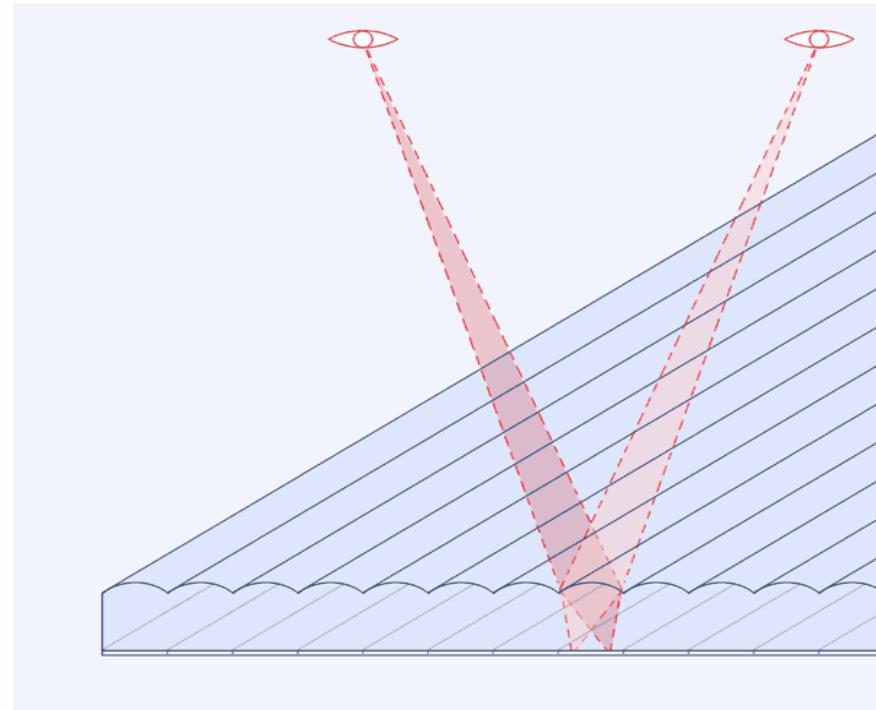
- Lunettes passives ou actives
 - Lunettes actives : shutter, une image est projetée sur le verre gauche pendant que le verre droit est fermé et vice-versa. Synchronisée sur la TV3D. Très cher, mais meilleur.
 - Lunettes passives, la TV 3D envoie deux images suivant deux polarisations différentes. Moins cher, mais couleur et contraste très fade...



Auto-stéréo display

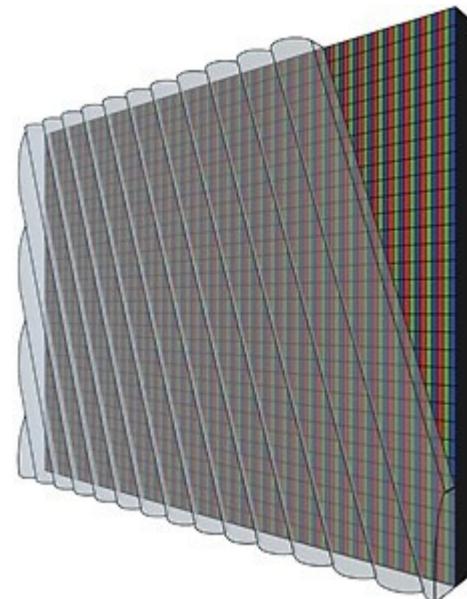
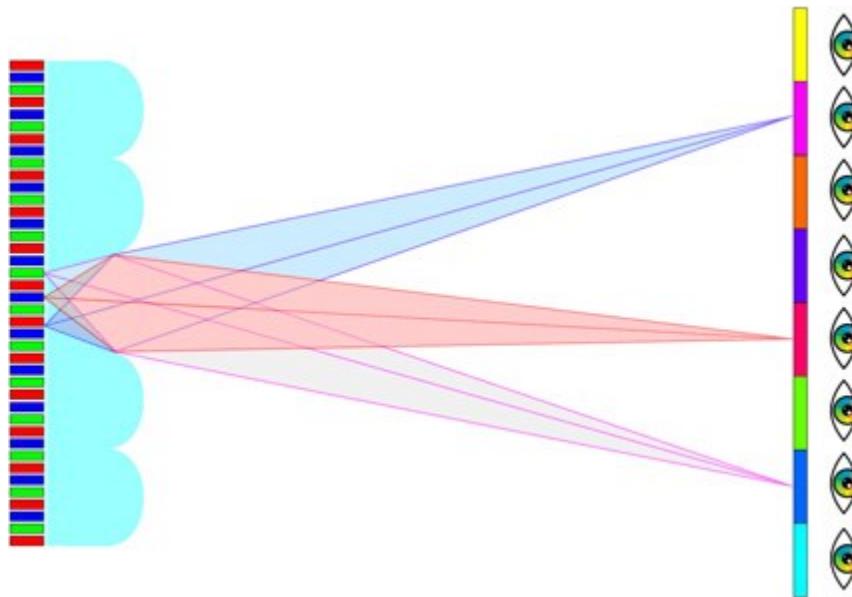
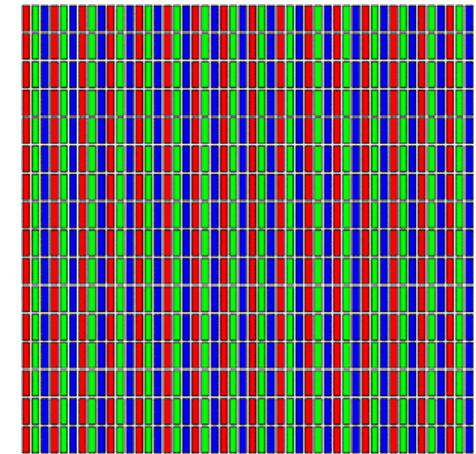
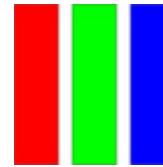
Réseau lenticulaire

- Principe ancien (Lippman 1915), s'utilisait et s'utilise en impression encore actuellement. Le principe a été étendu aux écrans



L'écran auto-stéréo display

- Un pixel : 3 couleurs
- Une matrice HD LCD
- Un réseau de micro-lentilles cylindriques est placé sur la matrice LCD / ou prisme de Fresnel
- Philips : Des TV 3d sans lunettes et de qualité professionnelle 2013 .
- **2017 des annonces pour une TV3D**



Sources : <http://www.alioscopy.com/Overview>

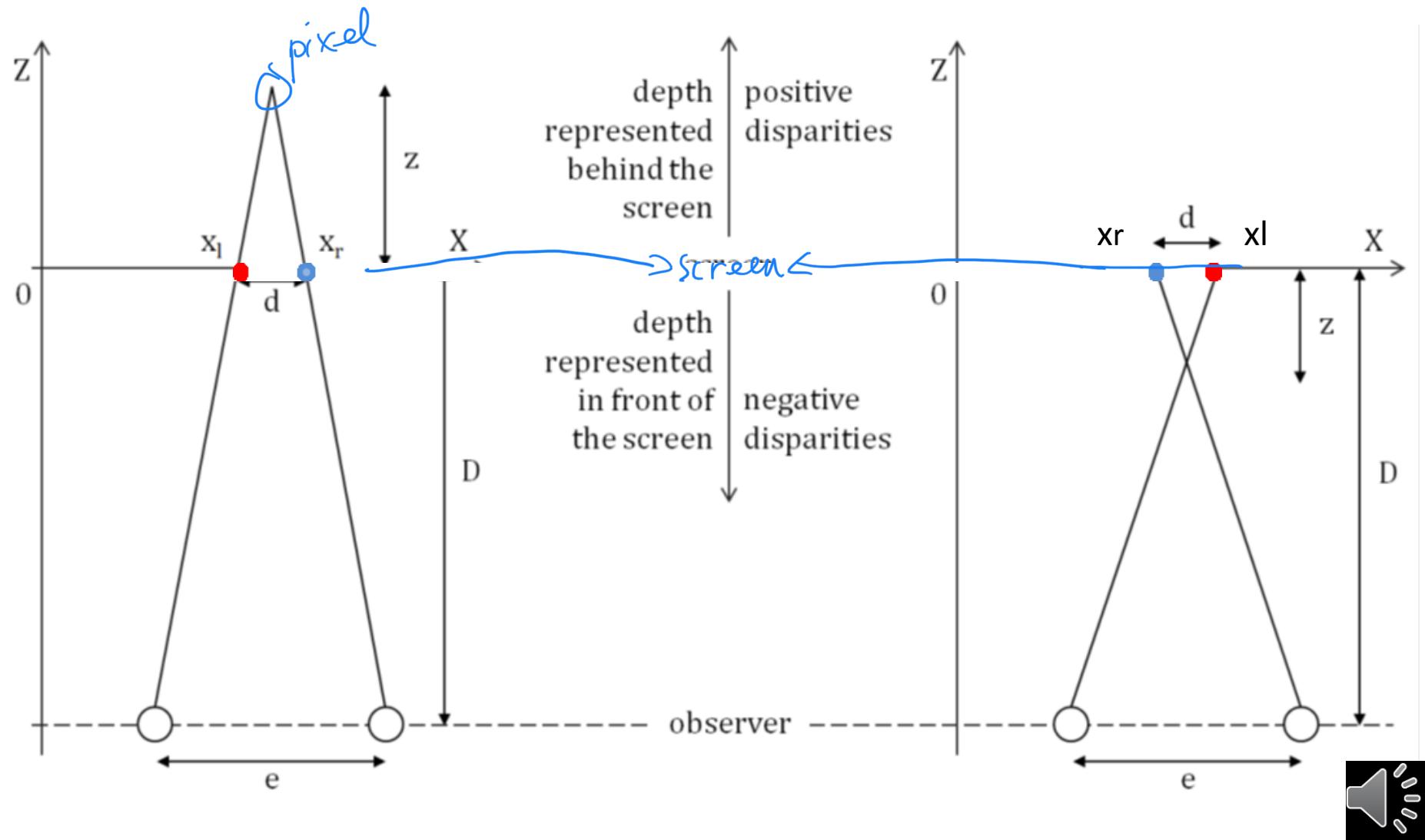


Exemple écran auto-stéréo display

- Ecran de la nintendo DS4 ou de l'appareil photo Fujifilm



Perception de la profondeur : principe



Perception profondeur

O : plan de l'écran

D : distance de l'observateur à l'écran

e : distance inter-pupilles

d : distance sur l'écran séparant la position d'un même point de la scène sur l'image gauche et l'image droite xl (x left) et xr (x right), donc d= xr-xl. Le z alors perçu est obtenu par :

$$z = \frac{d \cdot D}{e + d}$$

Suivant le signe de d (et la convention choisie), on peut alors percevoir l'objet derrière l'écran ou devant l'écran.



Utilitaire pour gérer la profondeur apparente et recaler les images

[Programme pour recaler les images L-R](#)



Calcul de la carte des profondeurs

- A partir des deux vues (gauche et droite)
 - Calculer les différents niveaux de profondeurs de la scène.
 - On cherche à calculer la différence de position $d=x_r-x_l$ pour chaque point de la scène.

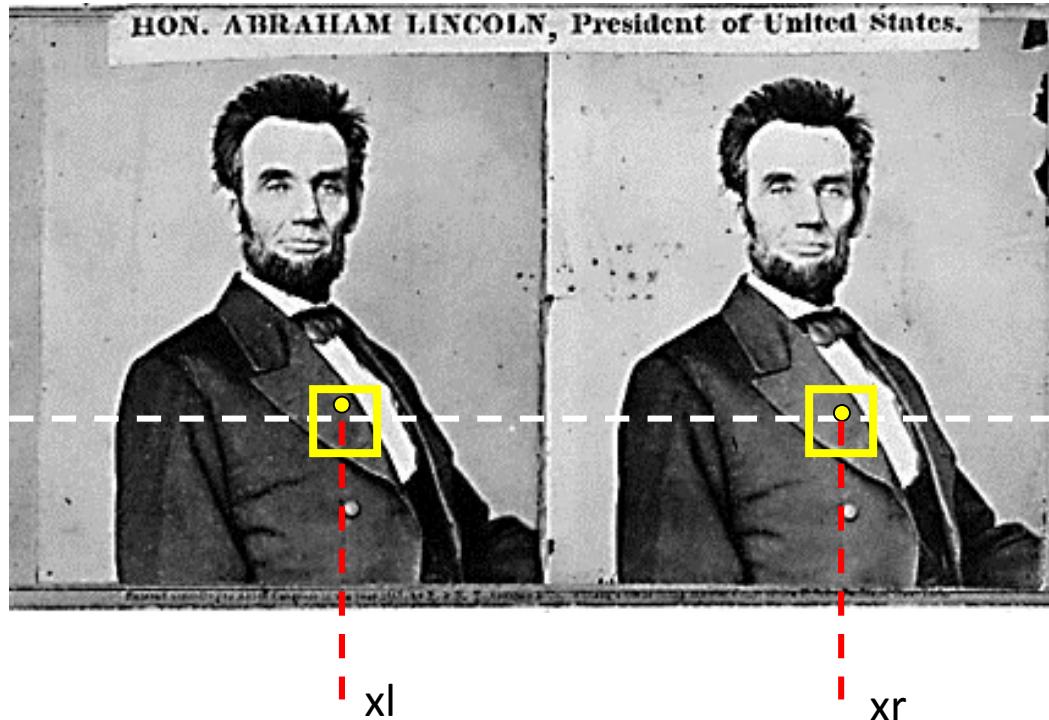


Les contraintes photométriques

- A priori le même point a la même intensité dans la vue gauche et droite et il est positionné sur la même ligne (les deux capteurs sont alignés).
- La différence de position nous donnera une information sur la position z du point dans l'image.
 - C'est le point le plus ressemblant dans la seconde image (critère de ressemblance)
 - C'est le plus près des points ressemblants



Exemple



Ligne épipolaire

Pour chaque ligne épipolaire

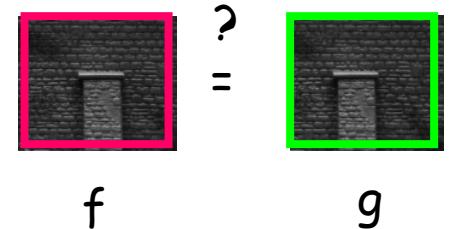
Pour chaque pixel dans l'image gauche

- compare avec chaque pixel sur la même ligne épipolaire dans l'image droite
- On recherche le pixel qui correspond le mieux
- Evidemment ça ne peut pas marcher

Amélioration : Faire correspondre des fenêtres de taille suffisante



On compare des fenêtres:



$$SSD = \sum_{[i,j] \in R} (f(i,j) - g(i,j))^2$$
$$C_{fg} = \sum_{[i,j] \in R} f(i,j)g(i,j)$$

Fonctions de coût
Les plus populaires

Pour chaque fenêtre on cherche celle positionnée la plus proche qui donne un minimum pour la fonction de coût choisie.

Il s'agit de l'auto-corrélation

Demo matlab

