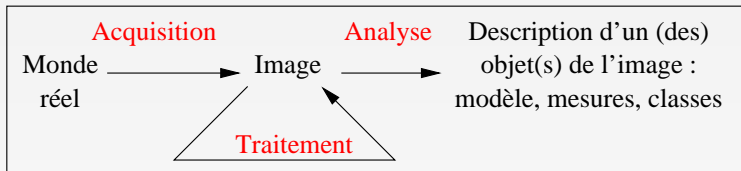


Traitement d'images

Bibliographie

- ▶ Cours de traitement d'images Elise Arnaud - Edmond Boyer Université Joseph Fourier
- ▶ Cours de traitement d'images Alain Boucher
- ▶ Cours de traitement d'images T Guyer Université de Chambéry
- ▶ Cours de traitement d'images Caroline ROUGIER université de Montréal

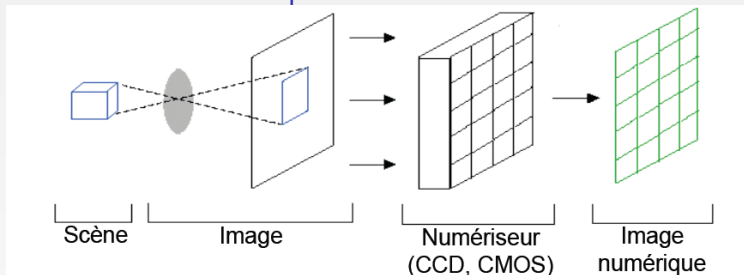


- ▶ Traitement (bas niveau) : suppression du bruit, augmentation de contraste, lissage, réhaussement des contours...
- ▶ Analyse (haut niveau) : extraction d'informations

Plan

1. Images / modèles de couleurs
2. Transformations d'histogramme : application au réhaussement de contraste
3. Transformations géométriques
4. Filtrage

Parenthèse sur l'acquisition



Numérisation = échantillonnage (nombre fini de points)
+
quantification (nombre fini de nuances)

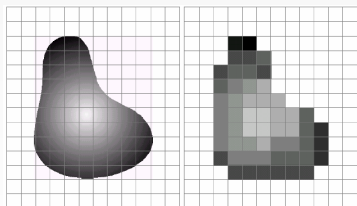
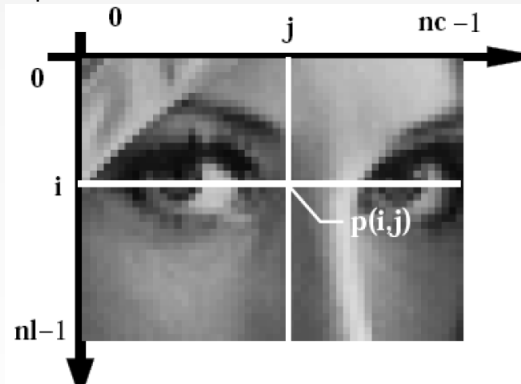


Image numérique

Un pixel est décrit par :

- ▶ ses coordonnées dans l'image (i, j)
- ▶ sa valeur $p(i, j)$, représentant son niveau de gris ou sa couleur

Une image numérique est une matrice de nombres : les valeurs des pixels



Résolution

La résolution est le nombre de points/pixels par unité de longueur



256x256



128x128



64x64



32x32

Exprimée en :

- ▶ dpi = ppi : dots/pixels per inch (points par pouce)
- ▶ ppc : points par centimètres

1 pouce = 2.54 cm

Calcul de résolution

L'une des trois valeurs suivantes s'obtient à partir des deux autres :

- ▶ Les dimensions physiques de l'image (sur un écran, imprimée) - ex. 30*20 cm
- ▶ La résolution de l'image - ex. 300 dpi
- ▶ La taille en pixels de l'image - ex. 3072*2048 pixels

Ecran du Nexus 6P : 2560 x 1440 pixels, 515ppp

En déduire la taille de la diagonale en pouces

Codage des valeurs des pixels

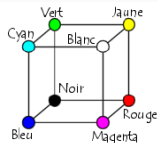
En niveau de gris

- ▶ la valeur $p(i, j)$ d'un pixel représente son intensité lumineuse
- ▶ binaire : $p(i, j) = 0$ noir ou $p(i, j) = 1$ blanc
- ▶ codage 8 bits (le plus classique) : $p(i, j) = 0, \dots, 255$ du plus foncé au plus clair

En couleur

- ▶ codage dans l'espace RGB : trois intensités lumineuses rouge, vert, bleu.
- ▶ codage 24 bits : $R(i, j) = 0, \dots, 255$; $V(i, j) = 0, \dots, 255$; $B(i, j) = 0, \dots, 255$

Information supplémentaire : transparence (canal alpha)



Un autre espace de couleurs : l'espace HSL

- ▶ Teinte - Hue : perception de la couleur (prune, orange, céladon)
- ▶ Saturation : pureté de la couleur (plus ou moins délavée)
- ▶ Luminance : intensité lumineuse (ombre et lumière)



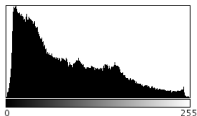
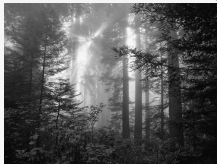
Définition des couleurs plus intuitive

Histogramme

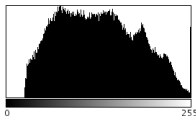
A chaque valeur on associe le nombre de pixels de l'image ayant cette valeur.

Algorithme (niveau de gris) :

```
for(i = 0; i < nl; i++)  
  for(j = 0; j < nc; j++)  
    hist[I(i, j)]++
```



Count: 187500 Min: 0
Mean: 82.163 Max: 255
StdDev: 60.492 Mode: 13 (2073)



Count: 187500 Min: 25
Mean: 127.573 Max: 255
StdDev: 55.361 Mode: 75 (1200)

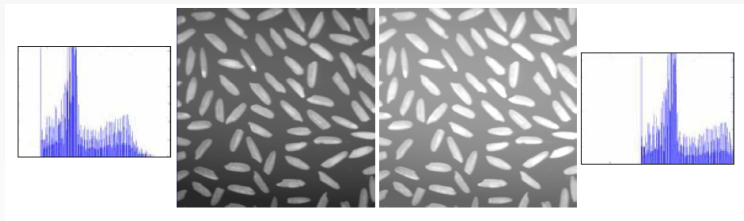


Count: 270000 Min: 2
Mean: 188.879 Max: 255
StdDev: 66.600 Mode: 229 (10857)

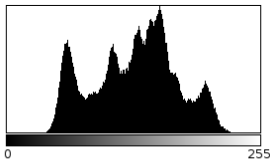
Dynamique : $D = [val_{min}, val_{max}]$

Luminance ou brillance d'une image

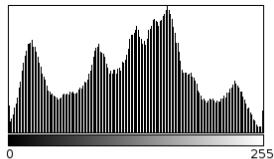
- ▶ La luminance (ou brillance) est définie comme la moyenne de tous les pixels de l'image.
- ▶ Pour augmenter la luminance, il suffit de décaler l'histogramme : $I'(i,j) = I(i,j) + b$
- ▶ Dans les deux images suivantes, seule la luminance est différente



Contraste



Count: 262144 Min: 35
Mean: 128.229 Max: 240
StdDev: 42.764 Mode: 154 (3171)



Count: 262144 Min: 0
Mean: 121.482 Max: 255
StdDev: 62.646 Mode: 159 (3171)

Un indicateur de contraste : variance des niveaux de gris

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

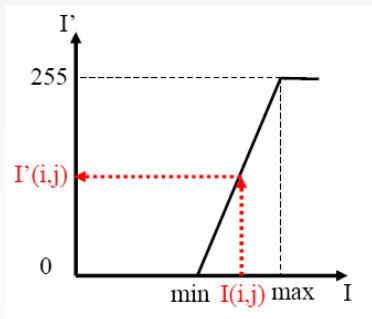
Comment améliorer le contraste ?

- ▶ Extension linéaire de dynamique
- ▶ Transformation linéaire par morceau
- ▶ Transformation non-linéaire
- ▶ Égalisation de l'histogramme

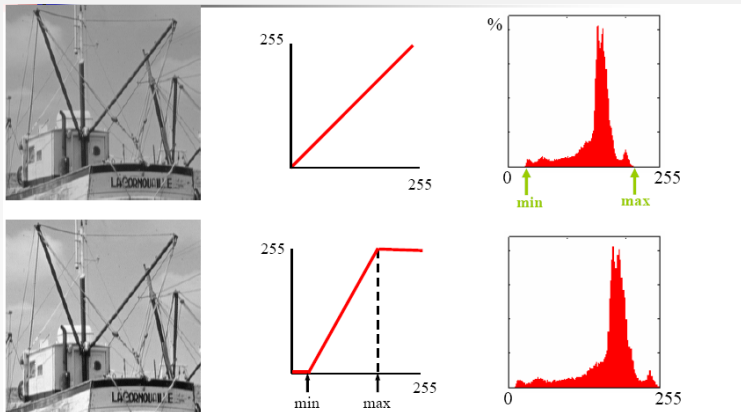
Extension linéaire de dynamique

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

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



Extension linéaire de dynamique : exemple



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

Extension linéaire de dynamique : implémentation

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)]
```

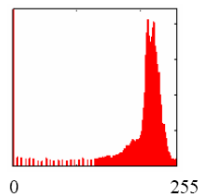
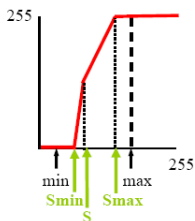
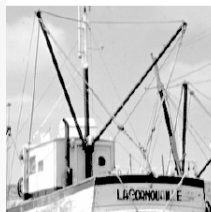
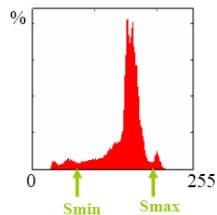
Limite

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



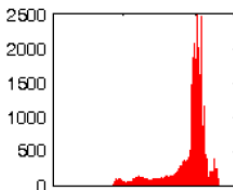
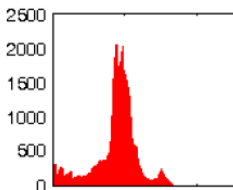
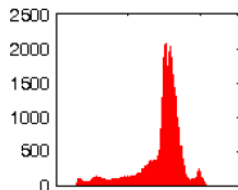
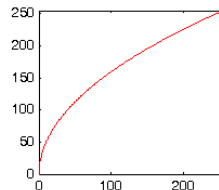
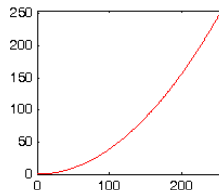
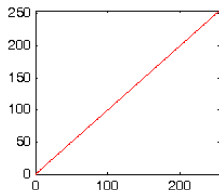
Source : www.crans.ens-cachan.fr/~genest/histogramme.pdf

Transformation linéaire par morceaux



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

Transformation non linéaire



Egalisation d'histogramme

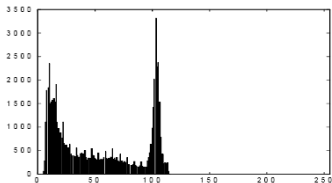


Image originale

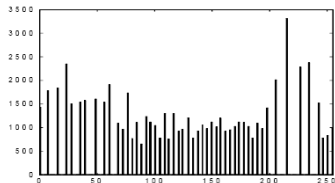
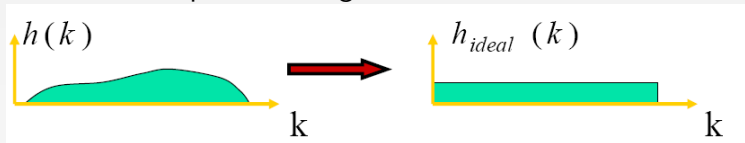


Image plus contrastée

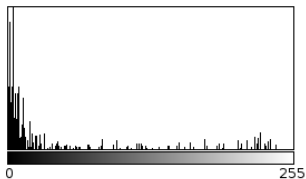
Egalisation d'histogramme : méthode

On cherche à aplanir l'histogramme

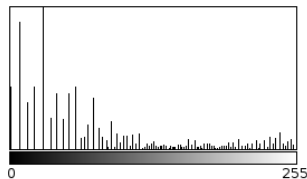


1. Calcul de l'histogramme $h(k)$ avec $k \in [0, 255]$
2. Histogramme cumulé $C(k) = \sum_{i=1}^k (h(i))$
3. Transformation des niveaux de gris de l'image
$$I'(x, y) = \frac{C(I(x, y)) * 255}{N}$$

Egalisation d'histogramme : exemple



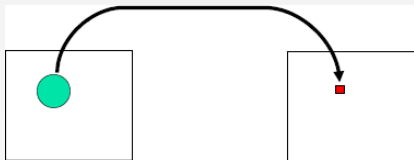
Count: 79524 Min: 0
Mean: 43.991 Max: 250
StdDev: 70.204 Mode: 4 (8764)



Count: 79524 Min: 0
Mean: 82.692 Max: 255
StdDev: 73.032 Mode: 29 (8764)

Convolution

- ▶ Transformation locale : utilisation du voisinage de chaque pixel



- ▶ Un filtre de convolution (ou masque ou noyau) est généralement une matrice $2n + 1 \times 2n + 1$.
- ▶ Calcul : somme de produits, on parle de filtre linéaire

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

Convolution : exemple



Image d'origine

*

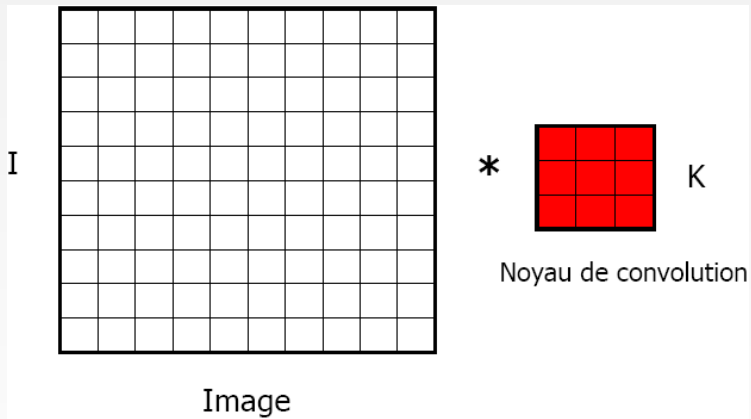


=

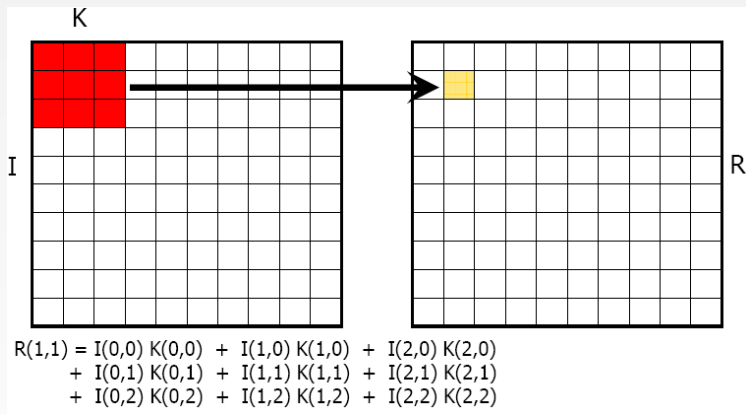


Image convoluée
(résultat)

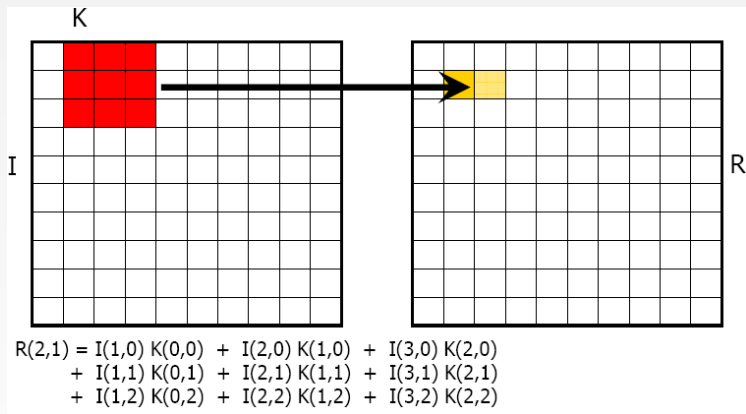
Convolution : détail 1/5



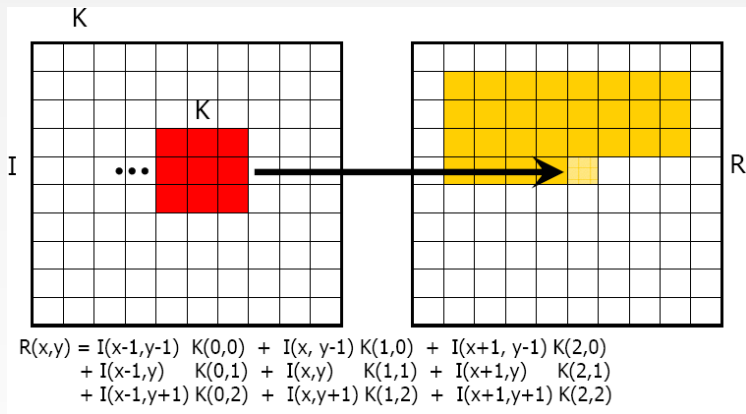
Convolution : détail 2/5



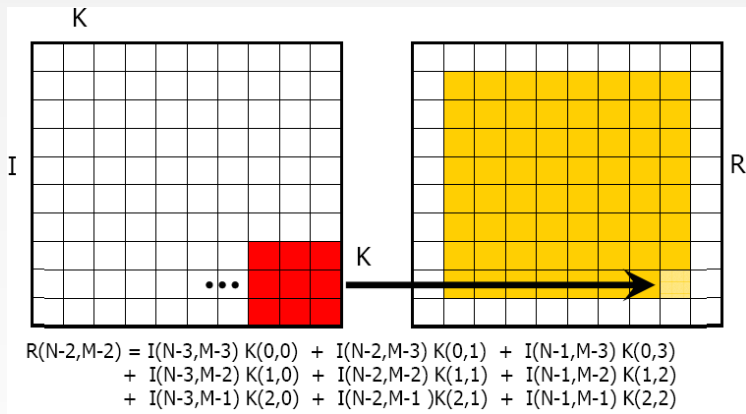
Convolution : détail 3/5



Convolution : détail 4/5



Convolution : détail 5/5



Familles de filtres

- ▶ Filtre passe-bas : atténue le bruit et les détails
- ▶ Filtre passe-haut : accentue les détails et les contours



Le filtre moyeneur

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)

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

 ou $1/9$

1	1	1
1	1	1
1	1	1

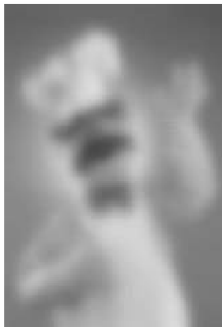
3x3

Filtre moyenneur : exemples

Plus le filtre grossit, plus le lissage devient important.



Original



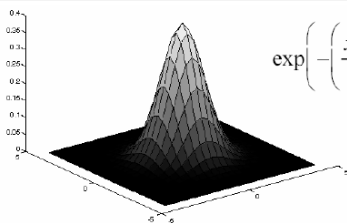
Moyenne 5x5



Moyenne 11x11

Le filtre Gaussien

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



Fonction gaussienne en 3D

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

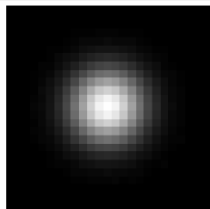


Image d'une gaussienne

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

Filtre Gaussien : exemple



Original



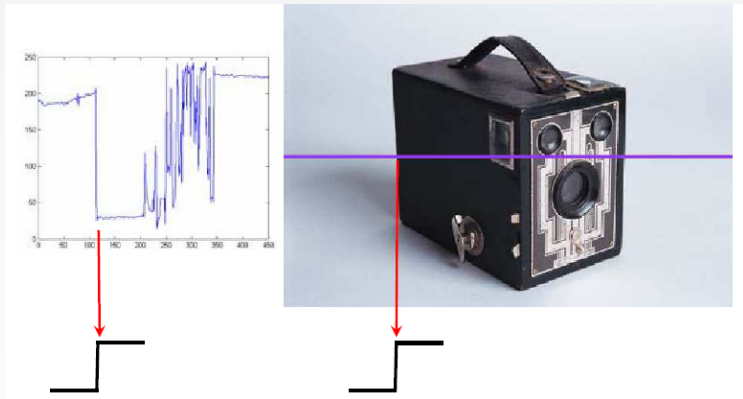
Gauss 5x5



Gauss 11x11

Qu'est ce qu'un contour ?

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



Discontinuités de l'image

Toute discontinuité n'est pas nécessairement située à la frontière entre deux objets.

de profondeur



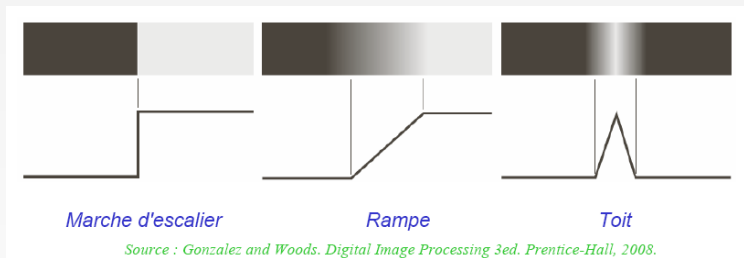
d'orientation
de surface

de réflectance

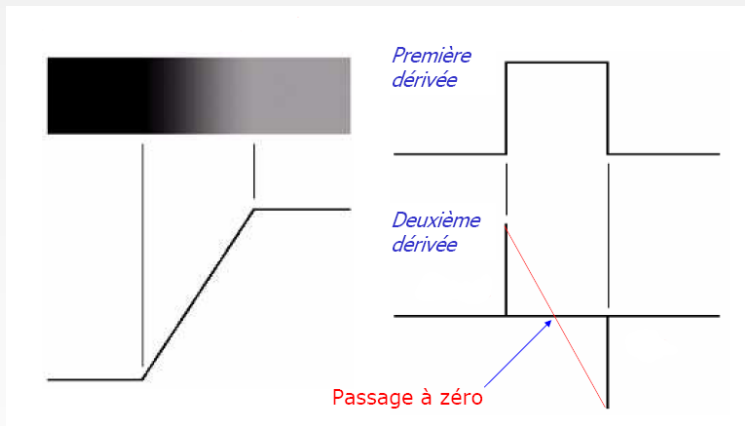
d'illumination

Source : Jacques-André Landry. *Vision robotique*. ETS.

Différents types de contours



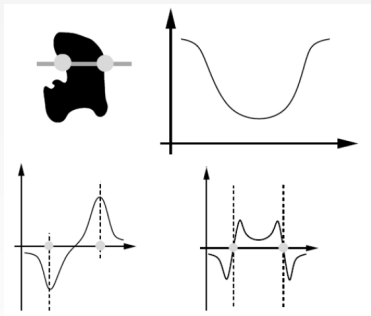
Dérivées d'un contour



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
- ▶ difficulté : la présence de bruit dans les images



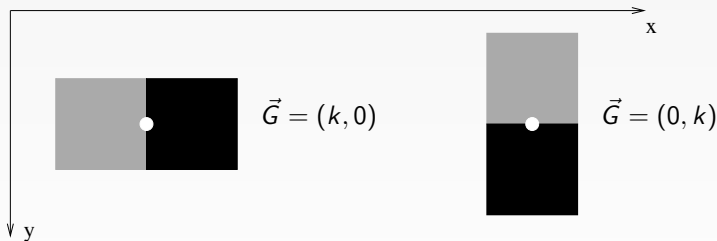
Gradient : dérivée première de l'image 1/2

Rappel : l'image est une fonction 2D.

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

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 2/2

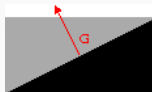
Le gradient peut être représenté en coordonnées polaires par un module m et une direction ϕ dans l'image.

- ▶ 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**

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



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*

Question : comment ces images sont-elles créées à partir du résultat brut de la convolution (qui peut donner par exemple des valeurs négatives) ?

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

