# Traitement Numérique des Données

**M1 – INF 2163**

**AIDN: Applications Interactives et Données Numériques**

**Sylvie Gibet**



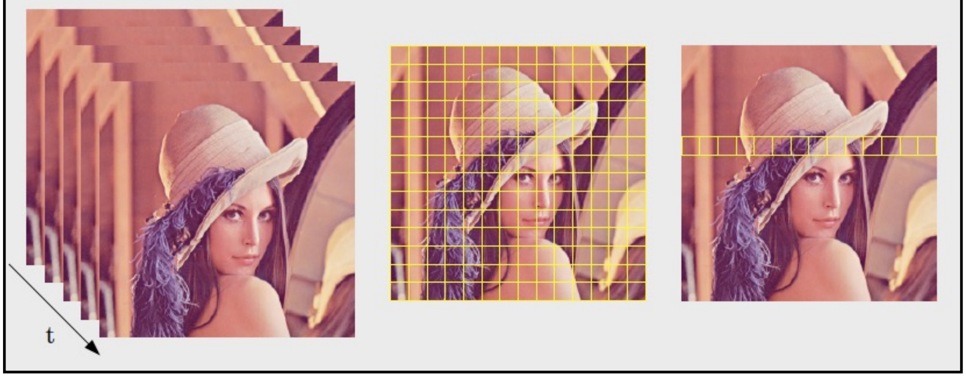


# Traitement des images

## *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**




3

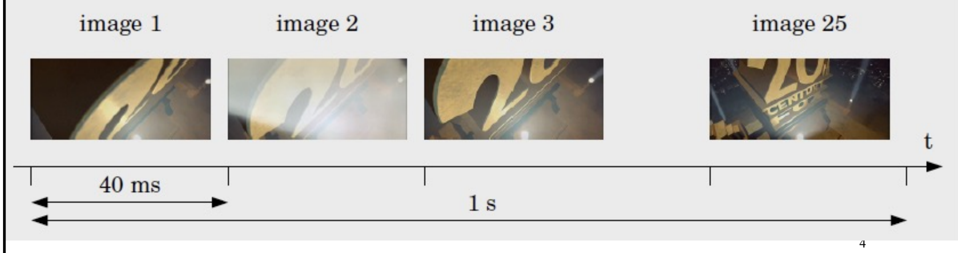
## *Echantillonnage temporel*

**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.**




4

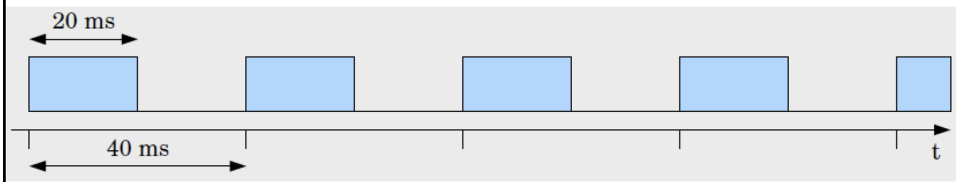
## *Intégration de l'information lumineuse*

### 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 (W = 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**








## *Compromis sensibilité / flou de bougé*

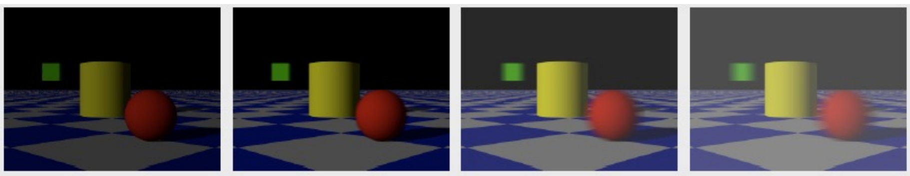
### Pourquoi ?

**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



t = 1 ms | t = 10 ms | t = 20 ms | t = 30 ms

## *Image continue : fonction de 2 variables*

- **Fonction image**

  **fonction I : $R^2$ -> $R^n$, (x,y) -> I(x,y)**

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

  **dans ce cas, l'espace $R^2$ est appelé plan image**

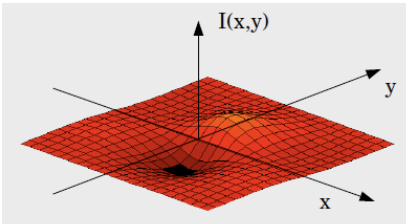
- **Support image**

  **sous-ensemble de $R^2$ de définition de I, de surface finie, en général un rectangle, de côtés Tx et Ty**

- **Exemple à 1 composante**

  **Pour x et y appartenant à [-π, +π]**

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





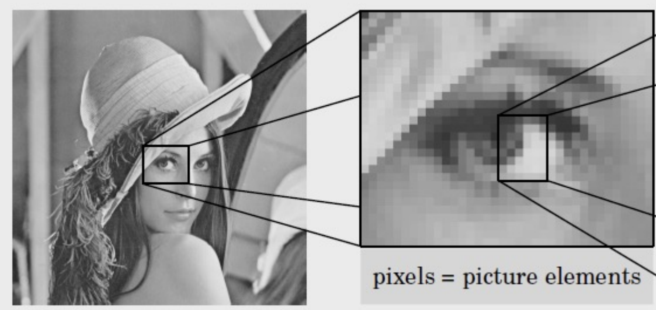

## *Image numérique : tableau bidimensionnel de valeurs*

- **Echantillonnage de la fonction image**

  **Echantillonnage en x et y de la fonction : ne conserve que les valeurs pour les points (x,y) = (c. Δx, l. Δy), avec c et l entiers**

  **Tableau de valeurs I : $Z^2$ -> $R^n$, (c,l) -> I(c. Δx, l. Δy)**

- **Exemple à 1 composante**

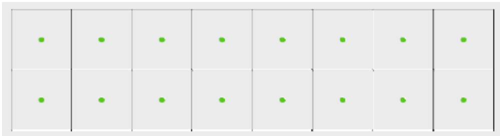



| | | | | | |
|---|---|---|---|---|---|
| 50 | 55 | 60 | 71 | 76 | 68 |
| 62 | 73 | 83 | 101 | 110 | 98 |
| 78 | 88 | 82 | 120 | 126 | 110 |
| 81 | 89 | 94 | 130 | 127 | 124 |
| 92 | 78 | 87 | 141 | 135 | 119 |
| 88 | 86 | 103 | 136 | 133 | 129 |
| 79 | 97 | 109 | 123 | 136 | 142 |
| 100 | 102 | 96 | 97 | 101 | 98 |

## *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.**





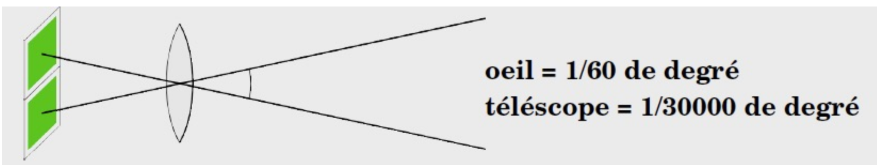
## *Echantillonnage spatial et résolution*

### 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

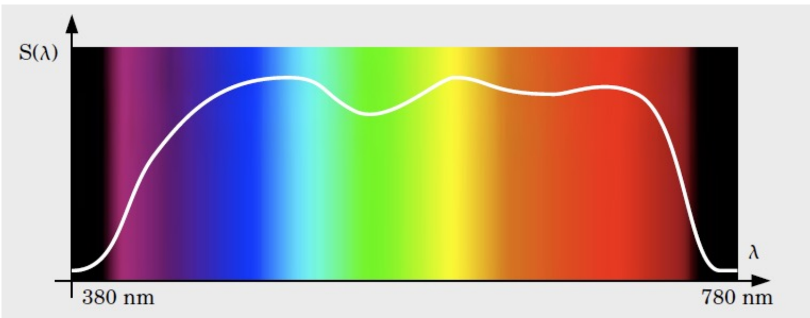
11

## *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.**



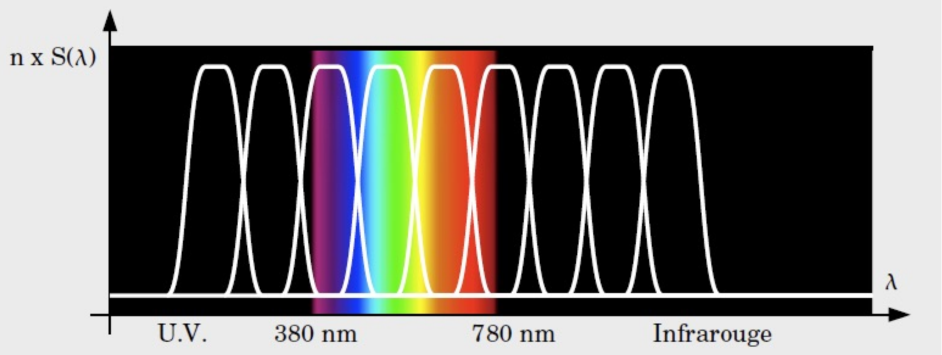

12

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.**

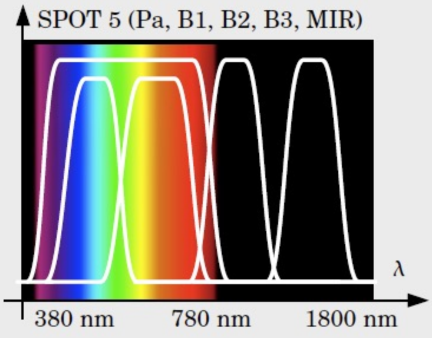



13

## 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)**



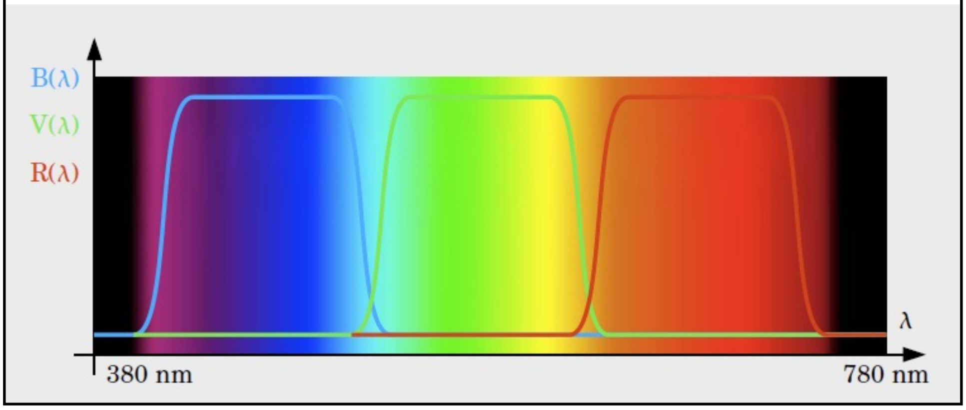

14

## *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.**




15

## *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 : Nx pixels sur Ny 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**



16

## *Effet de la quantification sur le rendu visuel*


17

## *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.

18

18

## *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 :**

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

- ω1 et ω2 : variables fréquentielle en rad./échantillon.
- F : appelée représentation fréquentielle de f
- Fonction à valeurs complexe qui est périodique en ω1 and ω2 avec la période 2π.
- A cause de la périodicité, on ne s'intéresse qu'à la fenêtre fréquentielle :
  -π < ω1 and ω2 < π

19

19

## *Transformée de Fourier inverse*

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

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

- Somme infinie d'exponentielles complexes (sinusoïdes) à des fréquences différentes.
- L'amplitude et la phase aux fréquences ω1 et ω2 sont données par :
  F(ω1, ω2)

20

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.





## *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 MxN et son inverse sont données par :

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

p = 0,1,…,M-1

q = 0,1,…,N-1

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

m = 0,1,…,M-1

n = 0,1,…,N-1

## *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(ω1, ω2).

$$F(p,q) = F(\omega_1,\omega_2)\Big|_{\substack{\omega_1 = 2\pi p/M \\ \omega_2 = 2\pi q/N}} \qquad \begin{cases} p = 0,1,\ldots,M-1 \\ q = 0,1,\ldots,N-1 \end{cases}$$

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

23

23

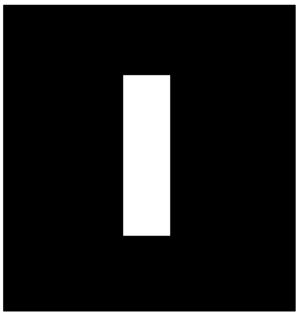
## *TFD - image*

- 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

24

## *Echantillonnage d'une image et effet de repliement de spectre*

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

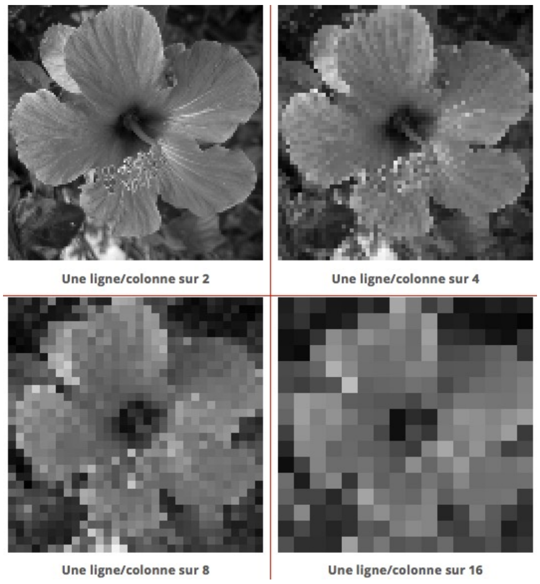
Construire les images de même taille contenant respectivement :
Une ligne / colonne sur 2
Une ligne / colonne sur 4
Une ligne / colonne sur 8
Une ligne / colonne sur 16

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

25

25

## *Echantillonnage d'une image*




26

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*).

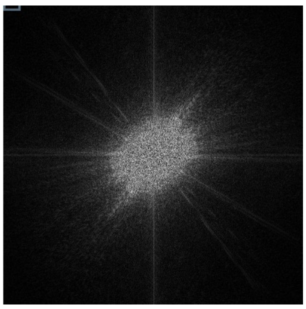
```
for i in range(255):
    for j in range(255):
            Im128(i,j) = Im256(i,j);
            Im128(i,j+1) = Im128(i,j);
    end
    Im128(i+1,j) = Im128(i,j);
end;
```

27

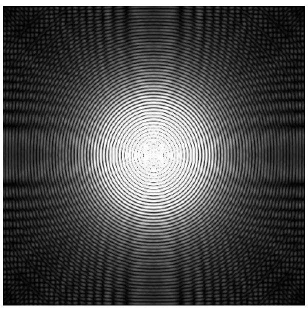
27

## *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| = fft2(im)



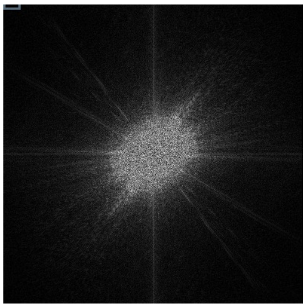
Léger sous-échantillonnage : présence de hautes fréquences -> effet de moiré au centre

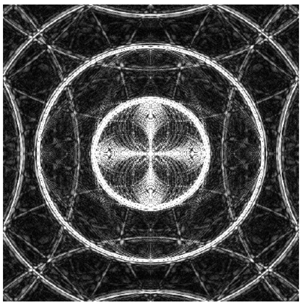
28

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.



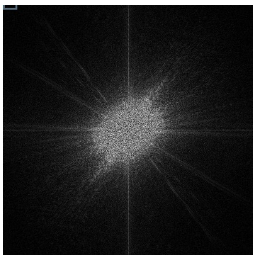


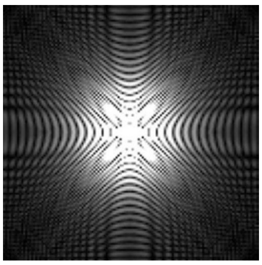
29

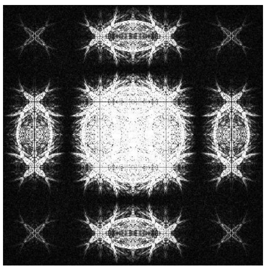
29

## Echantillonnage d'une image et effet de repliement de spectre

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







30

30

## *Inversion de l'amplitude de la FFT de 2 images*

- $F = |F| . \exp(j.\Phi)$
- Partie réelle :

$real(F) = |F| \cos(\Phi)$

- Partie imaginaire:

$im(F) = |F| \sin(\Phi)$

- $tg(\Phi) = imag(F) / real(F)$
- $\Phi = atan(imag(F) / real(F))$

**Importance de la phase !**

```
F1 = fft2(im1);
F2 = fft2(im2);
M1 = abs(F1);      ampliture
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=real(ifft2(F11))
imshow(im11);
im22=real(ifft2(F22))
 imshow(im22);
```

31

31

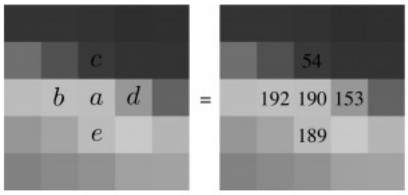
## *Inversion de l'amplitude du spectre de deux images*

32

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

33

## *Détection des bords d’un objet*

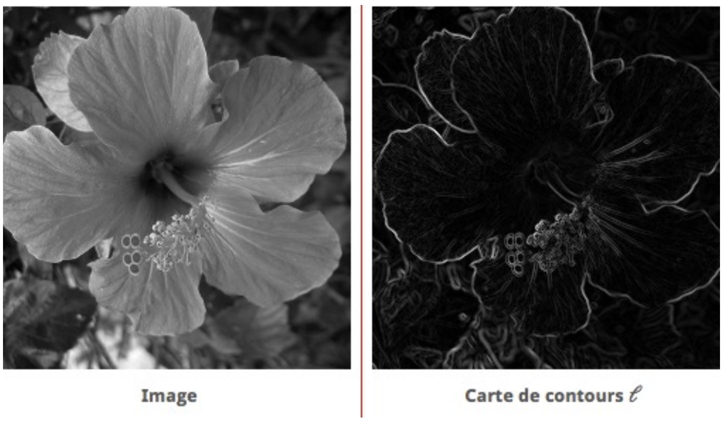


Image | Carte de contours $\ell$

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

34

34