



Traitement Numérique des Données

M1 – INF 2163
**AIDN: Applications Interactives et
Données Numériques**
Sylvie Gibet

1

Traitement des images

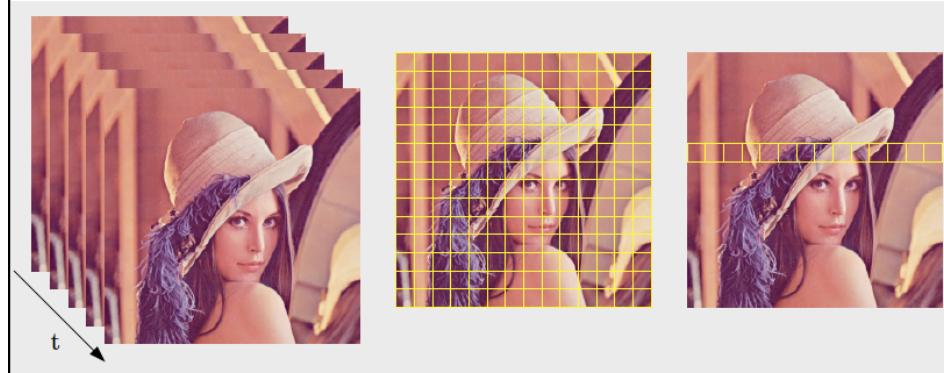
2

1

Monde réel -> monde numérique

Transformation de l'espace 3D en un tableau de valeurs

- **Formation** de l'image (projection) à travers un système optique
- **Echantillonnage temporel, spatial et spectral** par le **capteur**
- **Numérisation** par le capteur et / ou le système de traitement



Echantillonnage temporel

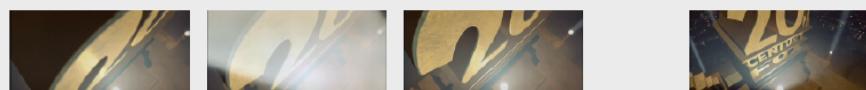
Cadence d'acquisition des images

- On ne peut pas mesurer l'information lumineuse de façon **continue** dans le temps, donc on prend des **échantillons**.
- Ainsi, on mesure **périodiquement** l'information lumineuse.
- La **fréquence** d'acquisition est l'inverse de la **période** d'acquisition.

Exemple

- Fréquence vidéo en France : 25 images / s, période de 40ms.

image 1 image 2 image 3 image 25



40 ms

1 s

4

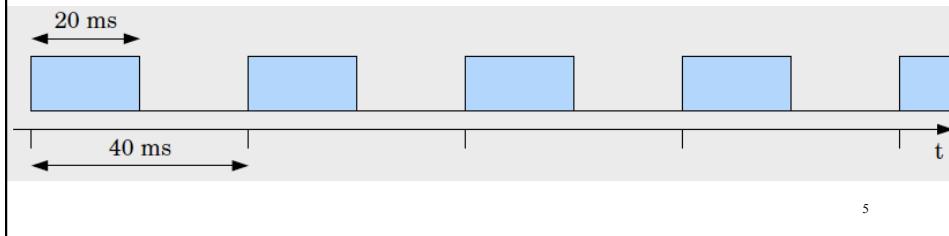
Intégration de l'information lumineuse

Durée d'intégration

- Un capteur élémentaire devrait théoriquement mesurer instantanément le **flux énergétique** qu'il reçoit, c'est à dire l'énergie par unité de temps ($\text{W} = \text{J/s}$).
- En pratique, un capteur mesure l'énergie totale (en J) qu'il reçoit pendant un intervalle de temps (en s) de durée non nulle, appelé **temps d'intégration**.

Exemple

- Fréq. d'acquisition = 25 im/s, durée d'intégration = 20 ms



5

Compromis sensibilité / flou de bougé

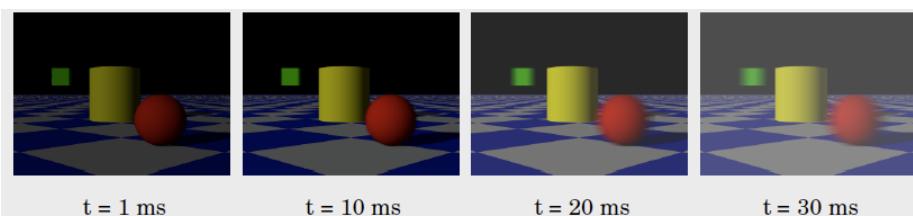
Pourquoi ?

En **augmentant** le temps d'intégration, on améliore la **sensibilité**.

Si un objet **bouge**, son image bouge également sur le capteur. Dans ce cas, le capteur accumule des mesures **non constantes**, ce qui entraîne un phénomène appelé **flou de bougé**.

Un **point mobile** apparaît comme une **ligne** dans l'image.

Exemple



6

Image continue : fonction de 2 variables

■ Fonction image

fonction $I : \mathbb{R}^2 \rightarrow \mathbb{R}^n$, $(x,y) \rightarrow I(x,y)$

n est le nombre de **composantes** de l'image

dans ce cas, l'espace \mathbb{R}^2 est appelé plan image

■ Support image

sous-ensemble de \mathbb{R}^2 de définition de I, de surface **finie**, en général un **rectangle**, de côtés T_x et T_y

■ Exemple à 1 composante

Pour x et y appartenant à $[-\pi, +\pi]$

$$I(x,y) = e^{-(x^2+y^2)} \cos(x) \cdot \sin(y)$$

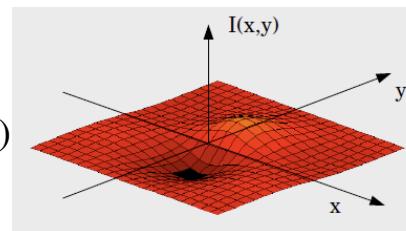


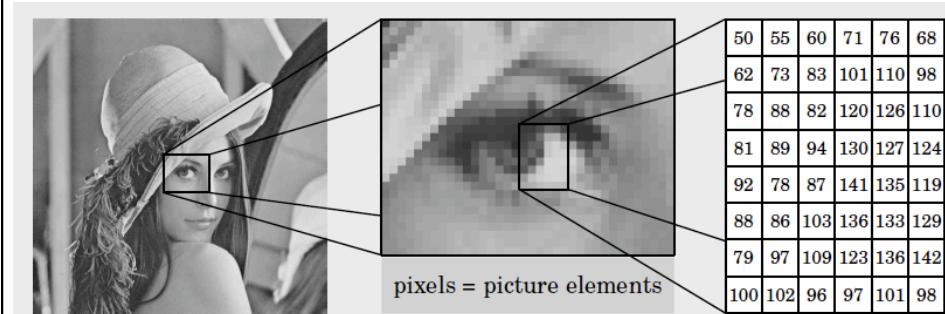
Image numérique : tableau bidimensionnel de valeurs

■ Echantillonnage de la fonction image

Echantillonnage en x et y de la fonction : ne conserve que les valeurs pour les points $(x,y) = (c \cdot \Delta x, l \cdot \Delta y)$, avec c et l entiers

Tableau de valeurs I : $\mathbb{Z}^2 \rightarrow \mathbb{R}^n$, $(c,l) \rightarrow I(c \cdot \Delta x, l \cdot \Delta y)$

■ Exemple à 1 composante

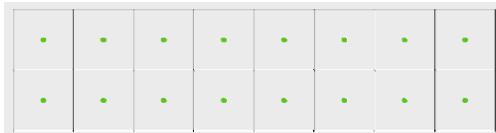


Echantillonnage spatial et résolution

■ Type d'échantillonnage spatial

Parfait : la valeur d'un pixel est une mesure du flux énergétique

Reçu par un point sensible (de surface nulle) du capteur.



Réel : la valeur d'un pixel est une mesure du flux énergétique

Reçu par une surface sensible élémentaire. C'est le produit de l'éclairement énergétique par la surface de l'élément sensible.



9

Echantillonnage spatial et résolution

Résolution d'une image

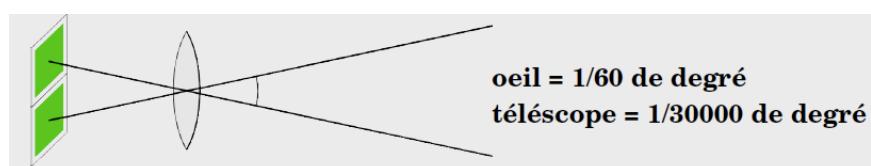
Nombre de pixels par unité de longueur. La résolution s'exprime en pixels / m (des fois en pixels / inch).

La résolution est l'inverse du pas de la grille d'échantillonnage.

La résolution est le rapport du nombre de pixels divisé par la dimension du capteur.

Pouvoir de résolution

Mesure, par un angle, la capacité d'un système d'acquisition d'image (optique + capteur) à distinguer des détails fins.



Echantillonnage spatial et résolution



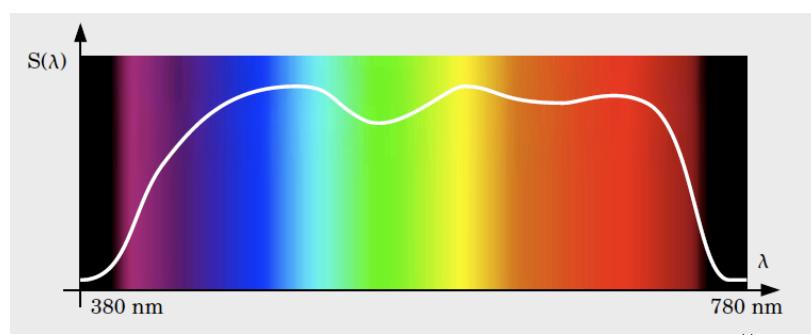
11

Echantillonnage spectral

■ Image monochrome

Pour chaque pixel on mesure le **flux énergétique total**, c'est à dire pour **toutes** les longueurs d'onde du spectre.

En pratique, un capteur monochromatique est caractérisé par une courbe de **sensibilité** spectrale.



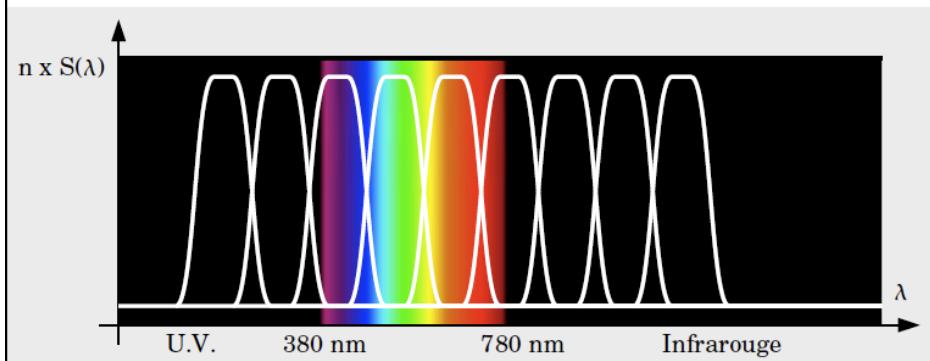
12

Echantillonnage spectral : image hyperspectrale

Échantillonnage du spectre en n bandes

Pour acquérir une **composante**, on utilise un capteur qui a une **sensibilité spectrale** correspondant à une **bande** du spectre.

Une **composante** est mesurée par l'intégrale du **flux énergétique monochromatique**, pondérée par la sensibilité spectrale.

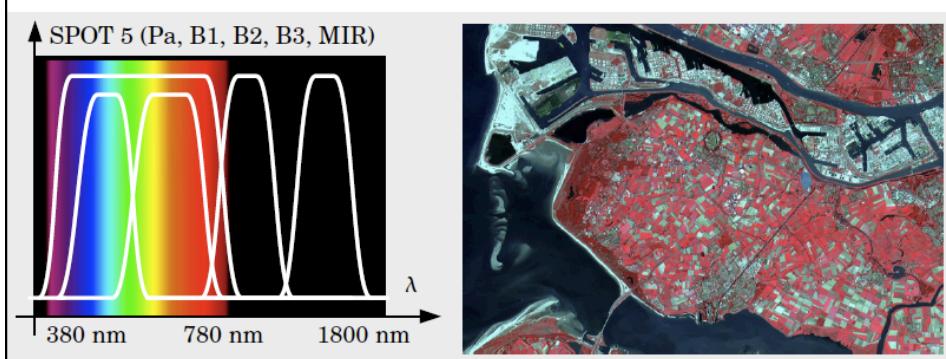


Echantillonnage spectral : image hyperspectrale

Image multispectrale

Les bandes sont en **nombre limité** et sont choisies parce que leurs longueurs d'ondes ont des propriétés particulières.

Exemple : **imagerie satellitaire**, la surface de la terre est observée dans le **visible** et l'**infrarouge** (végétation)

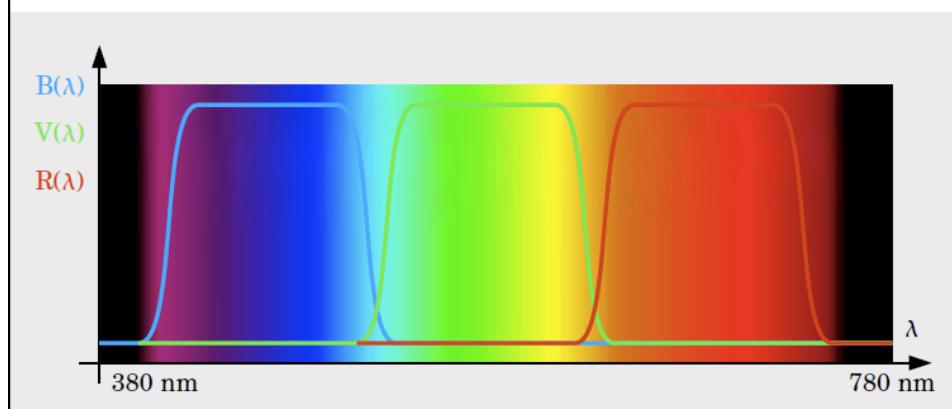


Echantillonnage spectral : image couleur

Image couleur, 3 composantes : Rouge, Verte, Bleue

On échantillonne le spectre visible dans trois bandes.

Les trois bandes (rouge, verte, bleue) ont été sélectionnées afin de correspondre à la vision humaine standard.



Numérisation des composantes : quantification

Valeurs numériques

Chaque valeur d'une composante de la fonction image est représentée par un mot binaire, codé sur un nombre fini de bits.

Pour un mot de m bits, la valeur varie entre 0 et $2^m - 1$.

Exemple : sur 8 bits, composantes entre 0 et 255.

Taille des données image

Dimensions du support : N_x pixels sur N_y lignes.

Nombre de composantes : n , nombre de bits de quantification : m

Exemples

Image monochrome binaire 256x256 : 65.536 bits = 8 Ko

Image spot 2048x2048, 4 canaux, 12 bits : 201.326.592 bits = 24Mo

Effet de la quantification sur le rendu visuel



Image et Transformée de Fourier

Définition de la transformée de Fourier

- Permet de passer d'une représentation spatiale à une représentation fréquentielle
- La transformée de Fourier d'une image permet de la représenter sous la forme de la somme d'exponentielles complexes d'amplitudes, de fréquences et de phases variables.
- Technique qui joue un rôle critique dans les applications de traitement d'image, incluant l'analyse, la restauration, le filtrage et la compression.

Image et Transformée de Fourier

- Fonction de deux variables spatiales m et n :

Si $f(m,n)$ est une fonction discrète spatiale de m et n, alors la transformée de Fourier à deux dimensions de $f(m,n)$ est définie par :

$$F(\omega_1, \omega_2) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} f(m,n) \cdot e^{-j\omega_1 m} \cdot e^{-j\omega_2 n}$$

- ω_1 et ω_2 : variables fréquentielles en rad./échantillon.
- F : appelée représentation fréquentielle de f
- Fonction à valeurs complexes qui est périodique en ω_1 and ω_2 avec la période 2π .
- A cause de la périodicité, on ne s'intéresse qu'à la fenêtre fréquentielle :
 $-\pi < \omega_1 \text{ and } \omega_2 < \pi$

19

Transformée de Fourier inverse

- L'inverse de la transformée de Fourier en 2D est donnée par :

$$f(m,n) = \frac{1}{4\pi^2} \int_{\omega_1=-\pi}^{+\pi} \int_{\omega_2=-\pi}^{+\pi} F(\omega_1, \omega_2) \cdot e^{j\omega_1 m} \cdot e^{j\omega_2 n} d\omega_1 d\omega_2$$

- Somme infinie d'exponentielles complexes (sinusoïdes) à des fréquences différentes.
- L'amplitude et la phase aux fréquences ω_1 et ω_2 sont données par :

$$F(\omega_1, \omega_2)$$

20

Transformée de Fourier discrète (DFT)

■ Les entrées et les sorties de cette transformation sont discrètes :

- Cela permet des traitements numériques sur les échantillons discrets.
- On peut utiliser un algorithme rapide FFT (fast Fourier transform) pour calculer la DFT.

21

Transformée de Fourier discrète (DFT)

Pour une fonction discrète $f(m,n)$ qui est non nulle dans la région finie $0 < m < M-1$ et $0 < n < N-1$, la DFT en deux dimensions $M \times N$ et son inverse sont données par :

$$F(p,q) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n) \cdot e^{-j\left(\frac{2\pi}{M}\right)p m} \cdot e^{-j\left(\frac{2\pi}{N}\right)q n} \quad p = 0, 1, \dots, M-1 \\ q = 0, 1, \dots, N-1$$

$$f(m,n) = \frac{1}{MN} \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} F(p,q) \cdot e^{j\left(\frac{2\pi}{M}\right)p m} \cdot e^{j\left(\frac{2\pi}{N}\right)q n} \quad m = 0, 1, \dots, M-1 \\ n = 0, 1, \dots, N-1$$

22

Transformée de Fourier discrète (DFT)

- Les valeurs $F(p,q)$ sont les coefficient DFT de $f(m,n)$.
- Ce sont les échantillons de la transformée de Fourier $F(\omega_1, \omega_2)$.

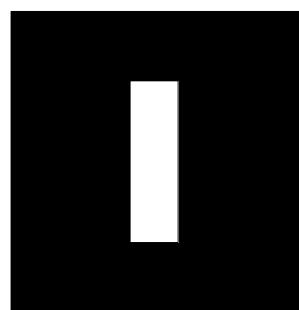
$$F(p,q) = F(\omega_1, \omega_2) \Big|_{\begin{array}{l} \omega_1 = 2\pi p / M \\ \omega_2 = 2\pi q / N \end{array}} \quad \begin{cases} p = 0, 1, \dots, M-1 \\ q = 0, 1, \dots, N-1 \end{cases}$$

- En Python:
fft, fft2, ifftn : implémentent la FFT en 1 dimension, 2 dimensions, n dimensions

23

TFD - image

- Construire une matrice f qui correspond à la fonction $f(m,n)$.
- Exemple : image binaire
 $f = zeros(30,30);$
 $f(5:24,13:17) = 1;$
 $f(m,n) = 1$ dans la région rectangulaire, et 0 ailleurs.



24

Echantillonnage d'une image et effet de repliement de spectre

- Soit une image : `im_fleur = imread('Fleur.jpg')`

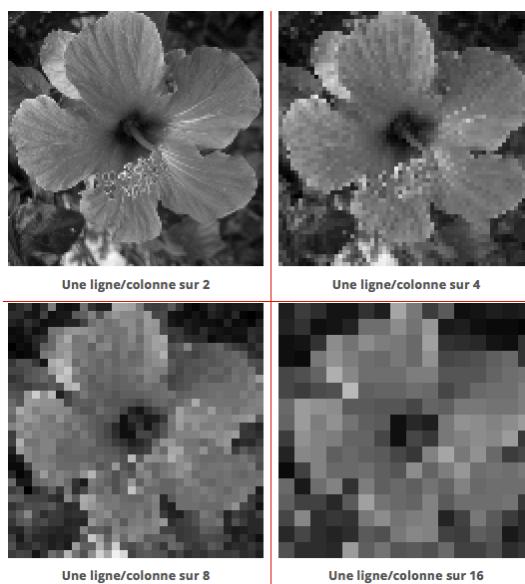
Construire les images de même taille contenant respectivement :

- Une ligne / colonne sur 2
- Une ligne / colonne sur 4
- Une ligne / colonne sur 8
- Une ligne / colonne sur 16

On diminue ainsi la résolution et on fait du sous-échantillonnage (discrétisation de l'image réelle avec plus ou moins de pixels).

25

Echantillonnage d'une image



26

Echantillonnage d'une image et effet de repliement de spectre

- Calculez la DFT de plusieurs images avec des niveaux de sous-échantillonnage croissants

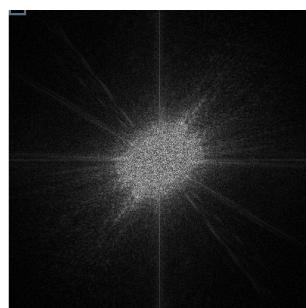
D'après **le théorème de Shannon**, on doit choisir une fréquence d'échantillonnage au moins deux fois supérieure à la fréquence maximale de l'image. Dans le cas contraire, on obtient le phénomène de repliement de spectre (ou *aliasing*).

```
for i=1:2:255  
    for j=1:2:255  
        Im128(i,j) = Im256(i,j);  
        Im128(i,j+1) = Im128(i,j);  
    end  
    Im128(i+1,j) = Im128(i,j);  
end;
```

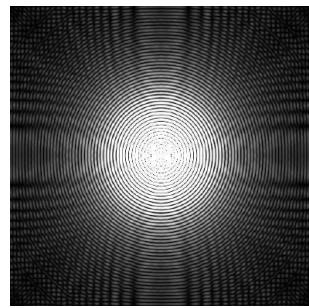
27

Echantillonnage d'une image et effet de repliement de spectre

Observez sur un exemple l'effet du repliement de spectre au niveau du **module** du spectre, pour un sous-échantillonnage de plus en plus grand :



$|F| = \text{fft2}(im)$

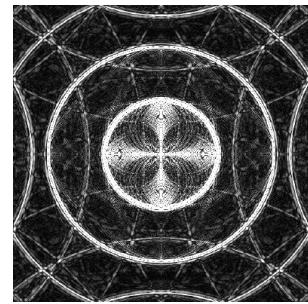
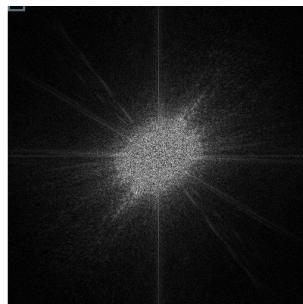


Léger sous-échantillonnage :
présence de hautes fréquences ->
effet de moiré au centre

28

Echantillonnage d'une image et effet de repliement de spectre

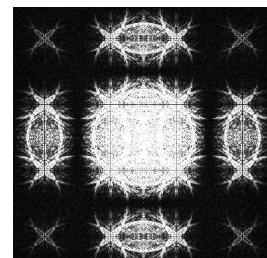
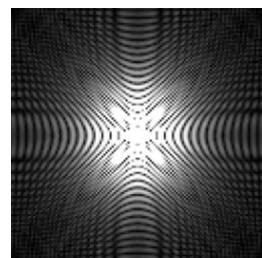
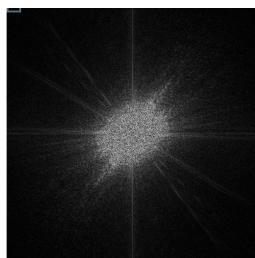
- A cause de la périodisation de l'image, on voit que les hautes fréquences ont tendance à revenir vers le centre : les fréquences réelles viennent se superposer aux fréquences plus basses -> repliement de spectre.



29

Echantillonnage d'une image et effet de repliement de spectre

- Image fortement sous-échantillonnée :
 - disparition des motifs
 - apparition de nouvelles formes



30

Inversion de l'amplitude de la FFT de 2 images

- $F = |F| \cdot \exp(j \cdot \Phi)$
 - Partie réelle :
 $\text{real}(F) = |F| \cos(\Phi)$
 - Partie imaginaire:
 $\text{im}(F) = |F| \sin(\Phi)$
 - $\text{tg}(\Phi) = \text{imag}(F) / \text{real}(F)$
 - $\Phi = \text{atan}(\text{imag}(F) / \text{real}(F))$
- Importance de la phase !
- ```
F1 = fft2(im1);
F2 = fft2(im2);
M1 = abs(F1); amplitude
M2 = abs(F2);
PH1 = atan(imag(F1)/real(F1));
PH2 = atan(imag(F2)/real(F2));
F11 = abs(F2)*exp(j*PH1);
F22=abs(F1)*exp(j*PH2);
im11=ifft2(F11); imshow(im11);
im22=ifft2(F22); imshow(im22);
```

31

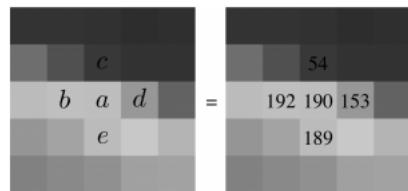
## *Inversion du module de deux images*



32

## Détection des contours d'un objet

- Bords : grandes variations des valeurs des pixels



Exemple d'un voisinage de 5 pixels

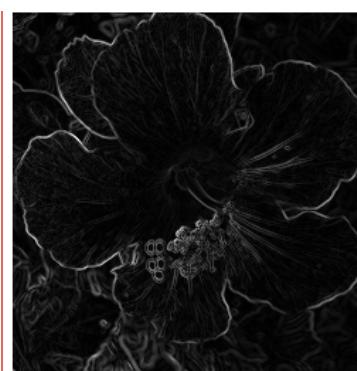
- On peut calculer  $l^2 = (b-d)^2 + (c-e)^2$
- Si  $l^2$  proche de zéro, soit  $b = d$  et  $c = e$ , alors pas de contour
- Si  $l$  grand, contour

33

## Détection des bords d'un objet



Image



Carte de contours  $\ell$

- Si  $l$  est faible, noir
- Si  $l$  est grand, blanc

34

## Filtrage

- Enlever le bruit par des moyennes locales :
- remplacer la valeur  $a$  de chaque pixel par la moyenne de  $a$  et des 8 valeurs  $b,c,d,e,f,g,h,i$  des 8 pixels voisins de  $a$ .

$$\begin{array}{ccc} \begin{matrix} g & c & h \\ b & a & d \\ f & e & i \end{matrix} & = & \begin{matrix} 79 & 54 & 47 \\ 192 & 190 & 153 \\ 166 & 189 & 203 \end{matrix} \end{array}$$

Exemple d'un voisinage 9 pixels



- On remplace  $a$  par  $(b+c+d+e+f+g+h+i)/9$

35

## Filtrage

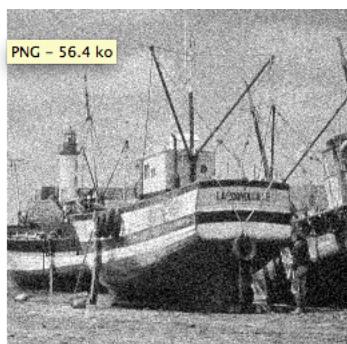


Image bruitée



Moyenne sur 9 pixels

- Introduit du flou dans l'image

36

### Filtrage : convolution

- Appliquer un filtre à une image revient à effectuer un produit de convolution :  $f * h$
- On dit que l'on convolute l'image et le noyau  $h$  (kernel)
- Exemple :



12

$$h = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 1 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

$$1*2+2*(-1)+1*0 + \\ 2*4+1*(-2)+2*(-1) + \\ 1*1+2*2+1*3$$

$$f = \begin{pmatrix} 2 & -1 & 0 \\ 4 & -2 & -1 \\ 1 & 2 & 3 \end{pmatrix}$$

Problème : il manque des informations sur le bord !

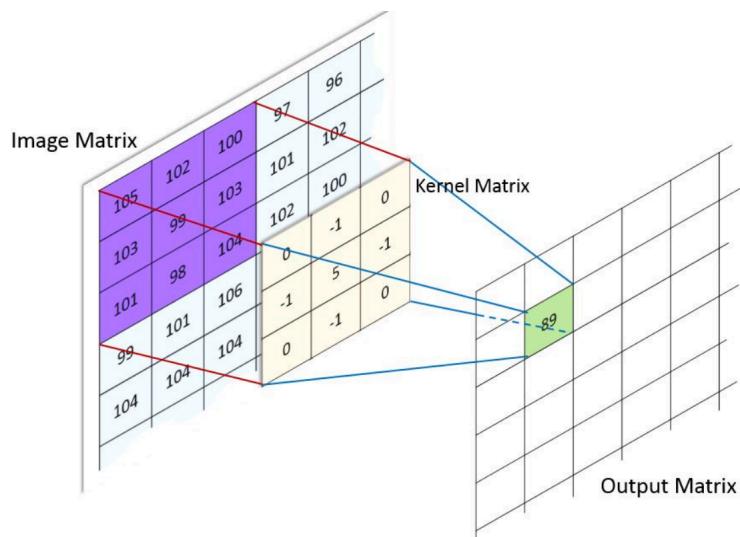
37

### Filtrage : convolution



38

## Filtrage : convolution



39

## Filtrage : convolution

- En fonction des éléments de  $h$ , on réalise des opérations différentes :  $f * h$
- Remarque : la taille de  $h$  peut varier : on prend plus ou moins de points au voisinage de chaque pixel de l'image à transformer.

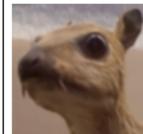
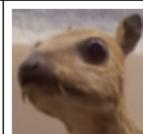
- Détection de contour

| Original | $h$                                                                         |  |
|----------|-----------------------------------------------------------------------------|--|
|          | $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$         |  |
|          | $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$       |  |
|          | $\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$        |  |
|          | $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ |  |

## Filtrage : convolution

- En fonction des éléments de  $h$ , on réalise des opérations différentes :  $f * h$
- Augmente le contraste

- Flouté

| $h$            |                                                                         |                                                                                     |
|----------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <b>Sharpen</b> | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ |  |
| <b>Blur*</b>   | $\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$     |  |
|                | $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$     |  |

41

## Filtrage : convolution

- Plusieurs méthodes sont utilisées pour traiter le bord de l'image :
  - Agrandissement : les pixels de bordure sont étendus conceptuellement aussi loin que nécessaire pour permettre la convolution.
  - Coupure : tout pixel qui demanderait des valeurs en dehors de l'image est enlevé -> peut résulter en une image de sortie plus petite.
  - Les valeurs de bord de l'image sont prises du coin opposé de l'image.
- Normalisation
  - Revient à normaliser chaque élément du noyau par la somme de tous les éléments du noyau. Cela permet d'assurer que les valeurs des pixels de l'image de sortie sont de la même amplitude relative que celles de l'image d'entrée.

42

## **Quand utiliser la transformée de Fourier ?**

- Lorsque le filtre est très petit, il vaut mieux passer par le produit de convolution que par l'espace de Fourier.
- Algorithmes rapides permettant de calculer la transformée de Fourier d'une image : complexité =  $t \log t$  avec  $t$  : taille de l'image  
Si  $t = n^2$ , alors complexité =  $n^2 \cdot \log(n^2) \rightarrow$  algo en  $n^2 \cdot \log(n)$
- Si le filtre est en  $m^2$ , alors le produit de convolution s'effectue en  $m^2 \cdot n^2$   
Donc si on a  $m^2 < \log n$ , alors  $m^2 \cdot n^2 < n^2 \cdot \log n$   
il est donc préférable dans ce cas de ne pas passer par Fourier.

43

## **Filtres gaussiens**

- Ils sont définis par :

$$g_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

- On pourrait définir à partir de là le masque, la convolution, ...
- Mais propriété intéressante de ces filtres. Forme de leur TF:

$$\begin{aligned} \mathcal{F}(g_\sigma)(u, v) &= \frac{1}{2\pi\sigma^2} \iint_{\mathbb{R}^2} \exp\left(\frac{i\pi}{\sigma^2}(x^2 + y^2)(ux + uv)\right) dx dy \\ &= \exp\left(\frac{-\sigma^2(u^2 + v^2)}{2}\right) \end{aligned}$$

44

## *Filtres gaussiens*

- Donc on peut directement appliquer la matrice de convolution :

$$(g_\sigma * f)(x, y) = \mathcal{F}^{-1}(\mathcal{F}(f) \cdot \mathcal{F}(g_\sigma))$$

- Eventuellement, il faudra compléter  $f$ .

45