Histogramme des images
&
Opérations sur les image

Plan

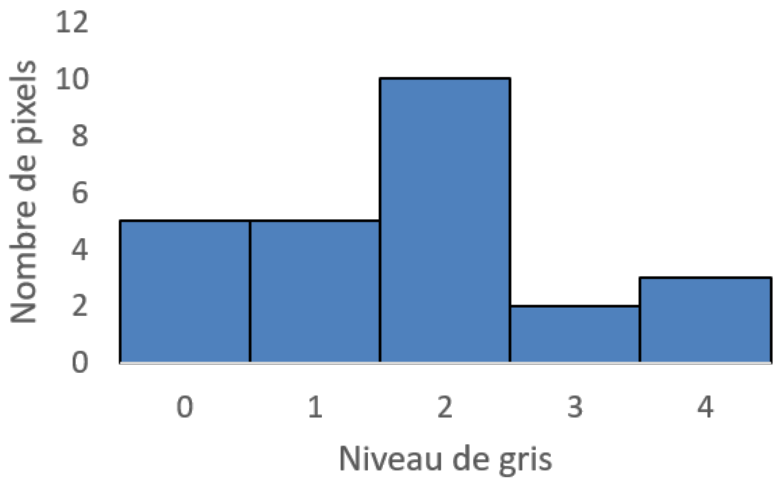
1) L'histogramme d'une image
2) Opérations sur les images
   - Seuillage
   - Addition d'images
   - Soustraction d'images
   - Multiplication d'une image par un coefficient

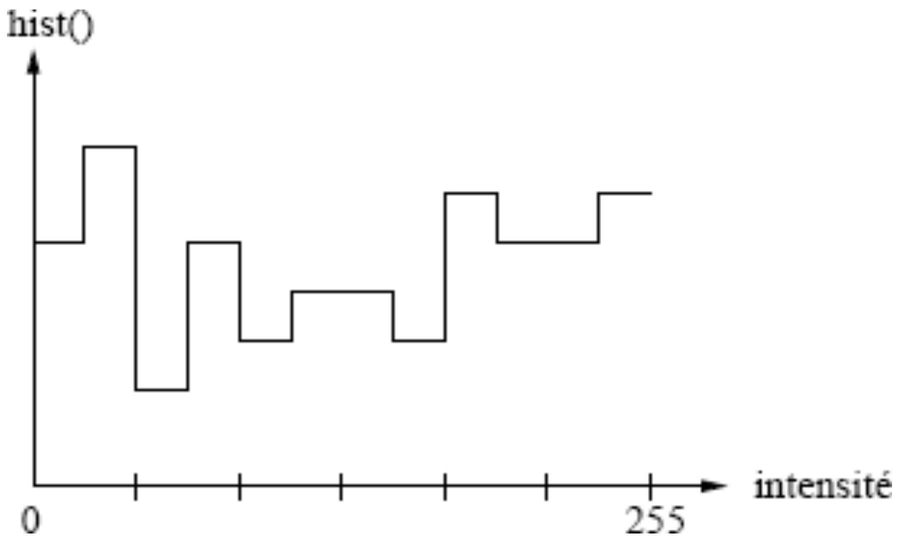
Partie 1: Histogramme d'une image

Histogramme

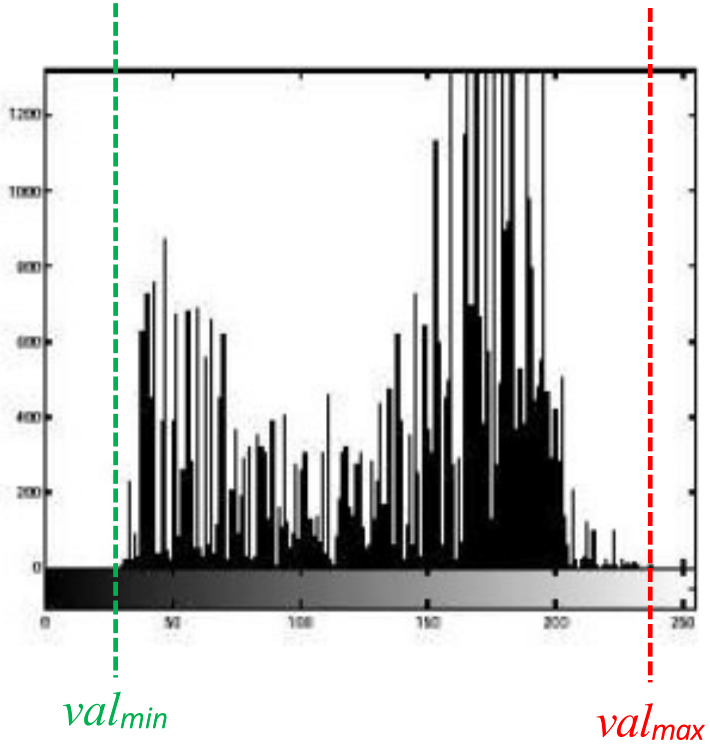
- Définition : à chaque valeur on associe le nombre de pixels dans l'image ayant cette valeur.
- Algorithme:

```
for(i = 0; i < nb_lignes; i++)
        for(j = 0; j < nb_colonnes; j++)
                hist[ Img(i, j) ]++
```





Exemple



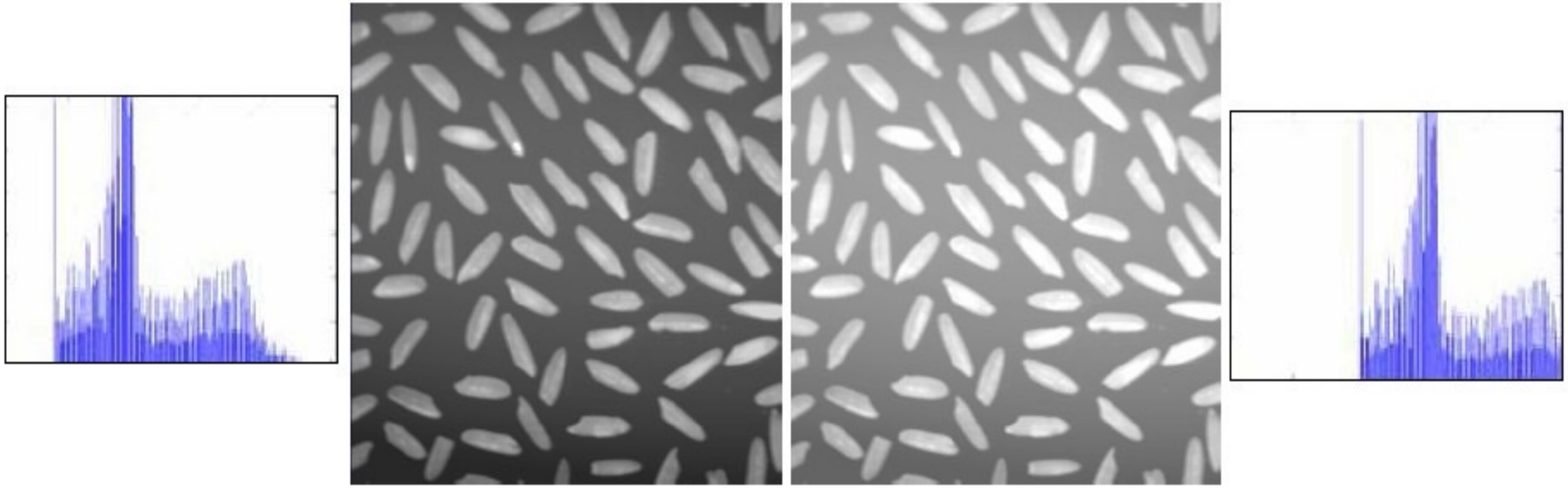
Dynamique d'une image : $D = [val_{min}, val_{max}]$

Luminance ou brillance d'une image

- La luminance (ou brillance) est définie comme la moyenne de tous les pixels de l'image.
- Pour augmenter la luminance, il suffit de décaler l'histogramme :

$$I'(i,j) = I(i,j) + b$$

- Dans les deux images suivantes, seule la luminance est différente



Contraste

- Le contraste peut être défini de plusieurs façons :

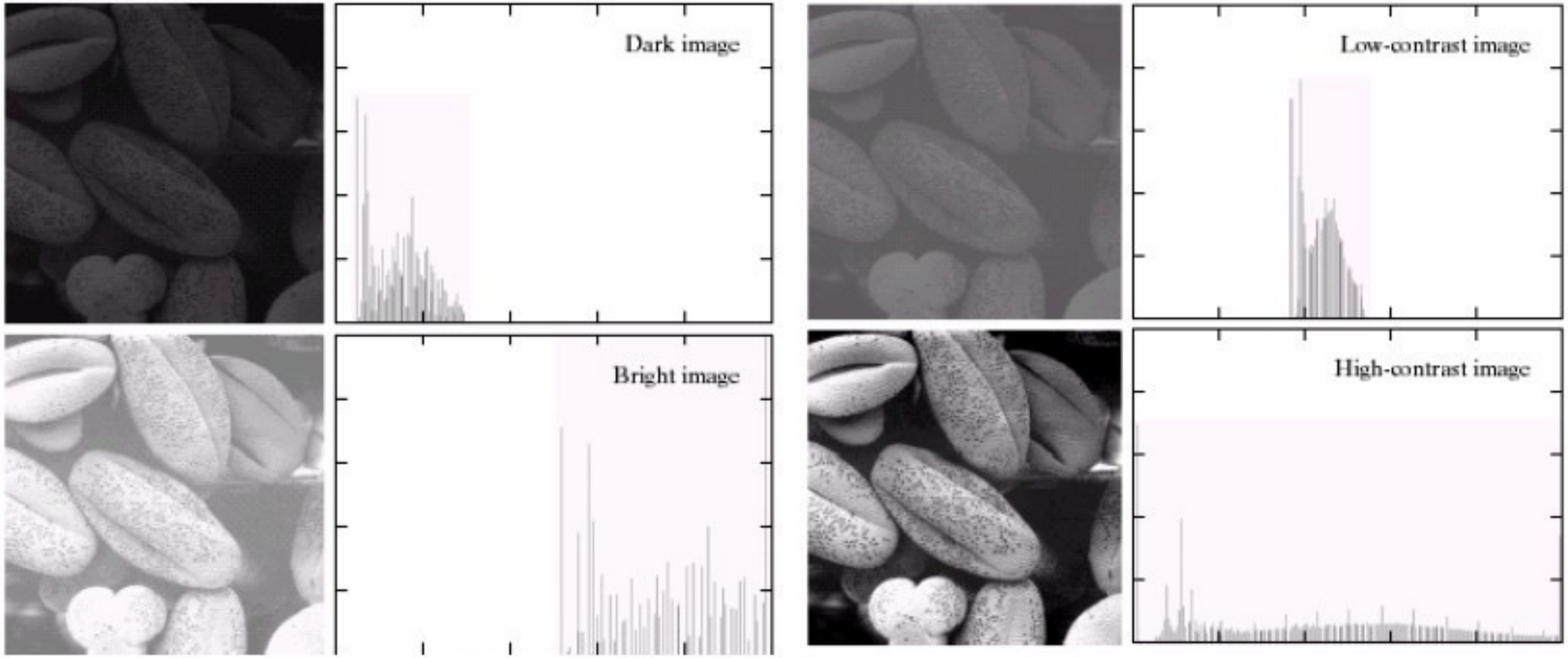
1) Variance des niveaux de gris (N nombre de pixels dans l'image)

$$\frac{1}{N}\sum_{n=1}^{N}(I(i,j) - Moy)^2$$

2) Variation entre niveaux de gris max et min

$$\frac{max[I(i,j)] - min[I(i,j)]}{max[I(i,j)] + min[I(i,j)]}$$

Exemple



a b

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.)

Comment améliorer le contraste ?

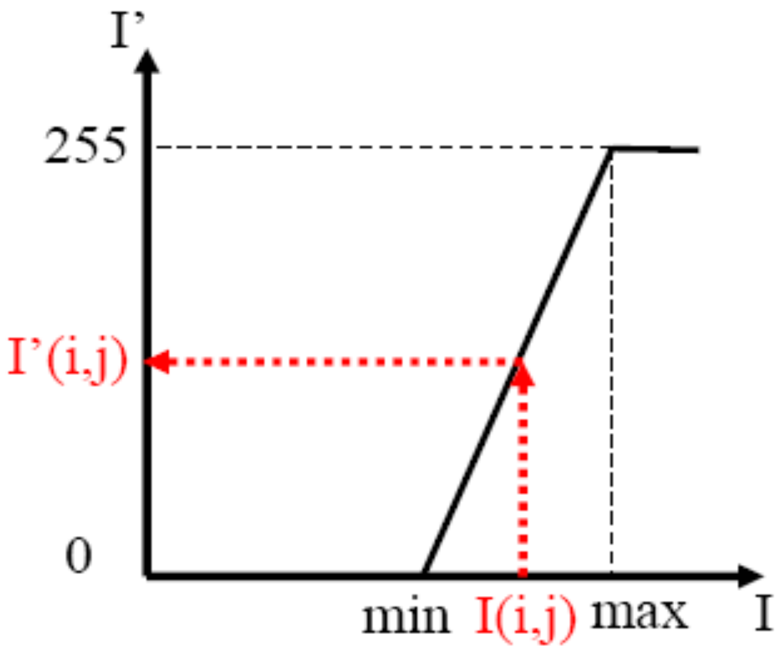
- Deux méthodes:

1) Extension linéaire de dynamique
2) Égalisation de l'histogramme

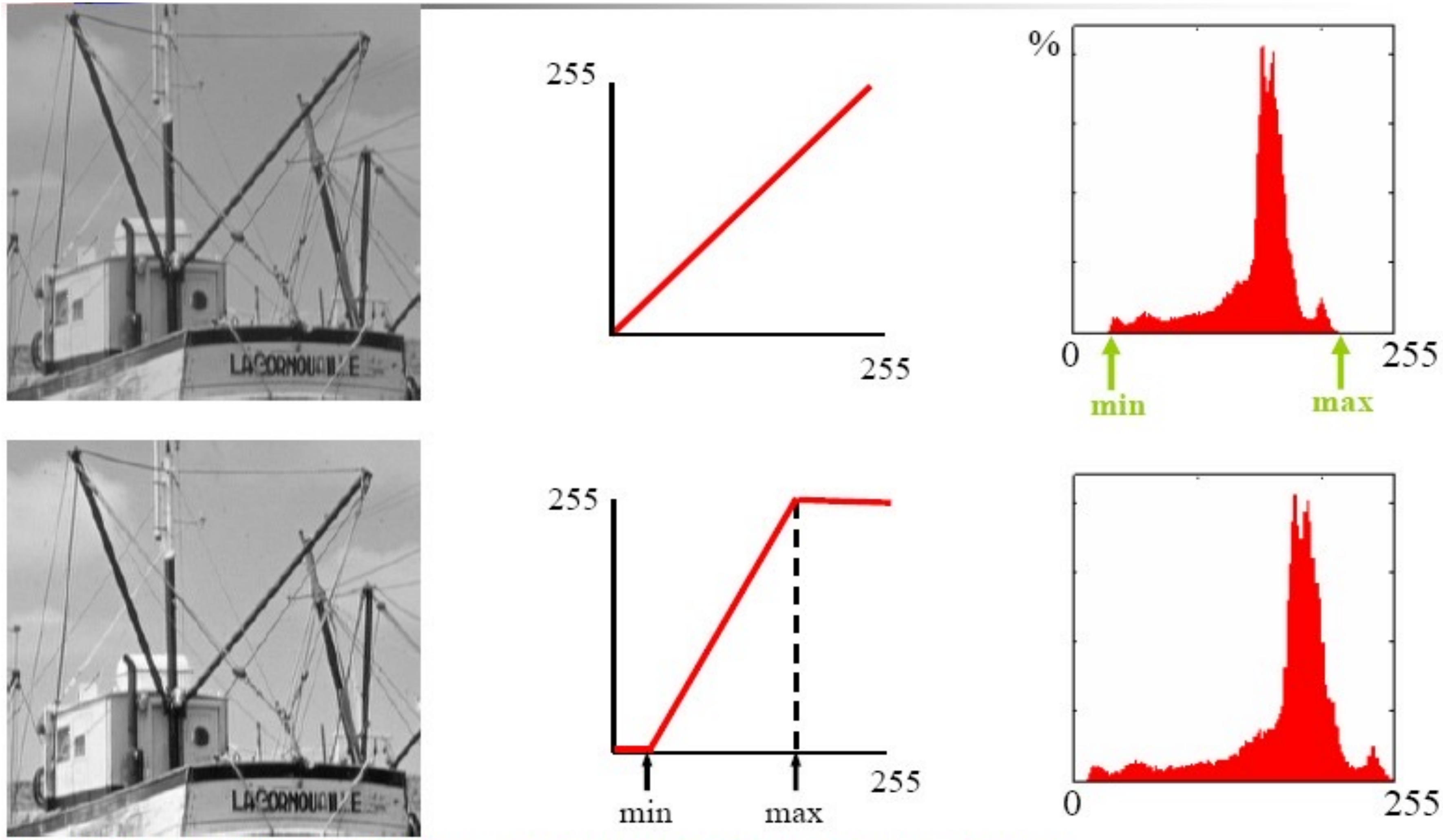
Méthode 1: Extension linéaire de dynamique

- On étire la dynamique en rééchelonnant les niveaux de gris entre 0 et 255

$$I'(i,j) = \frac{255}{max - min}(I(i,j) - min)$$



Exemple



ource : Caroline Rougier. Traitement d'images (IFT2730). Univ. de Montréal.

Extension linéaire de dynamique : algorithme

Première idée (pas optimal)

```
for(i = 0; i < nl; i++)
  for(j = 0; j < nc; j++)
```

$$I'(i,j) = \frac{255 * (I(i,j) - min)}{max - min}$$

Utilisation d'une "Look Up Table"

```
// Initialisation de la LUT
for(ng = 0; ng < 256; ng++)
```

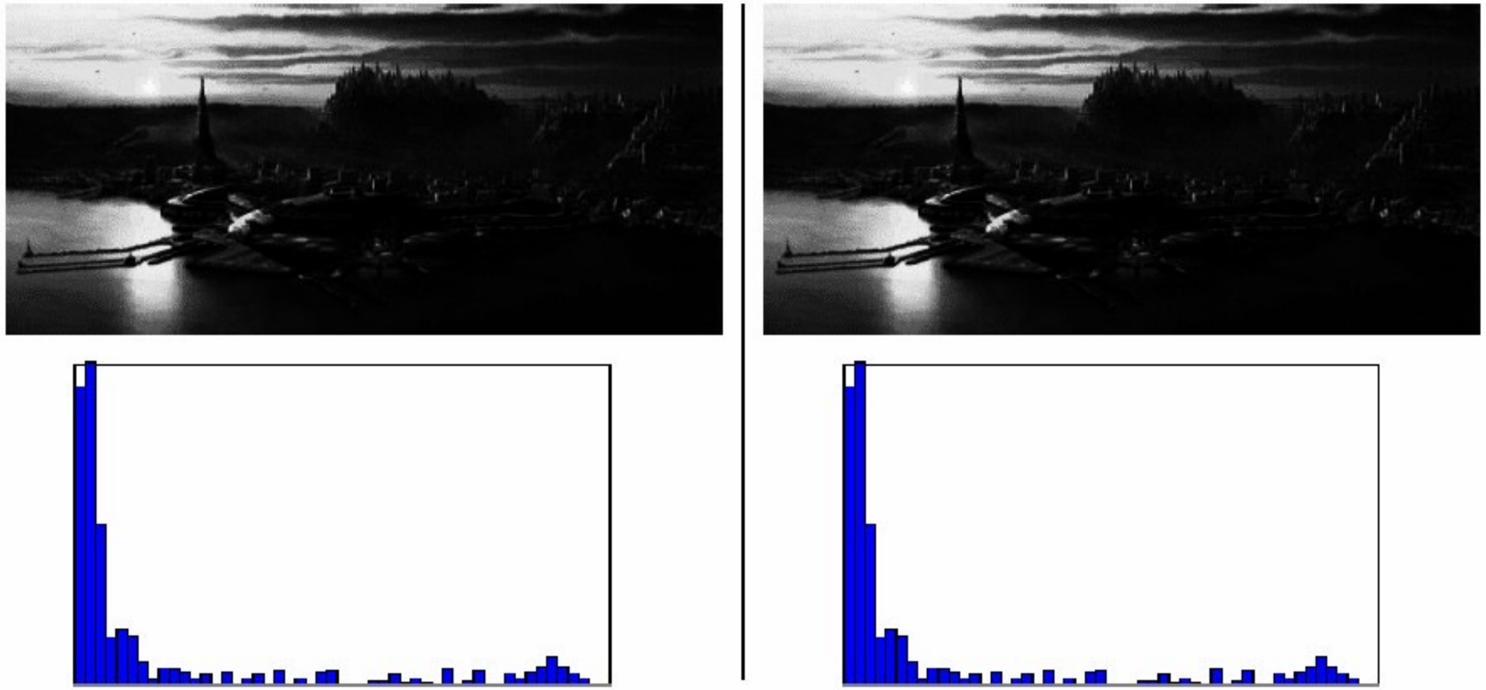
$$LUT[ng] = \frac{255 * (ng - min)}{max - min}$$

```
// Calcul de la transformation
for(i = 0; i < nl; i++)
  for(j = 0; j < nc; j++)
```

$$I'(i,j) = LUT[I(i,j)]$$

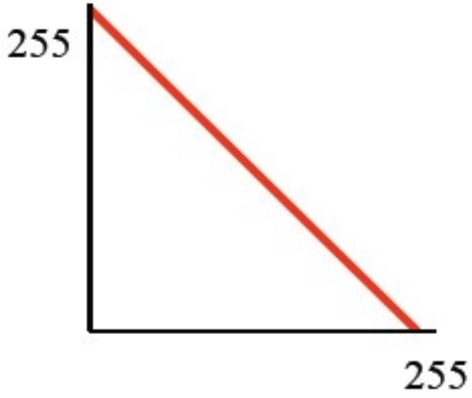
Limite

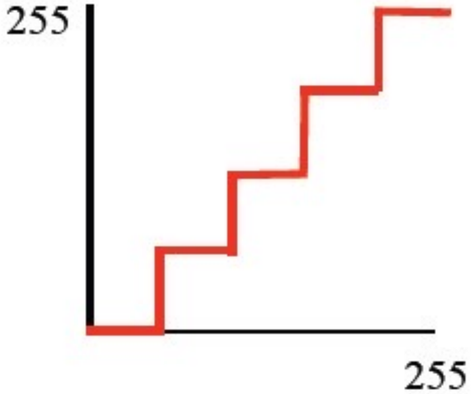
- Si la dynamique est déjà maximale, la transformation n'apporte aucun changement.



Source : www.crans.ens-cachan.fr/~genest/histogramme.pdf

Autres transformations d’histogramme





Méthode 2: Egalisation d'histogramme

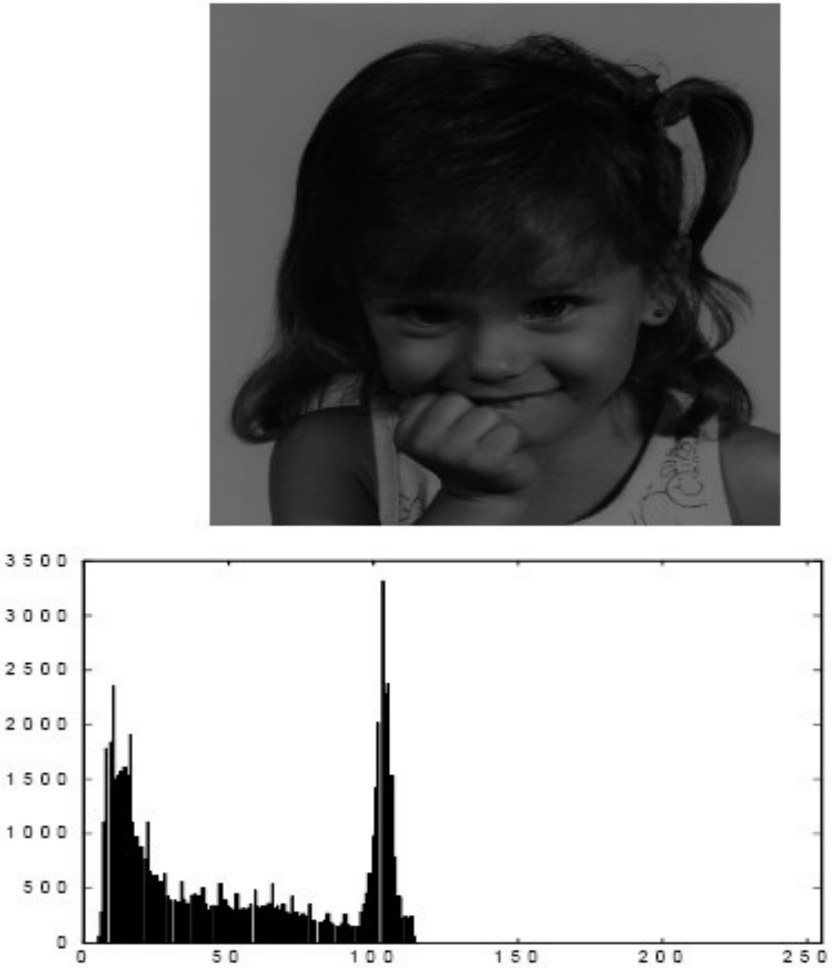


Image originale

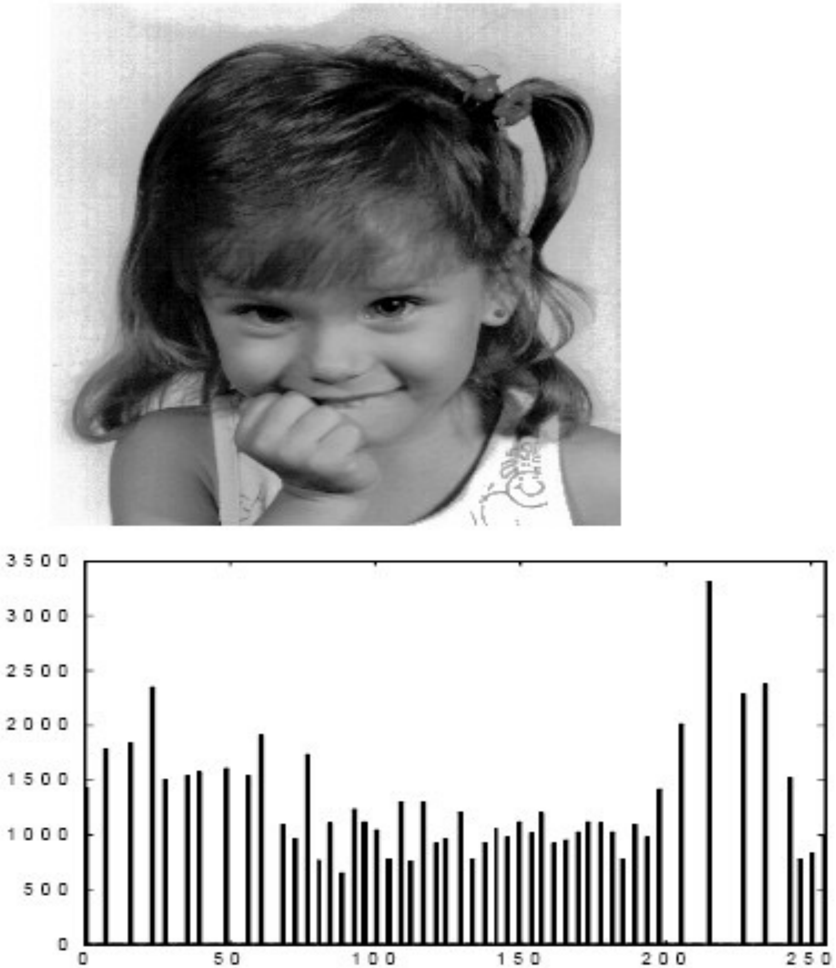


Image plus contrastée

Egalisation d'histogramme : algorithme

- On cherche à aplanir l'histogramme

1. Calcul de l'histogramme $h(k)$ avec $k \in [0, 255]$
2. Histogramme cumulé $C(k) = \sum_{i=1}^{k}(h(i))$
3. Transformation des niveaux de gris de l'image $I'(x, y) = \frac{C(I(x,y))*255}{N}$

Egalisation d’histogramme : exemple 1

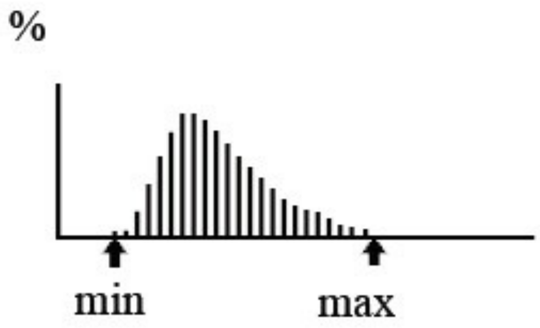
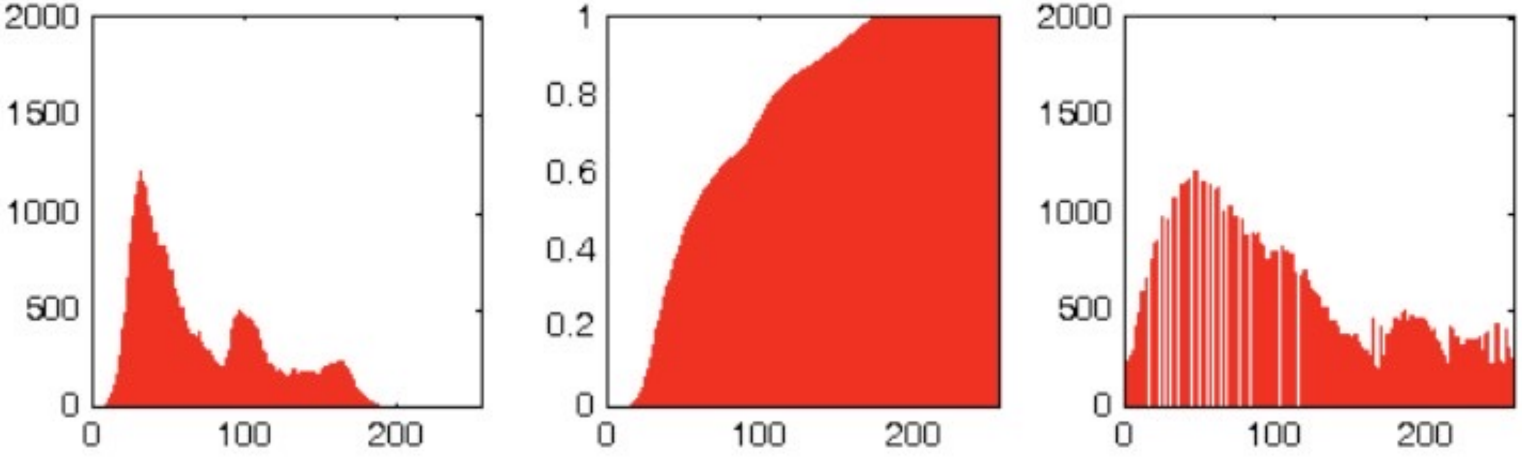
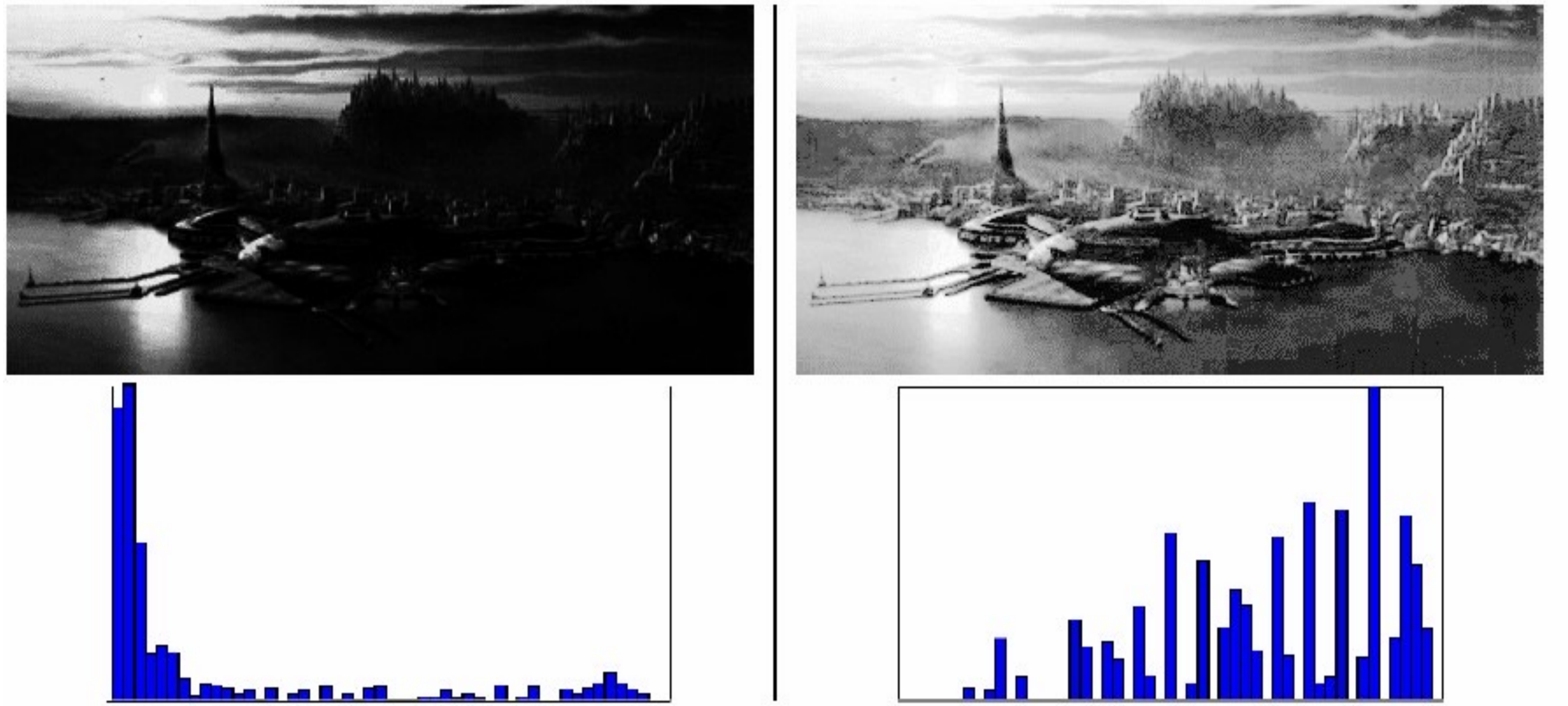


FIGURE 2-4 [rf SCHOWENGERDT, p. 64]



Egalisation d’histogramme : exemple 2



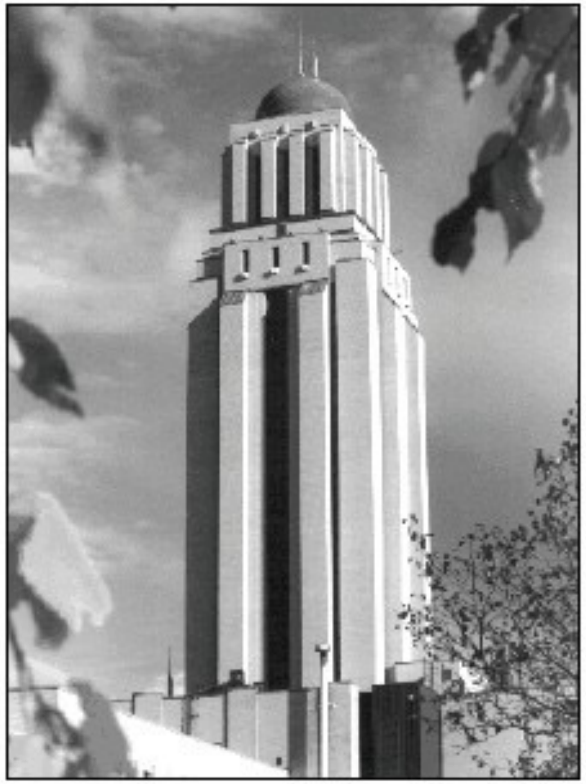
Egalisation de l'histogramme d'une image couleur

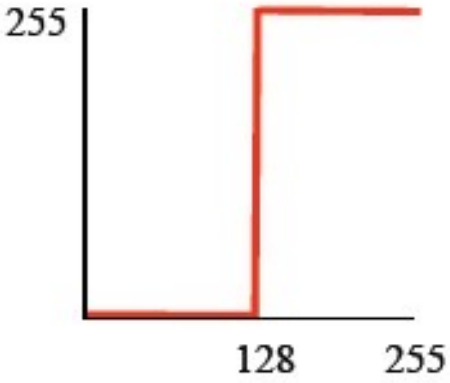
1) Calculer l'intensité de l'image couleur: *Img = (R + V + B)/3*
2) Calculer l'histogramme de *Img*
3) Calculer l'histogramme cumulé de *Img*
4) Appliquer l'égalisation de l'histogramme dans chaque plan de l'image couleur

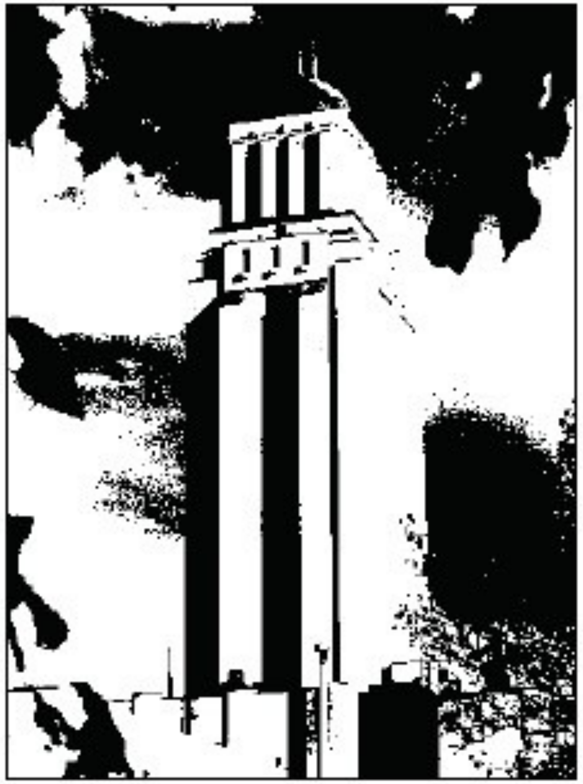
Exemple

Partie 2: Les opérations sur les images

Seuillage





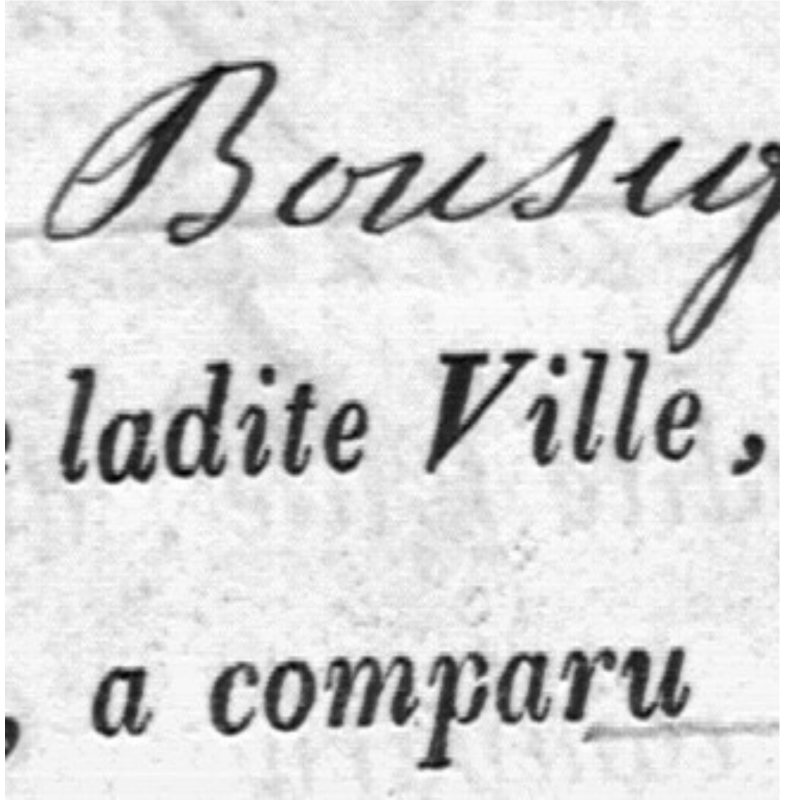


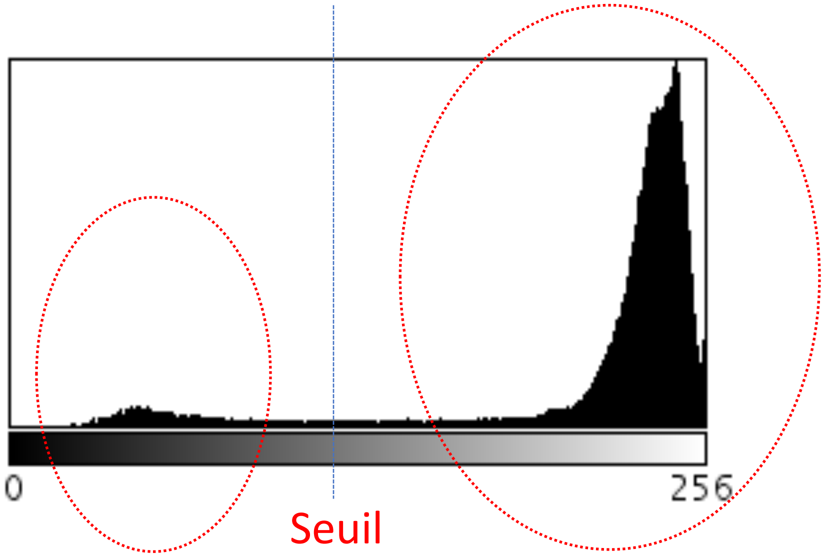
$I'(i,j) = 255$ si $I(i,j) >$ **Seuil**

$I'(i,j) = 0$ sinon

Comment trouver le seuil ?
Un objet sur fond

- L’histogramme d’une image comportant un objet (assez uniforme) sur un fond (assez uniforme) comporte deux pics.



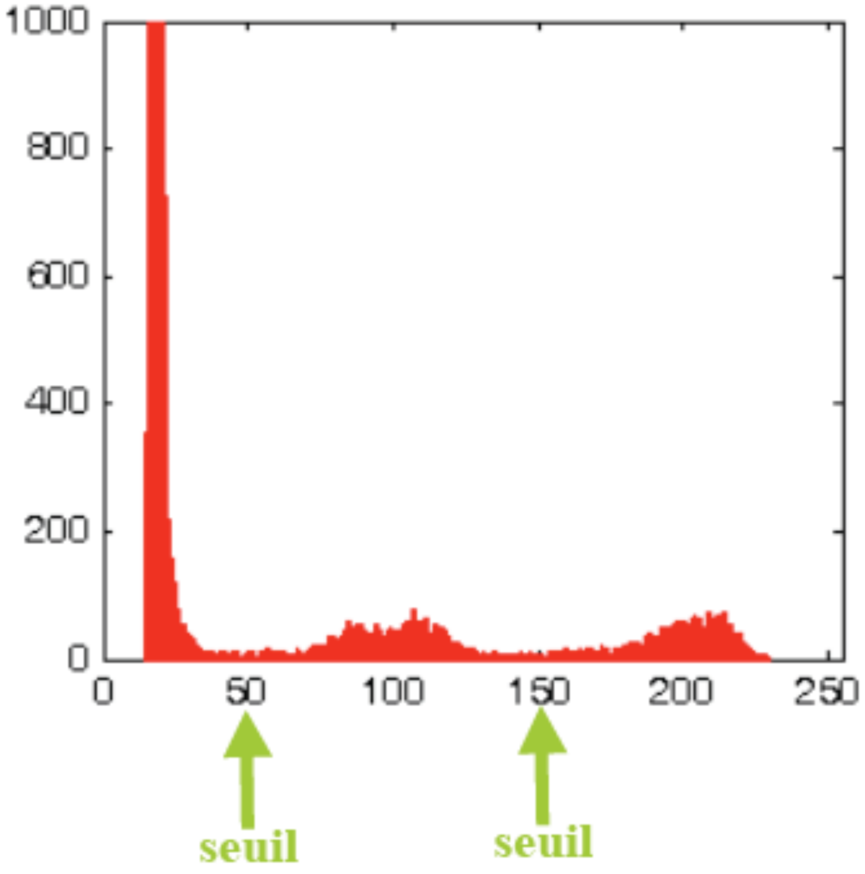


<u>Méthode</u>:
placer le seuil entre les deux pics

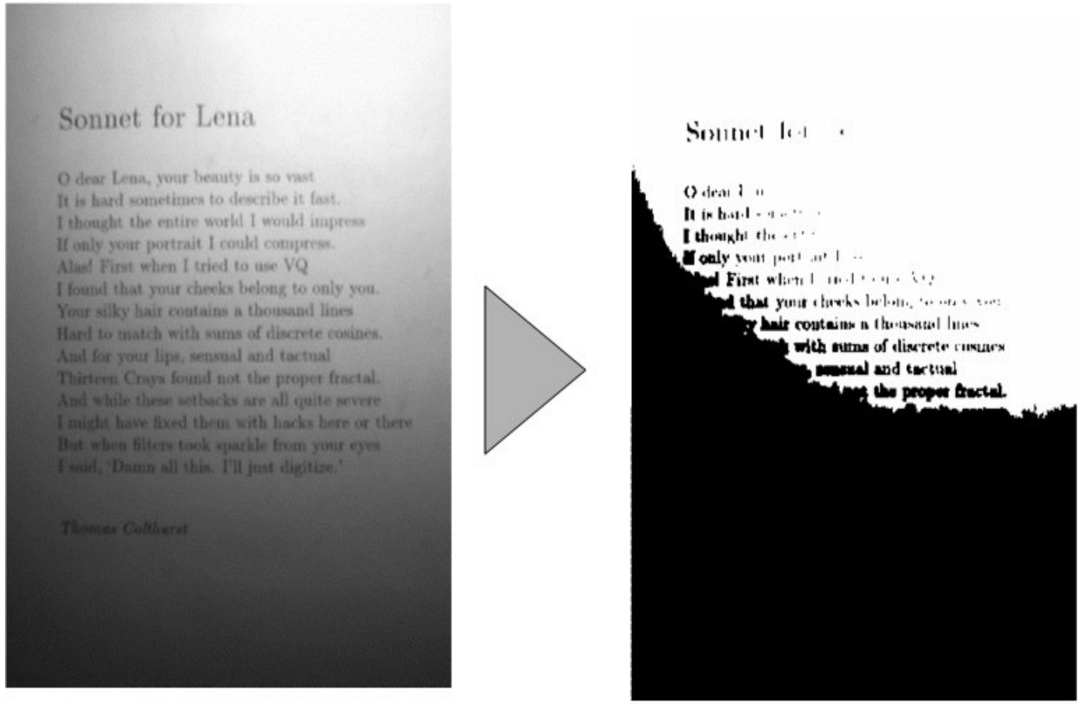
Comment trouver le seuil ?

Plusieurs objets sur un fond

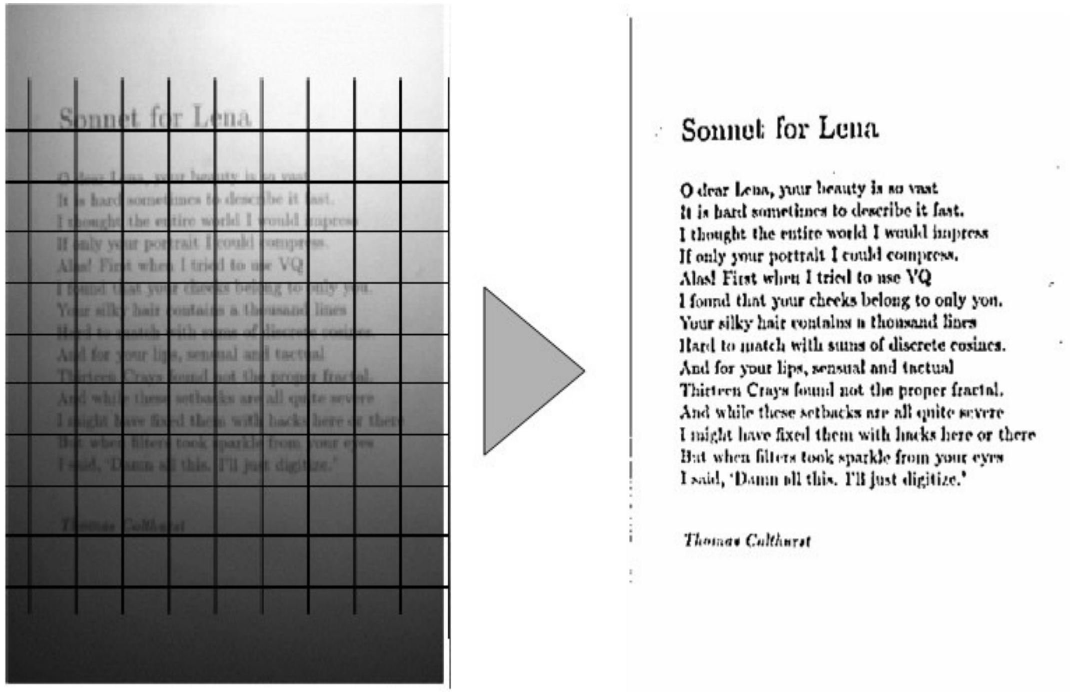
- Choix du seuil par inspection de l'histogramme



Seuillage dynamique



Un seuil global à toute l'image n'est pas toujours adapté



L'image est découpée en bloc puis pour chaque bloc on calcule un seuil global

Addition d'images

- L'addition pixel à pixel de deux images F et G est définie par :

$$A(x,y) = Min(F(x,y) + G(x,y); 255)$$

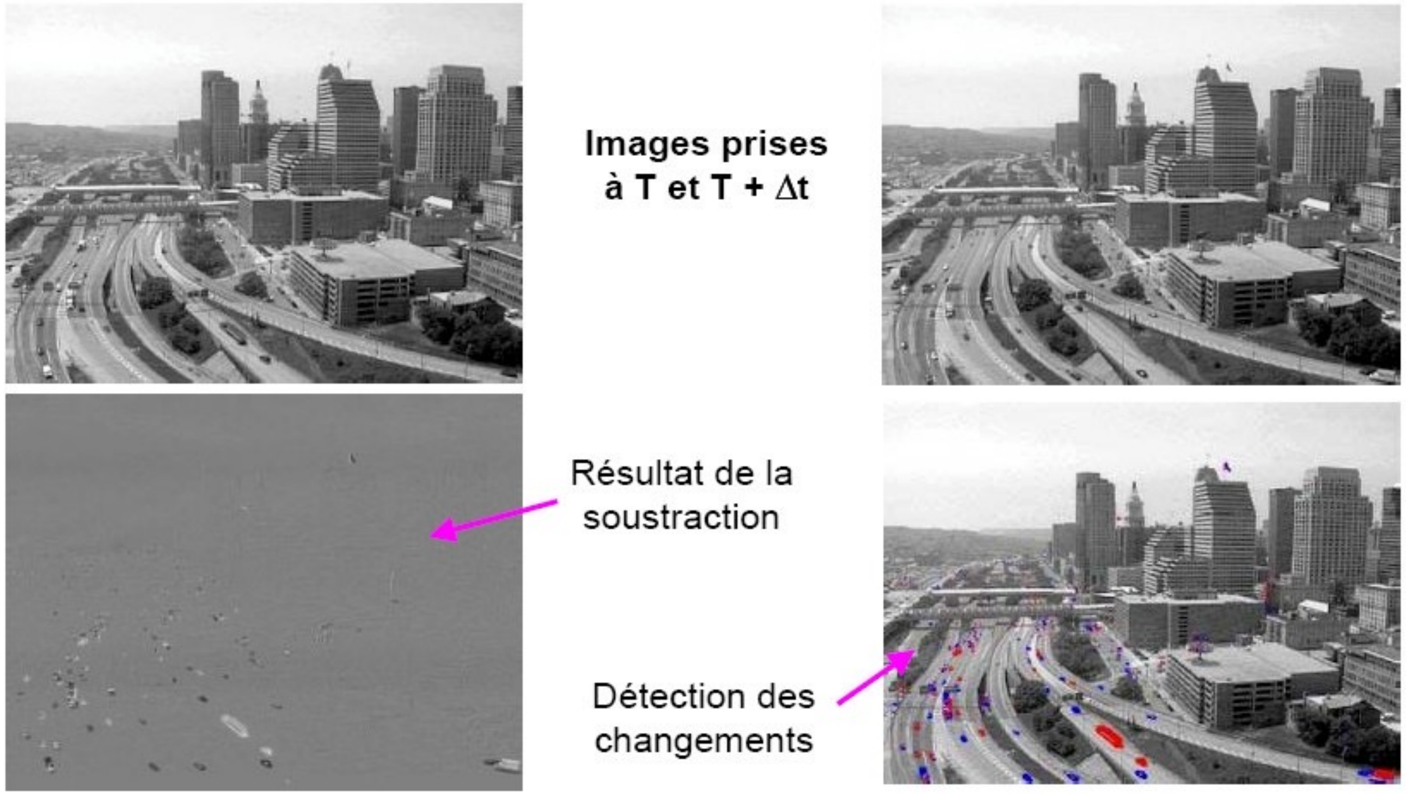
- L'addition d'images peut permettre
  a) de diminuer le bruit d'une vue dans une série d'images
  b) d'augmenter la luminance en additionnant une image avec elle-même

Source : Eric Favier. L'analyse et le traitement des images. ENISE.

Soustraction d’images

- La soustraction pixel à pixel de deux images F et G est définie par :

$$S(x, y) = Max(F(x, y) - G(x, y); 0)$$



La soustraction d’images peut permettre

a) la détection de défauts
b) la détection de mouvements

Multiplication d’une image par un coefficient

- La multiplication d’une image I par un ratio (facteur) est définie par :

$$M(x,y) = Max(F(x,y) * ratio; 255)$$

- La multiplication d’images peut permettre d’améliorer le contraste ou la luminosité

Source : Eric Favier. L'analyse et le traitement des images. ENISE.

Transformation géométrique et Filtrage

Plan

1) Transformations géométriques
2) Filtrage

Partie 1: Transformations géométriques

Transformation géométrique: Flip

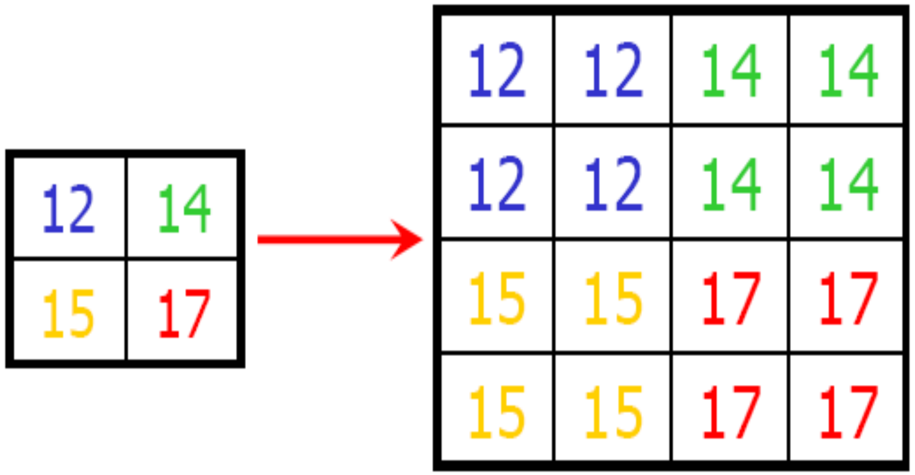
Transformation géométrique: Changement d’échelle

Transformation géométrique: Rotation

Transformation géométrique: Cisaillement

Changement d’échelle

- **Première idée** : agrandissement d'image par copie des pixels
- **Exemple** : multiplication par 2 de la taille de l’image



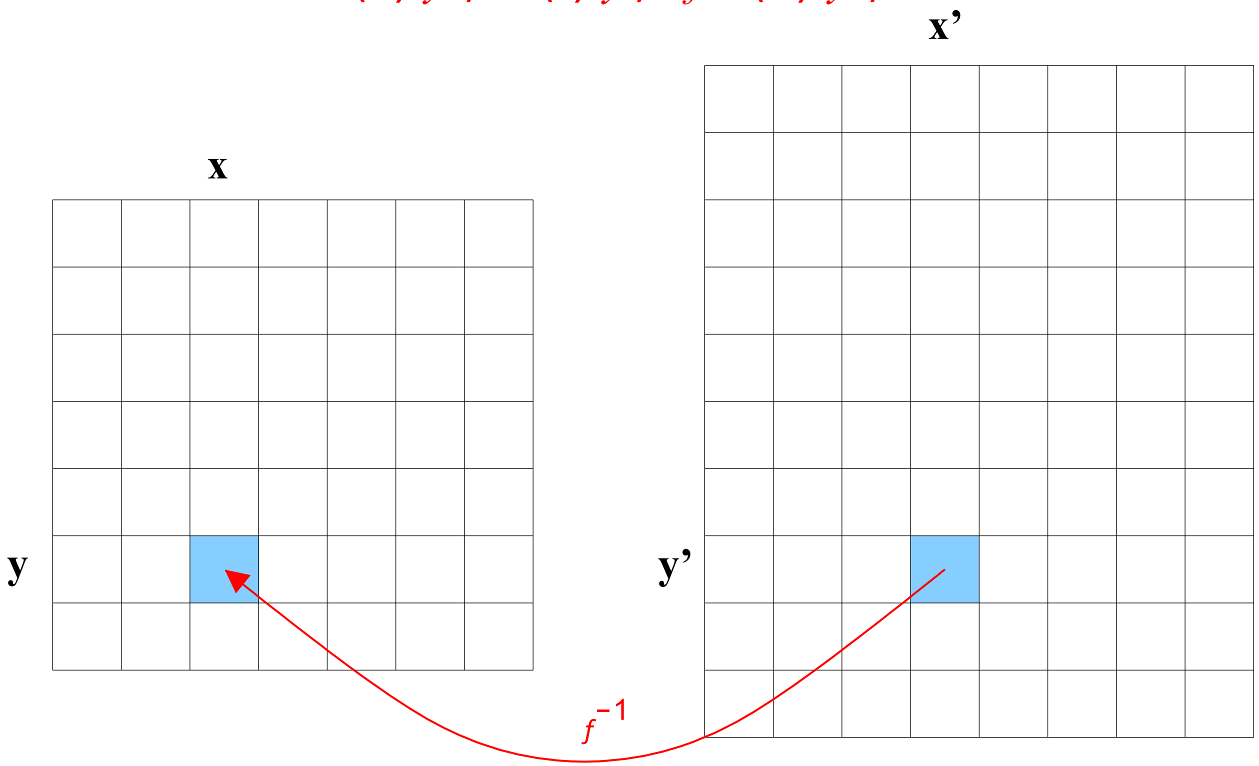
- Limite de cette approche ?

Principe général

Recherche du pixel antécédent: pour chaque pixel de l'image résultat, on cherche le pixel correspondant dans l'image initiale.

Transformation géométrique: $(x, y) \rightarrow (x', y') = f(x, y)$

Transformation inverse: $(x', y') \rightarrow (x, y) = f^{-1}(x', y')$



Changement d’échelle: algorithme

Le changement d'échelle est une homothétie de centre l'origine. On note S_x et S_y les facteurs d’échelle suivant chaque axe (agrandissement ou réduction).

$$x' = S_x.x \qquad x = \frac{1}{S_x}.x'$$

$$y' = S_y.y \qquad y = \frac{1}{S_y}.y'$$

Algorithme

```
W’ = Sx*W
H’ = Sy*H
créer l’image résultat R de taille W’, H’
for(y=0; y<H’; y++)
   for(x=0; x<W’; x++)
      R(x, y) = I(x/Sx, y/Sy)
```

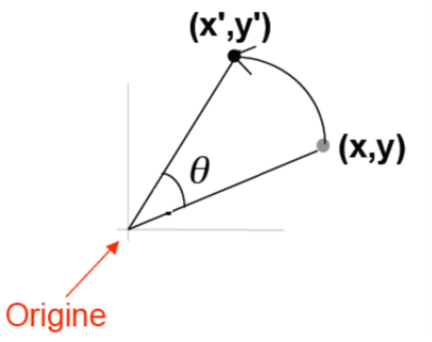
Remarque : arrondi entier des coordonnées de l’antécédent = « interpolation au plus proche voisin »

Changement d'échelle: zoom x 2

Rotation: *rotation au tour de l'origine*

$$x' = x\cos(\theta) - y\sin(\theta)$$
$$y' = y\cos(\theta) + x\sin(\theta)$$

avec θ angle de rotation



Démonstration

Coordonnées polaires : $x = r\cos(\alpha)$ et $y = r\sin(\alpha)$

Après rotation d'angle θ : $x' = r\cos(\alpha+\theta)$ et $y' = r\sin(\alpha+\theta)$

$$x' = r\cos(\alpha)\cos(\theta) - r\sin(\theta)\sin(\alpha)$$
$$y' = r\cos(\alpha)\sin(\theta) + r\cos(\theta)\sin(\alpha)$$

$$x' = x\cos(\theta) - y\sin(\theta)$$
$$y' = y\cos(\theta) + x\sin(\theta)$$

Rotation: *exemple*

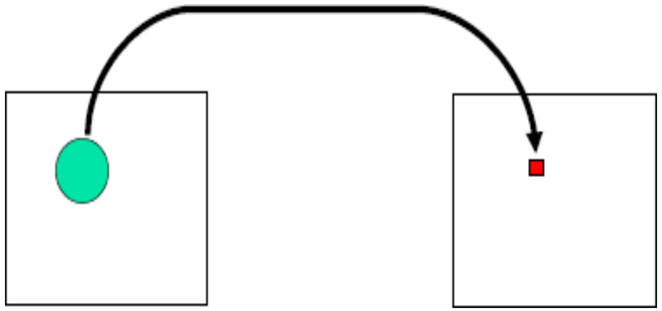
avec interpolation au plus proche voisin

avec interpolation bilinéaire

Partie 2: Convolution

Convolution

Transformation locale : utilisation du voisinage de chaque pixel



Un filtre de convolution (ou masque ou noyau) est généralement une matrice de
$2n+1 \times 2n+1$

<u>Calcul</u> : somme de produits, on parle de filtre linéaire

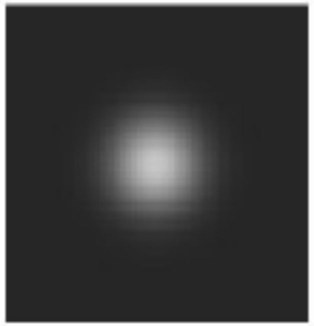
$$R(x,y) = \sum_{u=-n}^{u=n} \sum_{v=-n}^{v=n} I(x+u, y+v).K(u+n, v+n)$$

Pour éviter de modifier la luminance de l'image, la somme des coefficients du filtre doit être égale à 1

Convolution: Exemple

Image d'origine

*

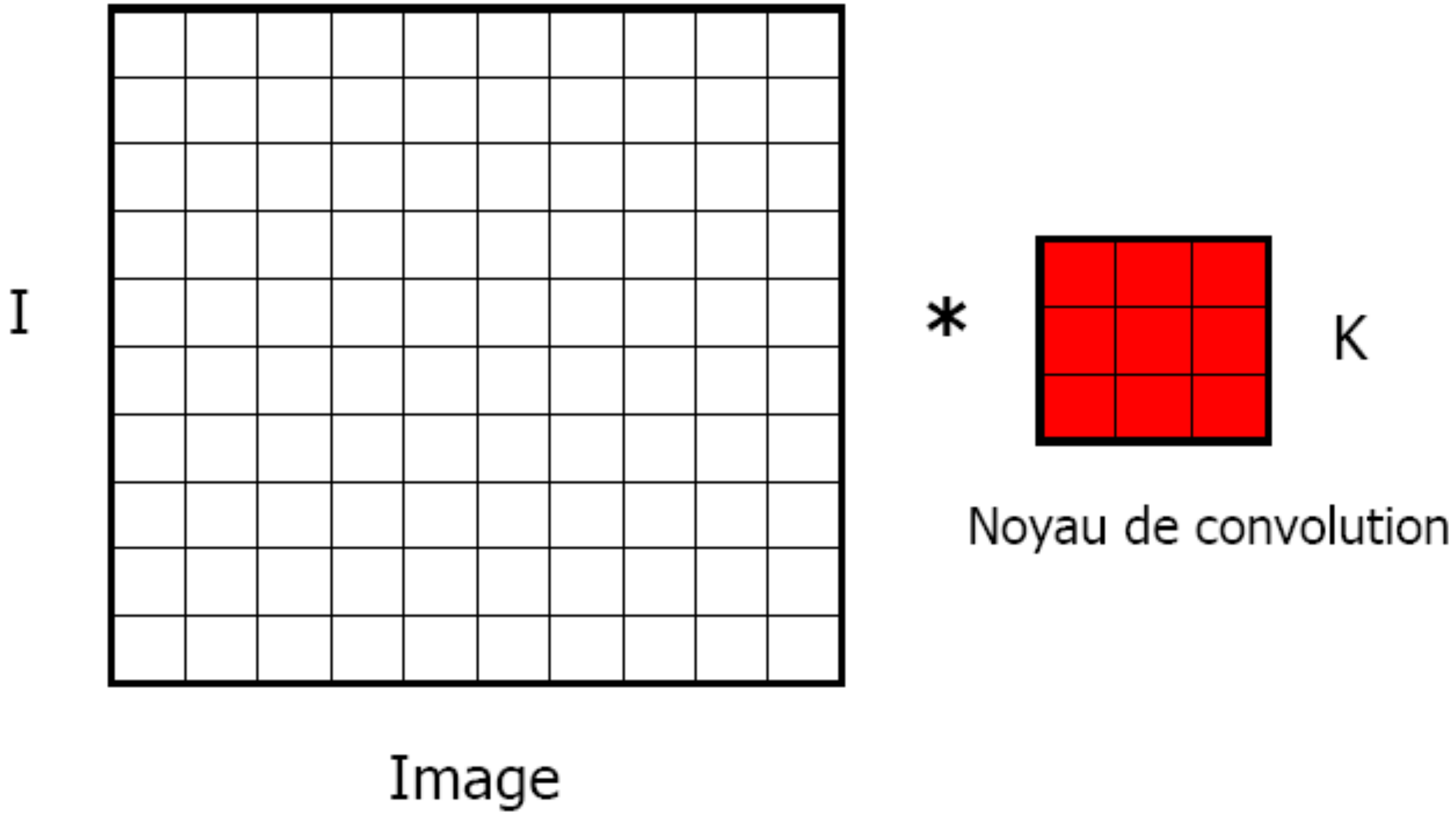


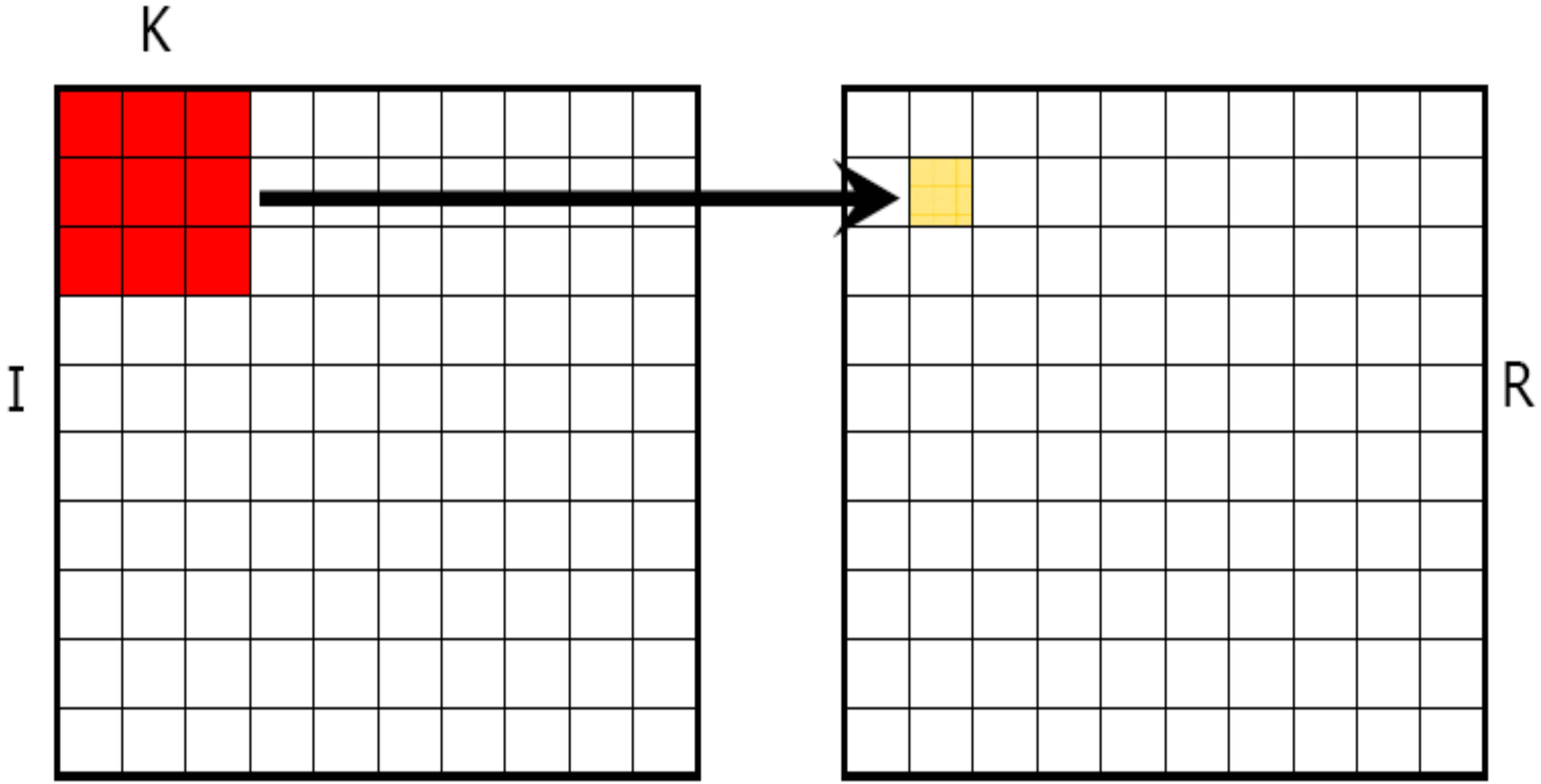
Filtre de convolution (masque)

=

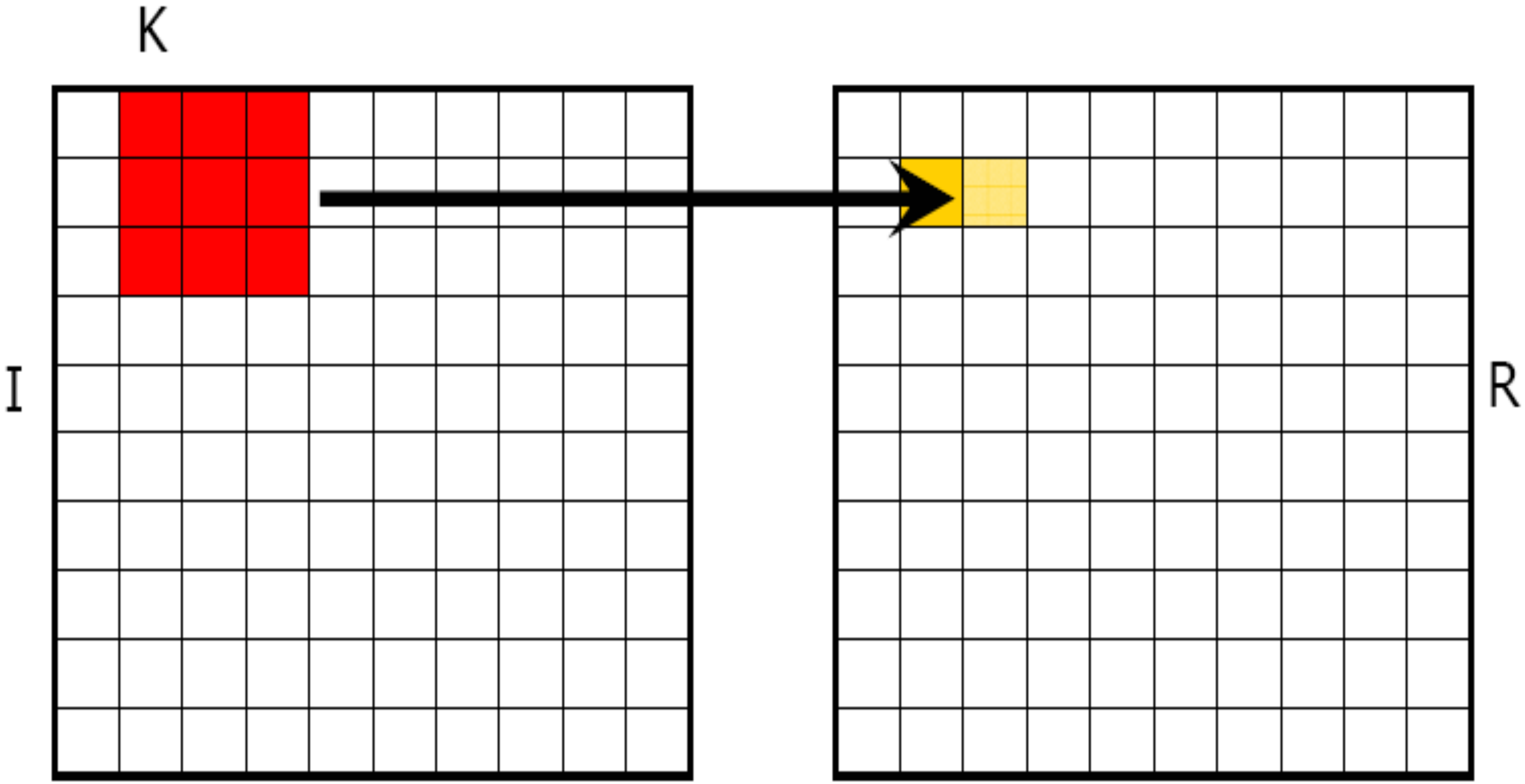
Image convoluée (résultat)

Convolution: principe de calcul

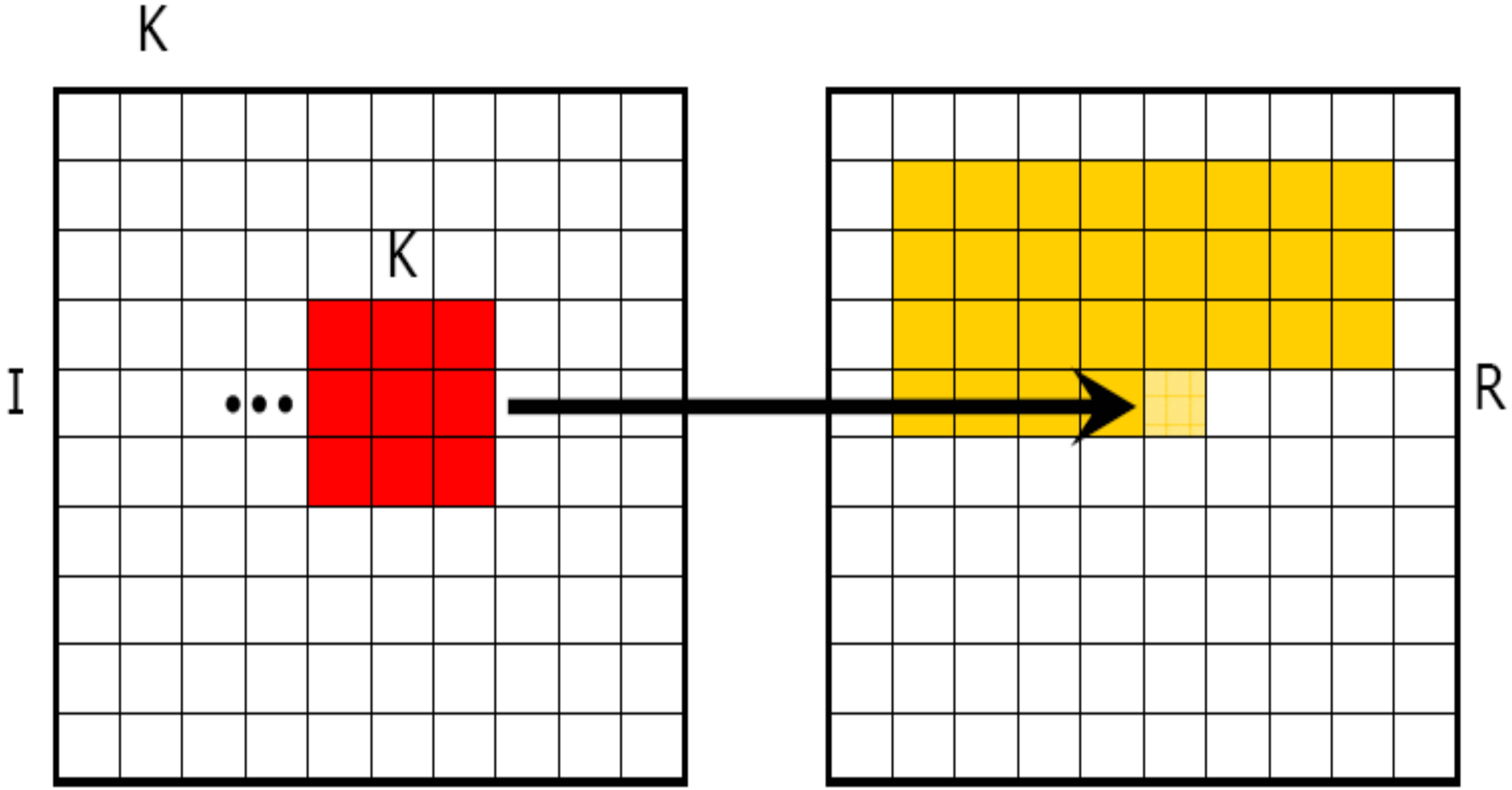




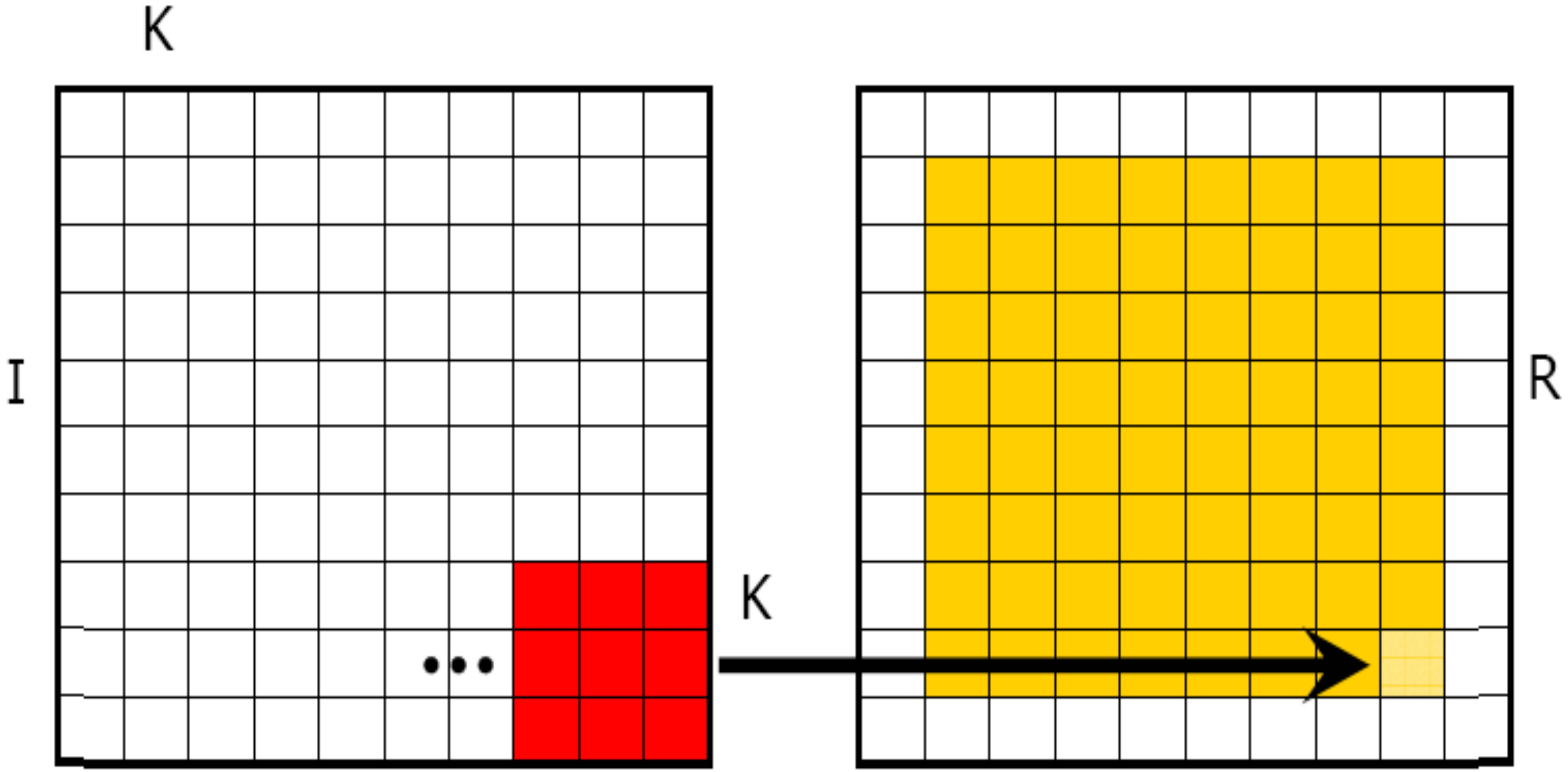
$$R(1,1) = I(0,0)\,K(0,0) + I(1,0)\,K(1,0) + I(2,0)\,K(2,0)$$
$$+ I(0,1)\,K(0,1) + I(1,1)\,K(1,1) + I(2,1)\,K(2,1)$$
$$+ I(0,2)\,K(0,2) + I(1,2)\,K(1,2) + I(2,2)\,K(2,2)$$



$$R(2,1) = I(1,0)\,K(0,0) + I(2,0)\,K(1,0) + I(3,0)\,K(2,0)$$
$$+\ I(1,1)\,K(0,1) + I(2,1)\,K(1,1) + I(3,1)\,K(2,1)$$
$$+\ I(1,2)\,K(0,2) + I(2,2)\,K(1,2) + I(3,2)\,K(2,2)$$



$$
\begin{aligned}
R(x,y) = {} & I(x-1,y-1)\,K(0,0) + I(x,y-1)\,K(1,0) + I(x+1,y-1)\,K(2,0) \\
& + I(x-1,y)\,K(0,1) + I(x,y)\,K(1,1) + I(x+1,y)\,K(2,1) \\
& + I(x-1,y+1)\,K(0,2) + I(x,y+1)\,K(1,2) + I(x+1,y+1)\,K(2,2)
\end{aligned}
$$

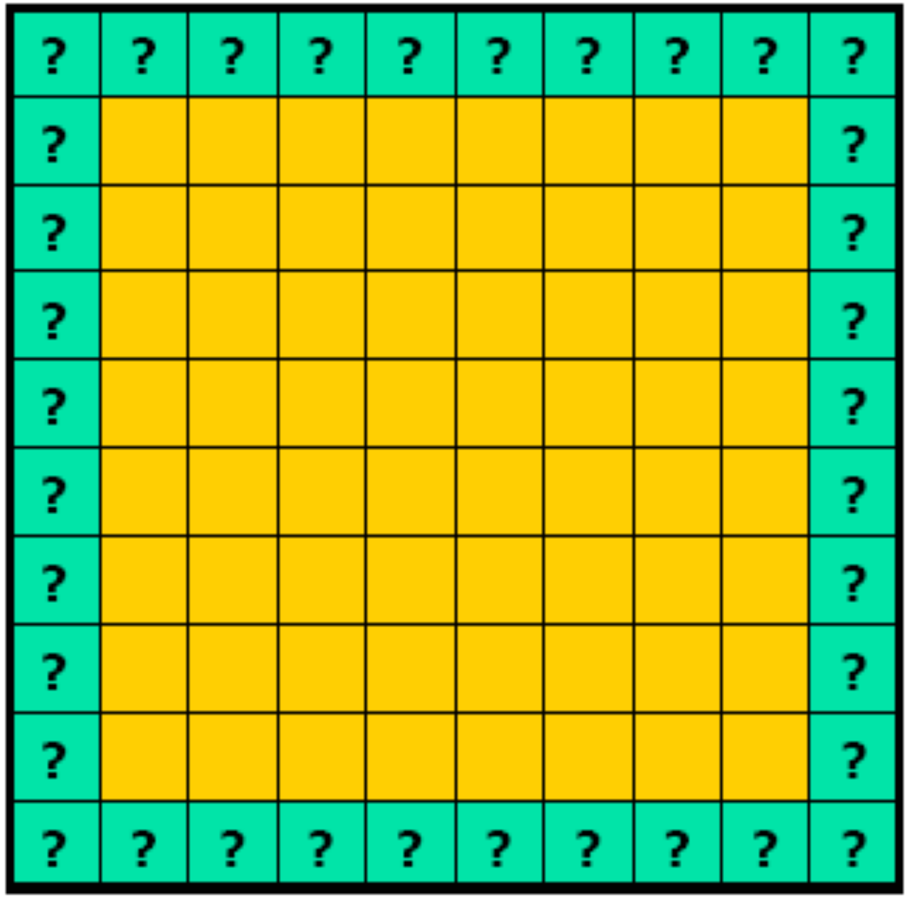


$$
\begin{aligned}
R(N-2,M-2) = {} & I(N-3,M-3)\,K(0,0) + I(N-2,M-3)\,K(0,1) + I(N-1,M-3)\,K(0,3) \\
& + I(N-3,M-2)\,K(1,0) + I(N-2,M-2)\,K(1,1) + I(N-1,M-2)\,K(1,2) \\
& + I(N-3,M-1)\,K(2,0) + I(N-2,M-1)\,K(2,1) + I(N-1,M-1)\,K(2,2)
\end{aligned}
$$

Calcul sur les bords de l’image

Plusieurs possibilités :

- Mettre à zéro
- Convolution partielle utilisant une portion du filtre
- Compléter les valeurs manquante en construisant le miroir de l’image



Familles de filtres

1) **Filtre passe-bas** : atténue le bruit et les détails
2) **Filtre passe-haut** : accentue les détails et les contours

Le filtre Moyenneur

- C'est un filtre passe-bas
  - Lisse l'image (effet de flou)
  - Réduit le bruit
  - Réduit les détails
- Filtre dont tous les coefficients sont égaux (chaque pixel est remplacé par la moyenne de ses voisins)

$$\begin{array}{|c|c|c|}\hline 1/9 & 1/9 & 1/9 \\ \hline 1/9 & 1/9 & 1/9 \\ \hline 1/9 & 1/9 & 1/9 \\ \hline \end{array} \quad \text{ou} \quad 1/9 \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

3x3

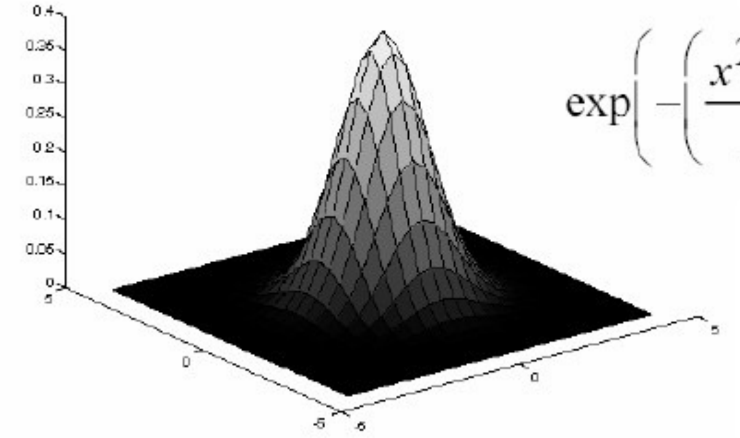
Filtre Moyenneur: *Exemple*

- Plus le filtre grossit, plus le lissage devient important.

Original | Moyenne 5x5 | Moyenne 11x11

Le filtre Gaussien

- Le filtre gaussien donne un meilleur lissage et une meilleure réduction du bruit que le filtre moyenne.



$$\exp\left(-\left(\frac{x^2+y^2}{2\sigma^2}\right)\right)$$

Fonction gaussienne en 3D

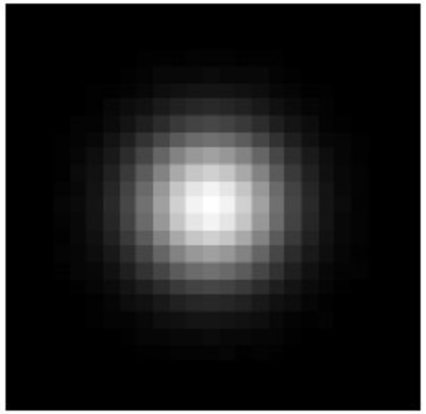


Image d'une gaussienne

$$\frac{1}{98}\times\begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 6 & 8 & 6 & 2 \\ 3 & 8 & 10 & 8 & 3 \\ 2 & 6 & 8 & 6 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

Le filtre Gaussien: *Exemple*

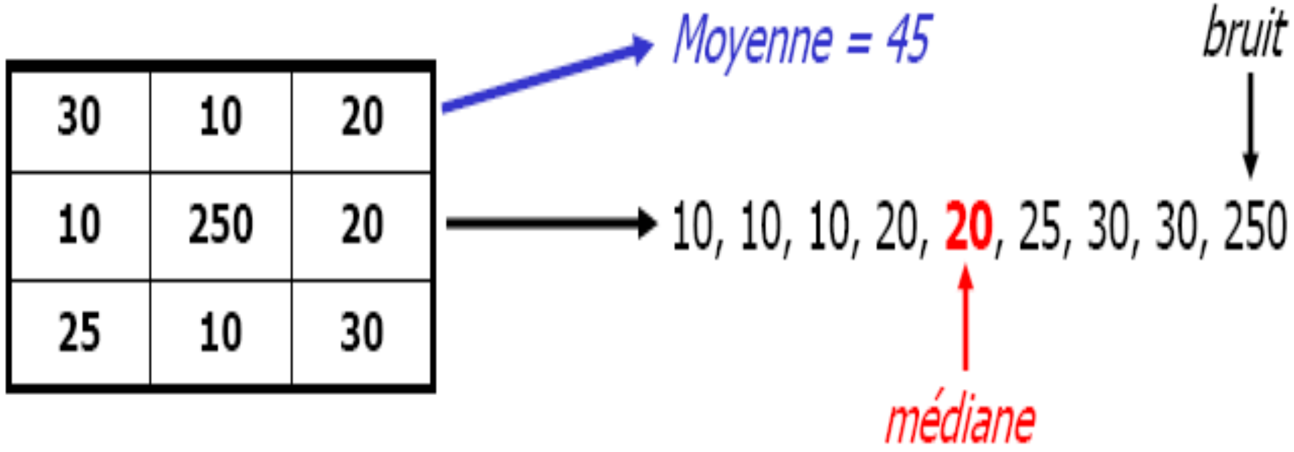
Original

Gauss 5x5

Gauss 11x11

Le filtre Médian

- Pour nettoyer le bruit dans une image, il existe mieux que le filtre Moyenneur ou le filtre Gaussien : le filtre Médian.
  - C'est un filtre non-linéaire, qui ne peut pas s'implémenter comme une convolution
  - On remplace la valeur d'un pixel par la valeur médiane dans son voisinage 2n+1 × 2n+1



Le filtre Médian: *Exemple 1*

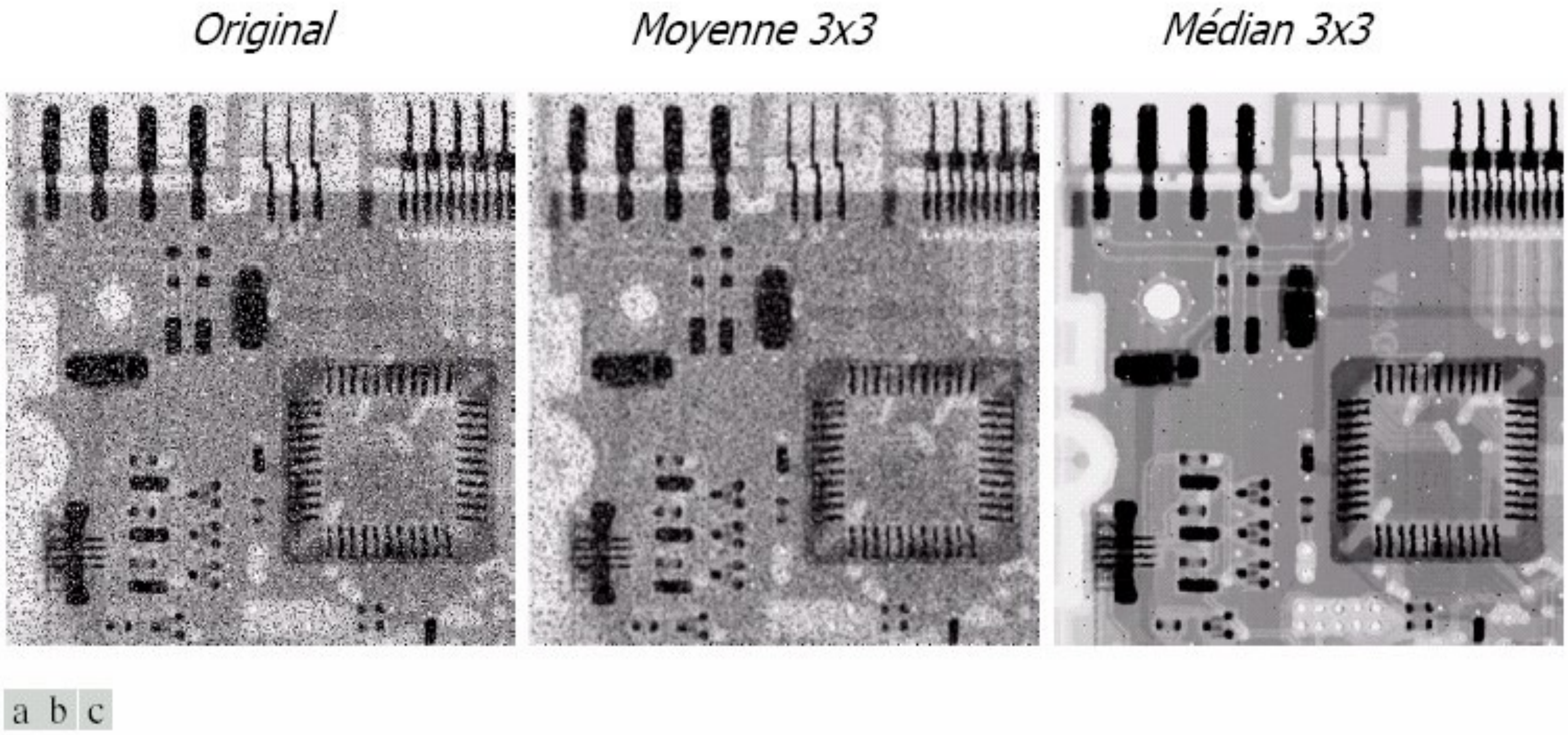
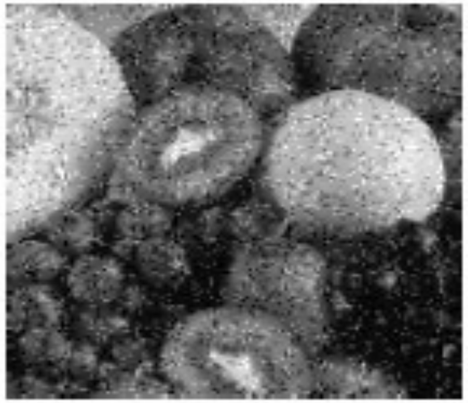
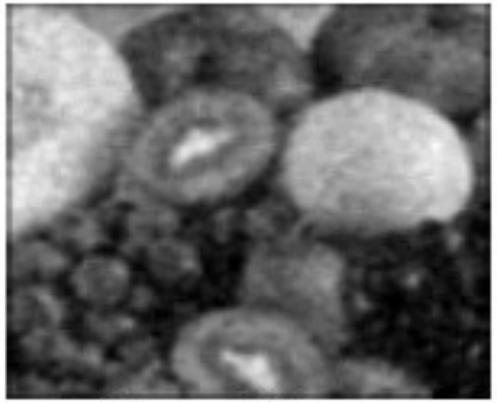
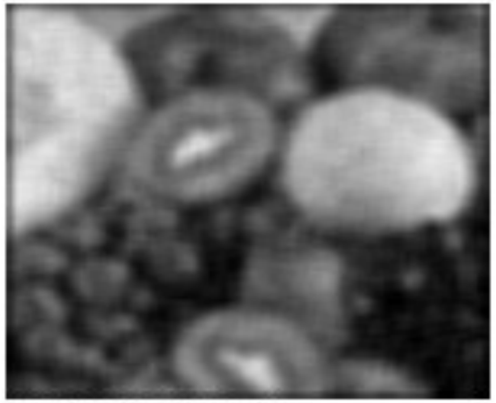


FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

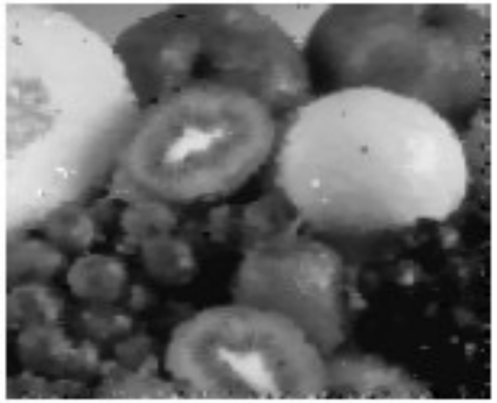
Le filtre Médian: *Exemple 2*


Bruit "poivre et sel"


3 X 3 Moyenne


5 X 5 Moyenne


7 X 7 Moyenne


Filtre médian

Le filtre Médian: *Exemple 3*

Image initiale

Bruit Poivre & Sel

Moyenne V8

Min V8

Max V8

Médian V8

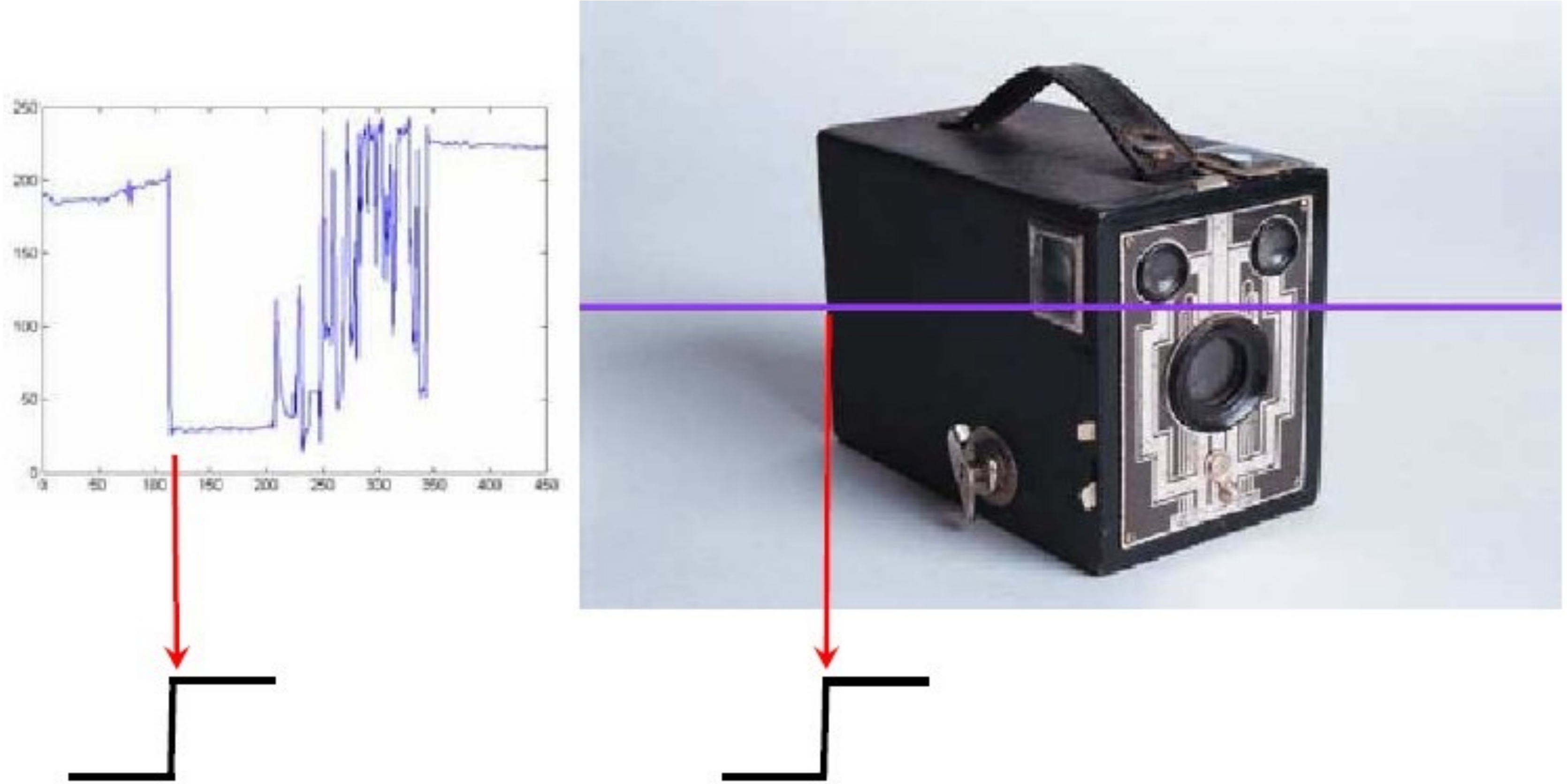
Détection de contours

Plan

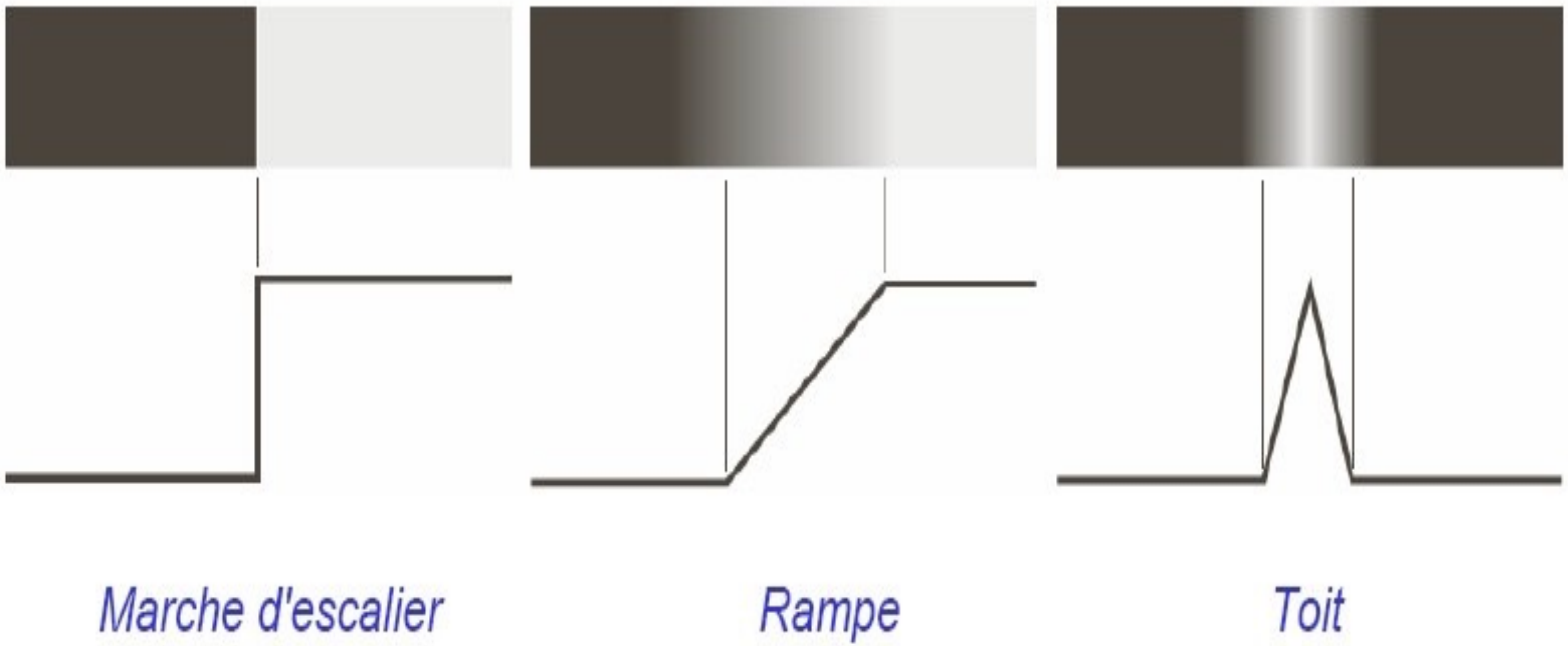
1) Qu'est-ce qu’un contour ?
2) Dérivées d'une image
3) Opérateurs: Gradient et Laplacien
4) Détecteur de points d’intérêts

Définition

Contour : frontière entre deux objets dans une image.
Définition plus large : discontinuité de l'image (variation brusque d'intensité).

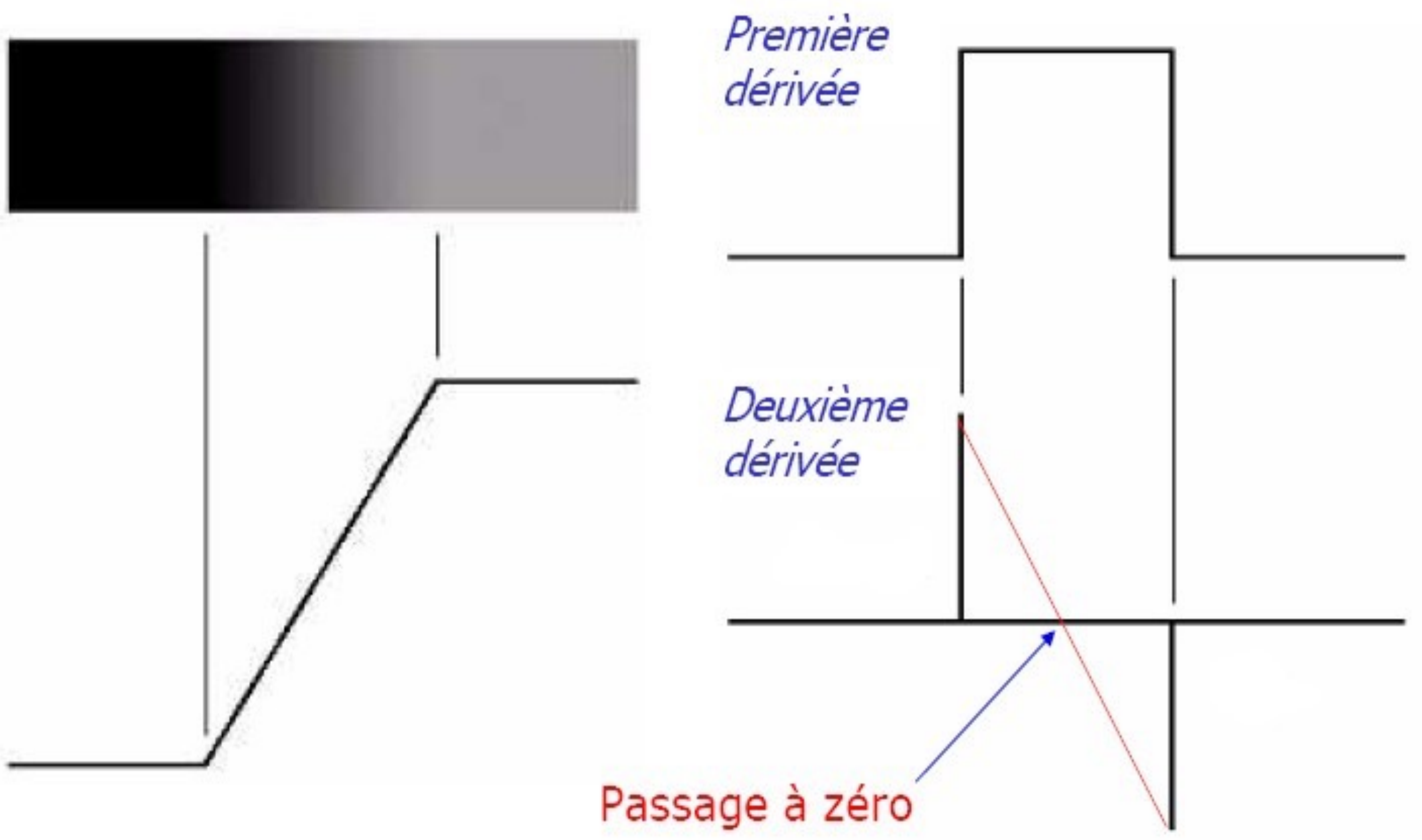


Différents types de contours

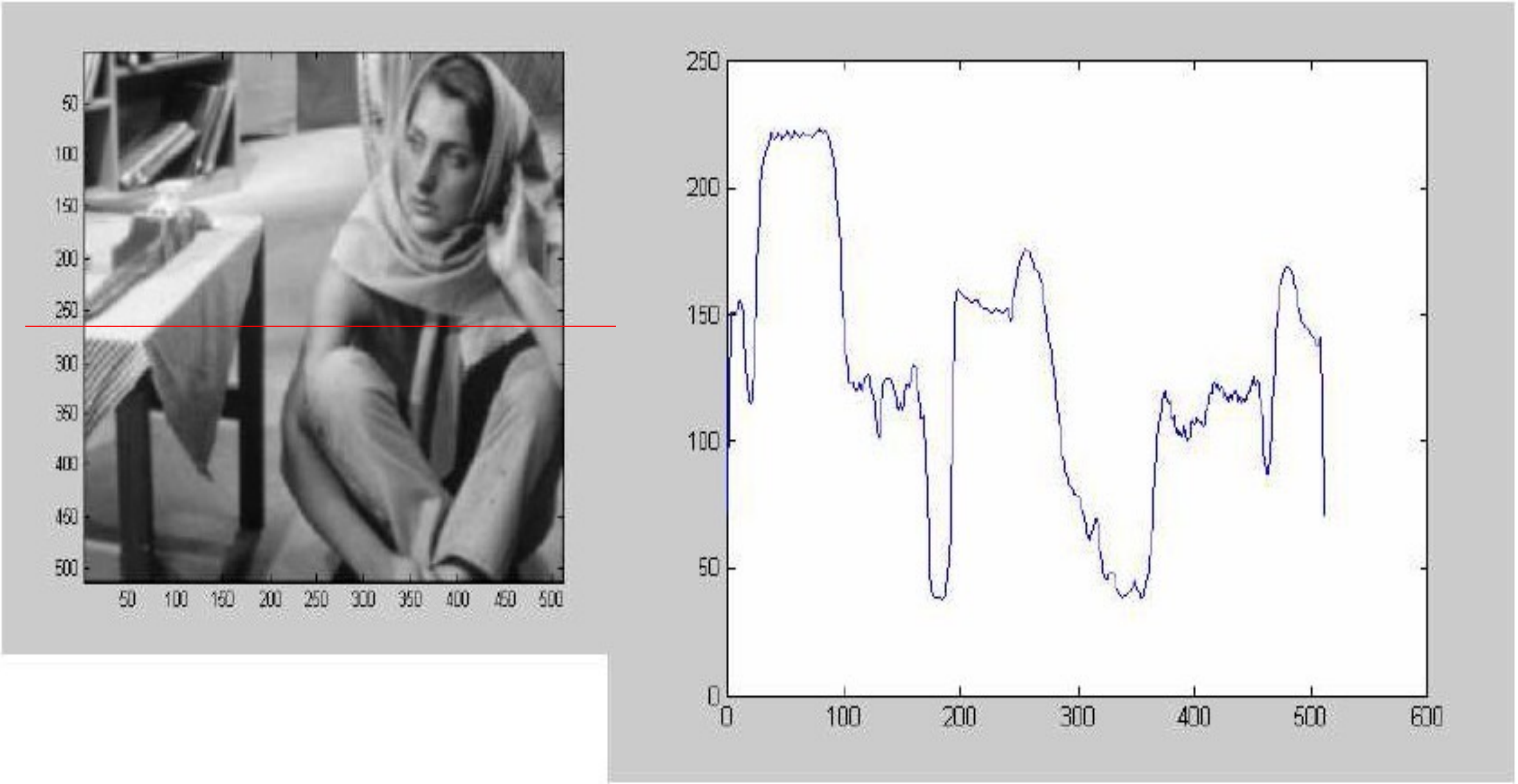


Source : Gonzalez and Woods. Digital Image Processing 3ed. Prentice-Hall, 2008.

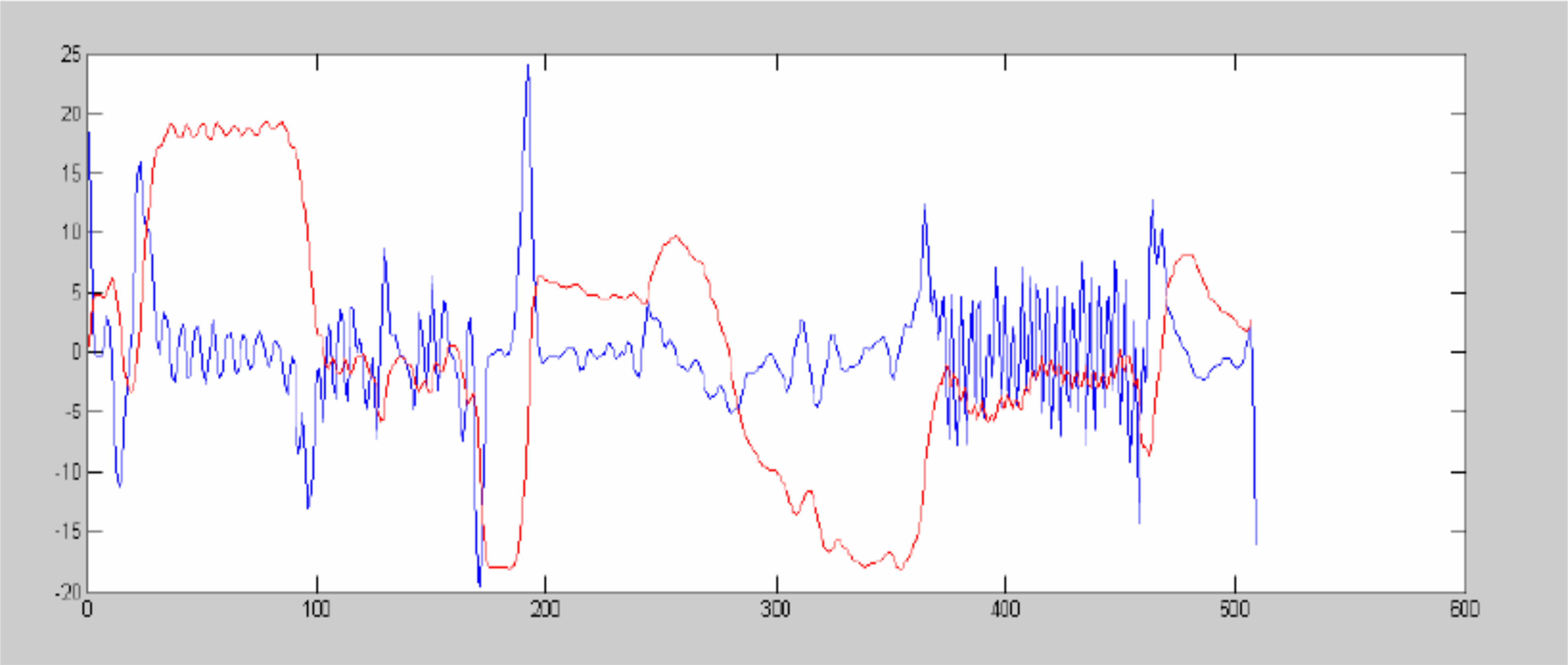
Dérivées d'un contour



Etude d'une ligne d'une image

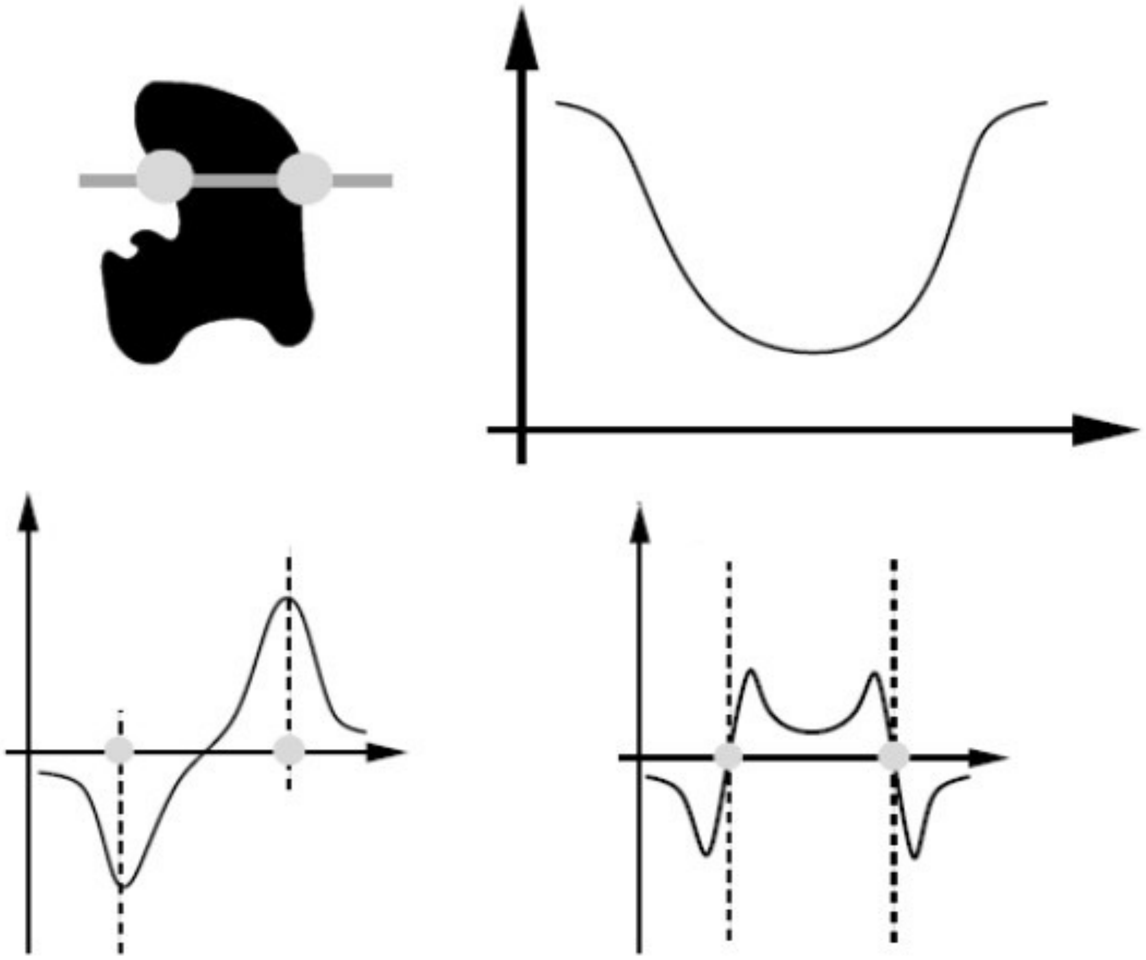


Etude d’une ligne d’une image : dérivée première

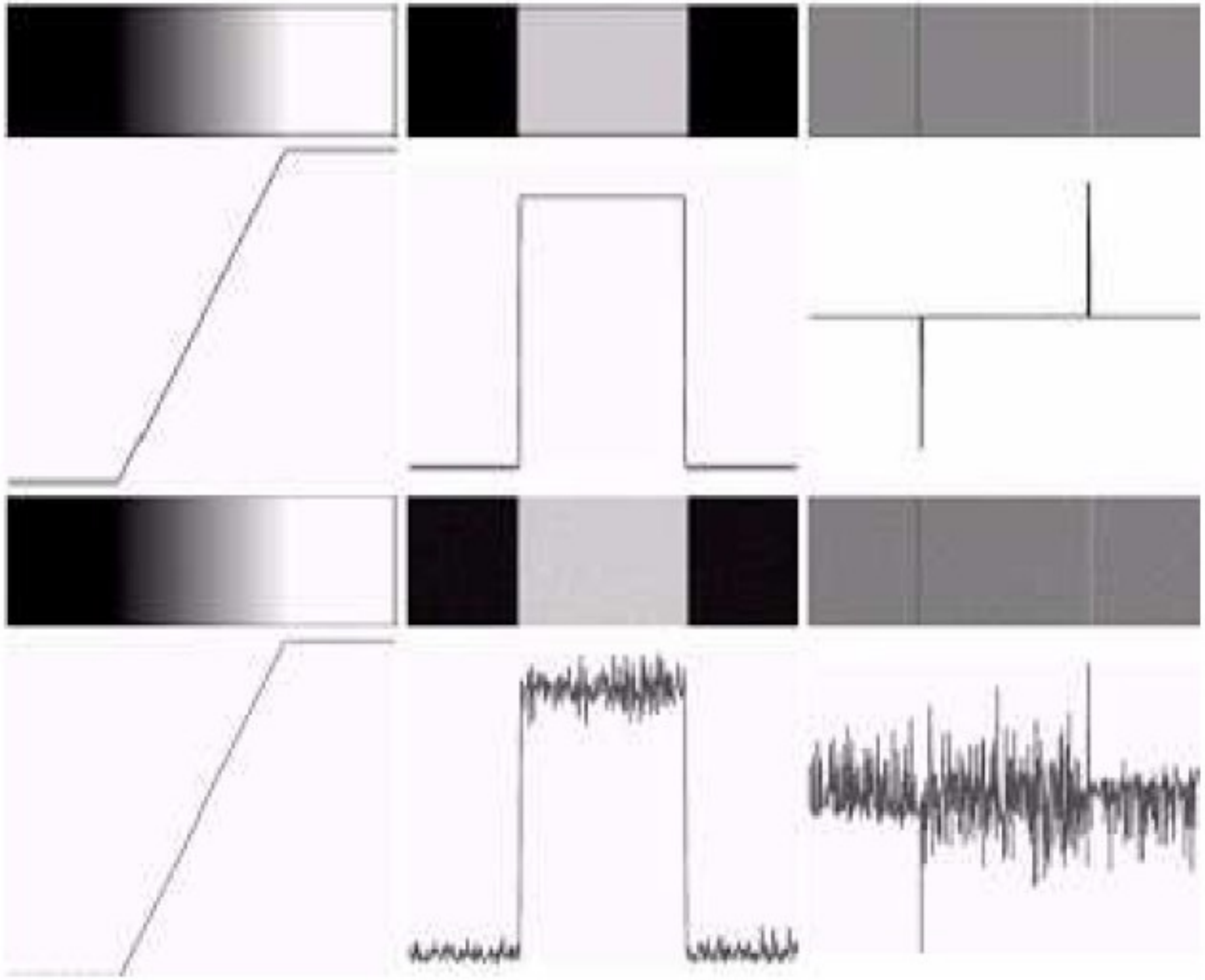


Détection de contours : principe

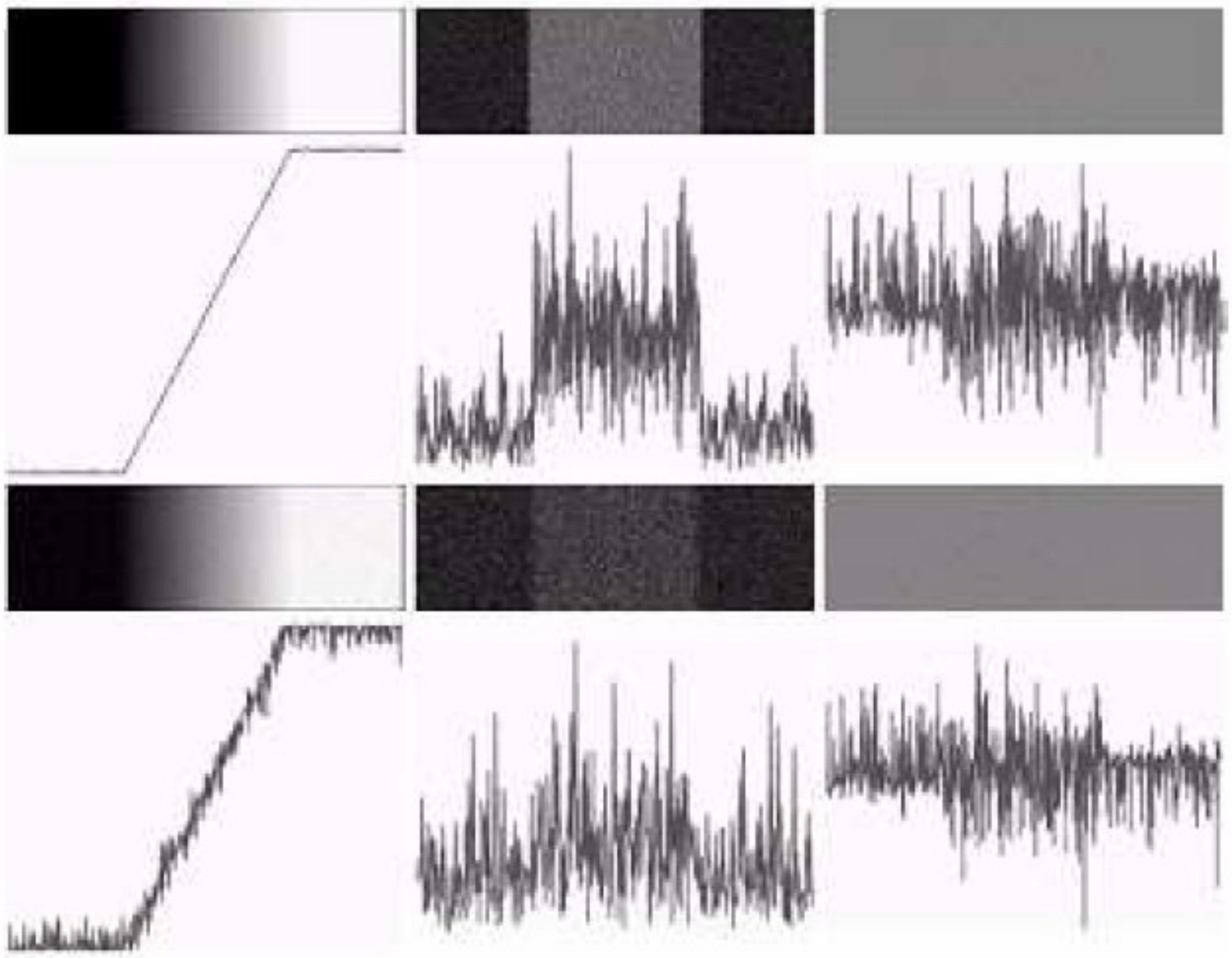
- Etude des dérivées de la fonction d'intensité dans l'image
- les extréma locaux de la dérivée première
- les passages par zéro de la dérivée seconde
- <u>difficulté</u> : la présence de bruit dans les images



Contours bruités (1)



Contours bruités (2)



Gradient : dérivée première de l'image (1)

- Rappel : l'image est une fonction 2D

$$I : (x, y) \to I(x, y)$$

- La première dérivée (gradient) de l'image est l'opérateur de base pour mesurer les contours dans l'image.

$$\vec{G} = (G_x, G_y) = \left(\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right)$$

Gradient : dérivée première de l'image (2)

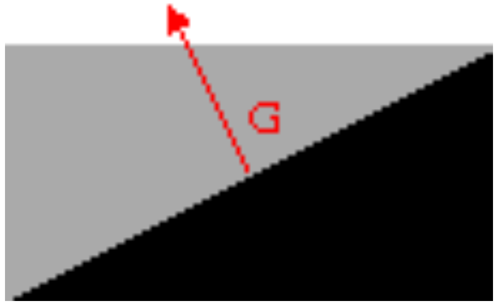
Le gradient peut être représenté en coordonnées polaires par un module *m* et une direction ϕ dans l'image.

- le module du gradient mesure la force du contour

$$m = \sqrt{G_x^2 + G_y^2}$$

- le gradient est un vecteur perpendiculaire au contour

$$\phi = arctan(\frac{G_y}{G_x})$$



Calcul du gradient

- On approxime les dérivées par "différences finies"

$$G_x(x, y) = I(x+1, y) - I(x-1, y)$$

- Calcul par convolution de l’image avec un masque de différences

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

Dérivation par différences finies: Opérateurs

- Opérateur de Prewitt :

$$h1 = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad h2 = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

- Opérateur de Sobel :

$$h1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad h2 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Pour limiter les effets du bruit, un lissage est compris dans le calcul (filtre moyenne pour Prewitt, filtre gaussien pour Sobel)

Il existe d'autres filtres plus sophistiqués donnant de meilleurs résultats (filtre de Canny, filtre de Deriche, filtre de Shen-Castan)

Exemples

Original

Gradient horizontal (Sobel)

Gradient vertical (Sobel)

Module du gradient de Sobel

Laplacien : deuxième dérivée de l'image

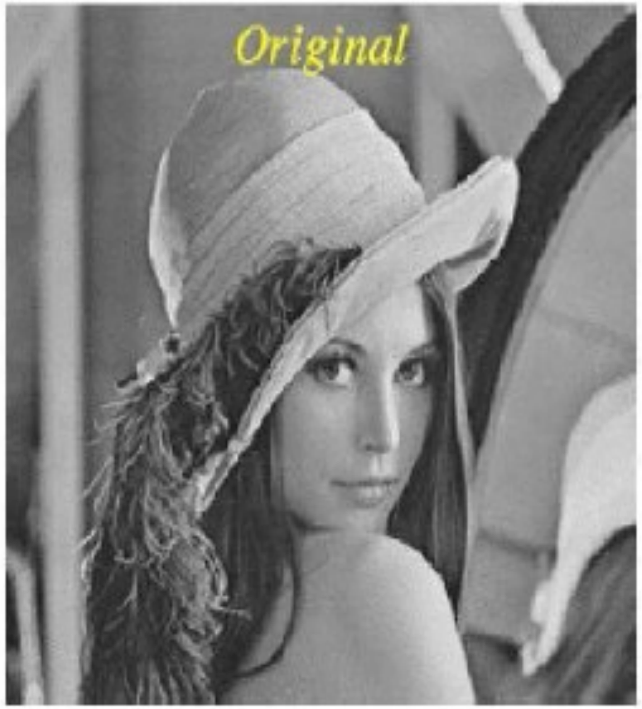
- Définition:

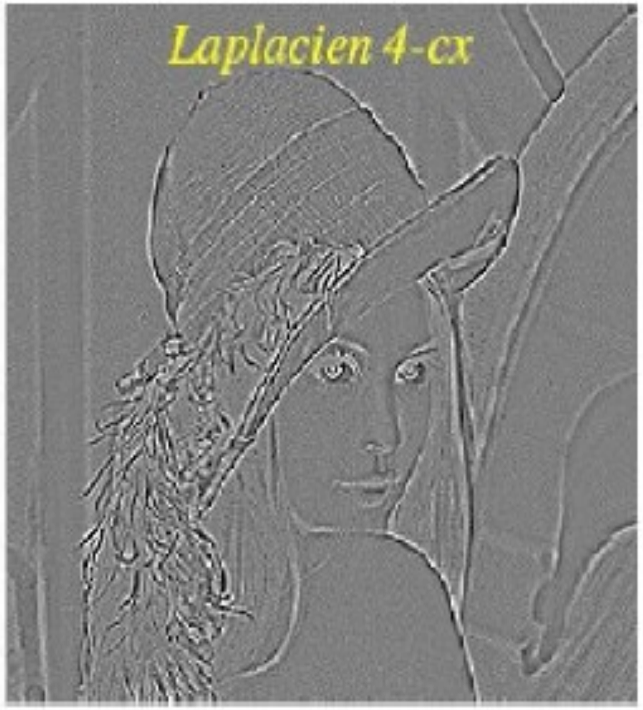
$$\nabla^2 I(x, y) = \frac{\partial^2 I}{\partial x^2}(x, y) + \frac{\partial^2 I}{\partial y^2}(x, y)$$

- Calcul par convolution avec les masques :

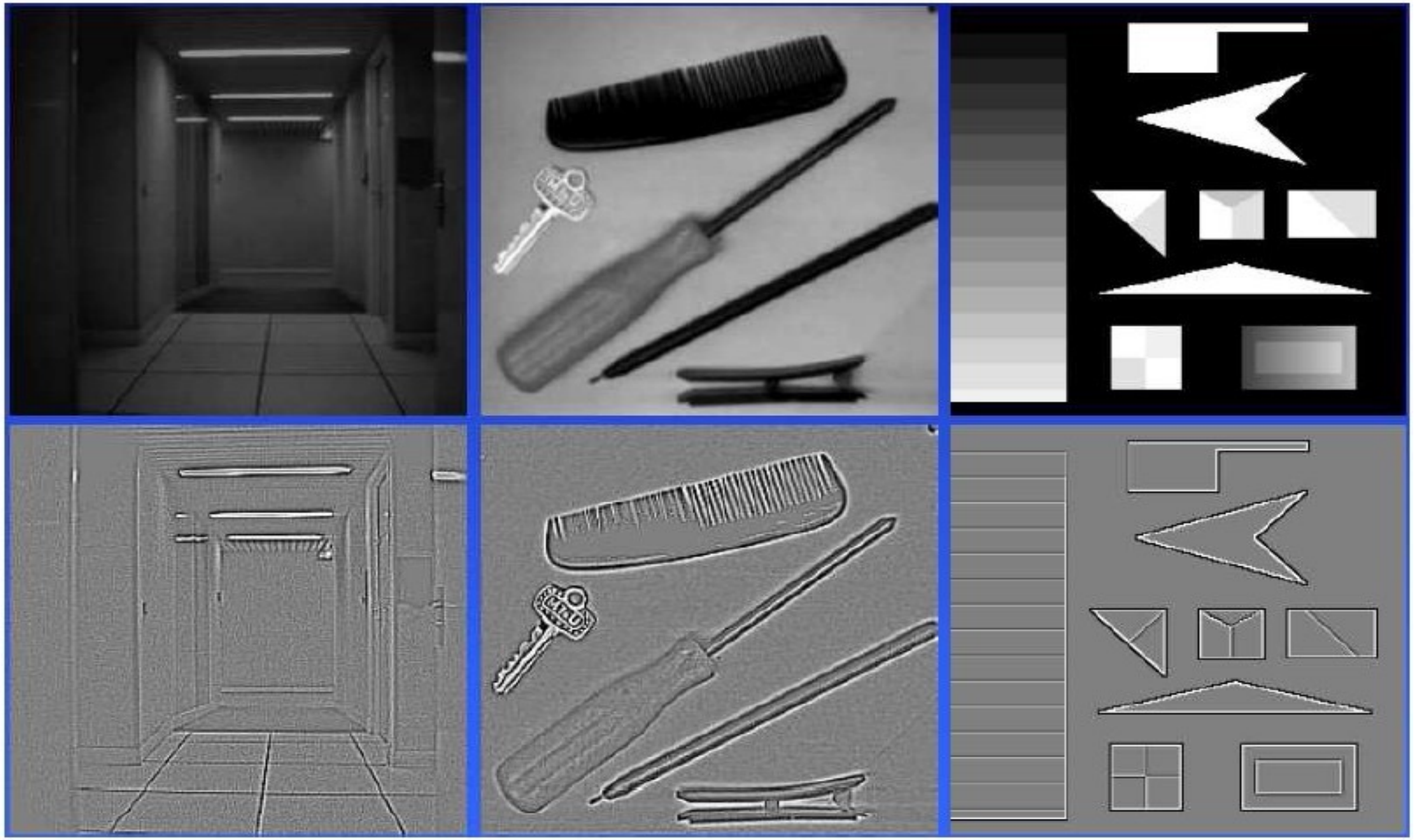
$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ ou } \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Calcul de Laplacien : exemple 1





Calcul de Laplacien : exemple 2



Autre exemple


Gradient


Gradient seuillé ($|G| > G_{min}$)


Seuil faible


Seuil grand