Traitement d'image - filtrage

Ecole Doctorale MIIS - 2017

Sebastien Kramm

LITIS Rouen

28 juin 2017

(Avec des diapos de Anne Vialard, LABRI)

Sommaire

Filtrage : introduction

- Definition : filtrer une image : lui appliquer une transformation globale
  - la position de chaque pixel est inchangée
  - sa valeur est fonction de son voisinage et du filtre considéré
- Une image est composée de signaux haute fréquence et basse fréquence (Fourier)
- De façon similaire au signal 1D :
  - filtre passe-haut : ("intégration")
    - 1D : laisse passer les hautes fréquences
    - 2D : laisse passer les variations brutales (contours)
  - filtre passe-bas : ("dérivation")
    - 1D : laisse passer les basses fréquences
    - 2D : atténue les variations brutales (bruit) $\rightarrow$ filtre flou, ou lissant (*blurring*)
- Utilisations :
  - Réduction de bruit
  - Détection de *feature* (contours, points)

Classification des filtres

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

- Filtres **linéaires** ($f(x_1 + x_2) = f(x_1) + f(x_2)$) ou filtres **non-linéaires**

Sous-sommaire

Filtrage : aspect mathématique

- Basé sur l'opération de **convolution**, dans le domaine continu ou discret :

$$f * g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) g(u - x, v - y) du dv$$

$$f * g(x, y) = \sum_{u=-\infty}^{\infty} \sum_{v=-\infty}^{\infty} f(x, y) g(u - x, v - y)$$

Filtrage : aspect mathématique

- Basé sur l'opération de **convolution**, dans le domaine continu ou discret :

$$f * g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)g(u-x,v-y)dudv$$

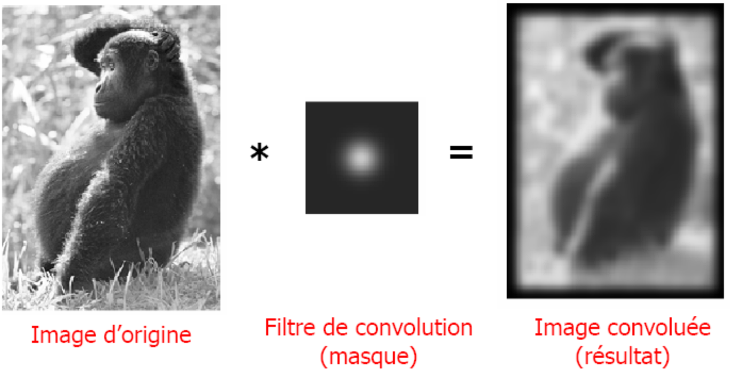
$$f * g(x,y) = \sum_{u=-\infty}^{\infty}\sum_{v=-\infty}^{\infty} f(x,y)g(u-x,v-y)$$

- En pratique, on utilise un noyau carré de coté $2\,n+1$
  Pour un noyau 3x3 :

$$I_2(i,j) = \sum_{k=0}^{2}\sum_{l=0}^{2} I_1(i+k-1,j+l-1)K(k,l)$$

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

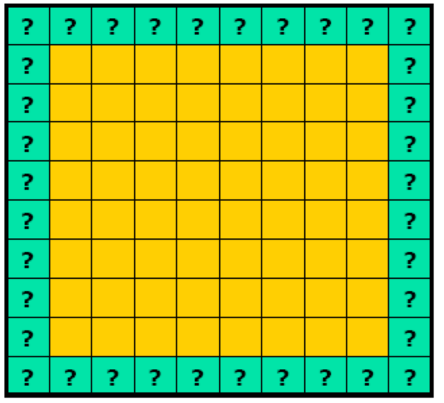
Convolution : exemple



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



En pratique

Coût calculatoire : K^2 opérations par pixel pour un noyau de taille $K \times K$
Solutions :

- on reste souvent avec des noyaux de taille réduite
- De nombreuses optimisations possibles (compromis mémoire / CPU)
- si le noyau est séparable : convolution 2D : 2 convolutions 1D successives
  $\rightarrow$ Complexité : $k^2 \rightarrow 2\,k$

Comment savoir si un noyau donné est séparable ?
Réponse : décomposition SVD du noyau
Si une seule valeur singulière non-nulle $\Rightarrow$ séparable

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)

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

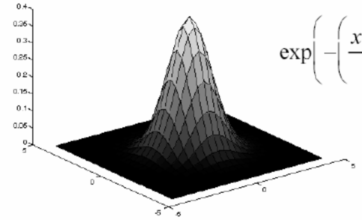
3x3

Filtre moyenneur : exemples

Plus le filtre grossit, plus le lissage devient important.

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

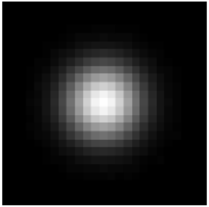
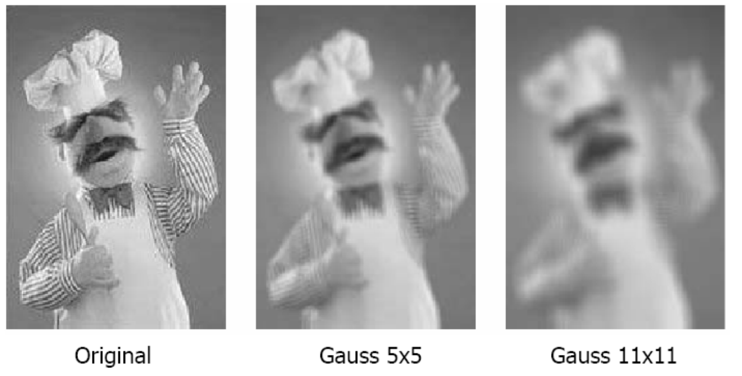


Image d'une gaussienne

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

Filtre Gaussien : exemple



Sous-sommaire

Filtre médian

- Principe : on remplace chaque pixel par la valeur médiane de son voisinage.
- Exemple pour un noyau 3×3 :
  - soit la fraction d'image :

5	6	7
6	111	8
7	8	9

  - on trie les 9 valeurs de façon croissante :

5	6	6	7	**7**	8	8	9	111

  - on remplace le pixel central par la valeur de la médiane (ici, 7) :

5	6	7
6	7	8
7	8	9

Filtre médian

- Principe : on remplace chaque pixel par la valeur médiane de son voisinage.
- Exemple pour un noyau 3×3 :
  - soit la fraction d'image :

5	6	7
6	111	8
7	8	9

  - on trie les 9 valeurs de façon croissante :

5	6	6	7	**7**	8	8	9	111

  - on remplace le pixel central par la valeur de la médiane (ici, 7) :

5	6	7
6	7	8
7	8	9

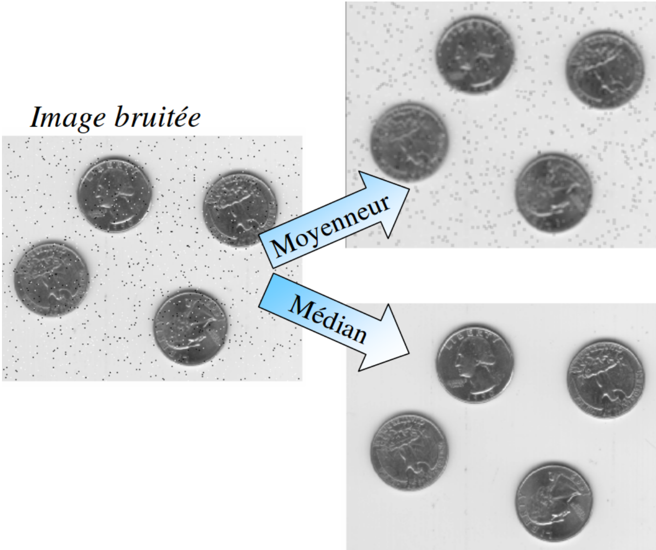
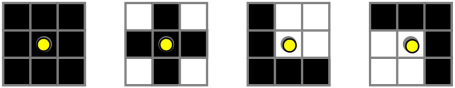


Image bruitée
Moyenneur
Médian

Sous-sommaire

Morphologie mathématique - 1

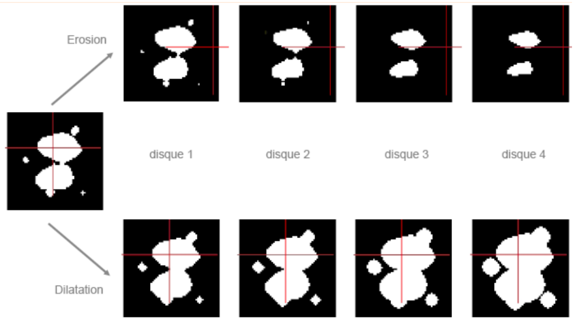
- S'applique sur une image binaire I, qui est considérée comme un ensemble $E = \{0, 1\}$
- Opérations de base : érosion / dilatation par un élement structurant S, doté d'une origine.
  Ces 2 opérations sont **duales**.
- Exemple d'élement structurant S : (noir = 1)



- Principe : on fait glisser S sur toutes les positions de l'image, on le compare avec l'image locale, et on fixe le pixel de I à 0 ou 1 en fonction.

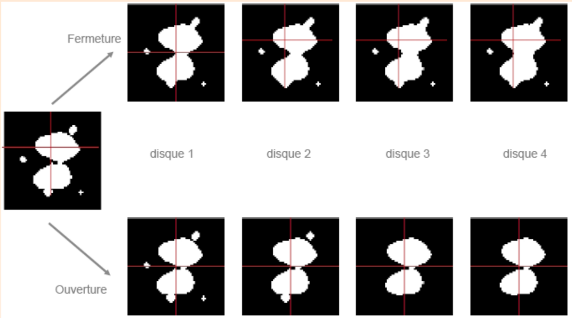
Erosion & dilatation

- Dilatation $\rightarrow$ on fixe à 1 si l'intersection n'est pas vide.
- Erosion $\rightarrow$ on fixe à 1 uniquement si tout S est englobé dans la zone.



Morphologie mathématique - ouverture / fermeture

- Opérateurs composés, utilisent le même élement structurant.
  - Ouverture : érosion suivie d'une dilatation
  - fermeture : dilatation suivie d'une érosion



Morphologie mathématique - Applications

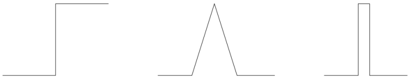
- Applications :
  - élimination du bruit
  - extraction de contours
  - remplissage de régions
  - amincissement et épaississement

Sommaire

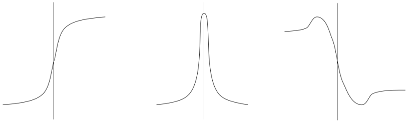
Détection de contour par filtrage

Qu'est ce qu'un contour ?

- Point de vue monodimensionnel : signal sur une ligne image



- Fonction échelon réelle et ses dérivées successives



$\Rightarrow$ On peut utiliser des filtres pour obtenir les contours pertinents, via un **seuillage**.

Gradient d'une image

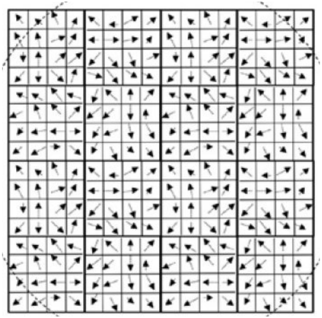
- Une image est un signal 2D : on peut calculer en tout point sa dérivée

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

- 2D : il est composé d'un module m et une direction ϕ :

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

$$\phi = \arctan(\frac{\partial I(x,y)}{\partial y} / \frac{\partial I(x,y)}{\partial x})$$



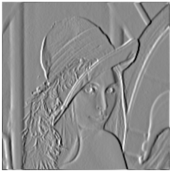
Calcul du gradient

- On peut en obtenir une bonne approximation aved le **filtre de Sobel** (1972)

$$h1 = 1/4 \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad h2 = 1/4 \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

- Rapide : est séparable en deux noyaux : $[1 \quad 2 \quad 1] * [-1 \quad 0 \quad -1]$


image originale


dérivée horizontale


dérivée verticale

Laplacien d'une image

- Le Laplacien correspond à la dérivée seconde

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

- On peut approximer par une différence sur chaque axe :

$$\frac{\partial^2 f}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2\,f(x,y)$$
$$\frac{\partial^2 f}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2\,f(x,y)$$

- Soit :

$$\nabla^2 f = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4\,f(x,y)$$

- Le noyau correspondant est :

Sur 4 voisins

0	1	0
1	-4	1
0	1	0

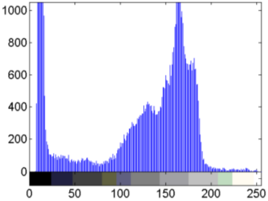
On peut le définir en utilisant 8 voisins :

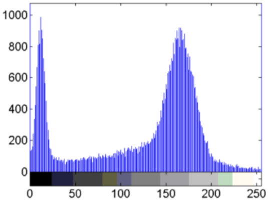
1	1	1
1	-8	1
1	1	1

Laplacien

(a)

(b)





Détection de contour

Deux approches :

- Approche gradient : détermination des extréma locaux dans la direction du gradient.
- Approche laplacien : détermination des passages par zéro du laplacien.

Détection de contour

Deux approches :

- Approche gradient : détermination des extréma locaux dans la direction du gradient.
- Approche laplacien : détermination des passages par zéro du laplacien.

Détection de contour via le Laplacien

- Le Laplacien est très sensible au bruit, on préfère l'appliquer sur une image préfiltrée par un noyau Gaussien
- On peut précalculer le noyau correspondant

$$LoG(x,y) = \frac{x^2+y^2-\sigma^2}{\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- On parle de LoG (*Laplacian of Gaussian*)
- Variante : DoG (*Difference of Gaussian*)

Détecteur via LoG

Détecteur Canny (John F. Canny, 1986)

Propriétés

- faible taux d'erreur dans la signalisation des contours,
- minimisation des distances entre les contours détectés et les contours réels,
- une seule réponse par contour et pas de faux positifs.

Détecteur Canny (John F. Canny, 1986)

Propriétés

- faible taux d'erreur dans la signalisation des contours,
- minimisation des distances entre les contours détectés et les contours réels,
- une seule réponse par contour et pas de faux positifs.

Etapes

- réduction du bruit par filtrage Gaussien (noyau 5 x 5, $\sigma = 1,4$),
- calcul du gradient par deux masques de convolution 3 x 1 (H et V), $G_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$ et $G_y = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}^T$
- suppression des non-maxima locaux,
- seuillage à hysteresis (2 seuils) : si l'intensité de son gradient est :
  - Inférieur au seuil bas, le point est rejeté ;
  - Supérieur au seuil haut, le point est accepté comme formant un contour ;
  - Entre le seuil bas et le seuil haut, le point est accepté s'il est connecté à un point déjà accepté.

Détecteur Canny