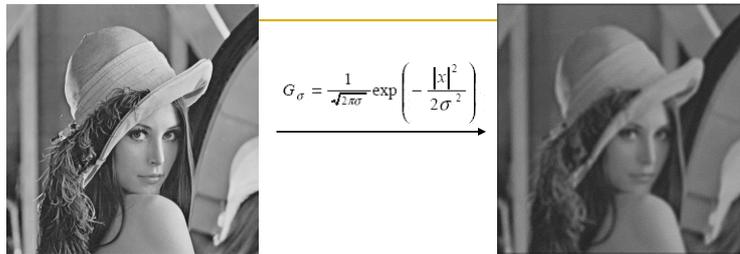


Traitement d'Image :

Filtrage d'image



Aymeric Histace

1

Plan

- 1. Rappel
- 2. Problématique du filtrage
- 3. Filtres de lissage
- 4. Filtres dérivateurs
- 5. Transformée de Hough
- 6. Restauration : initiation

Aymeric Histace

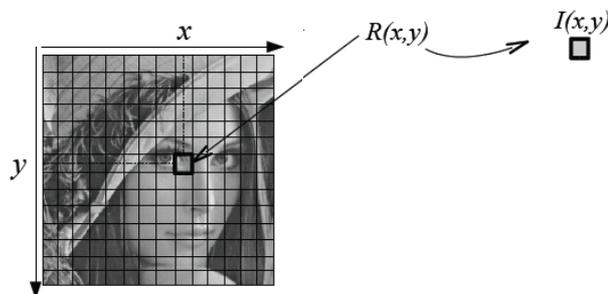
2

Plan

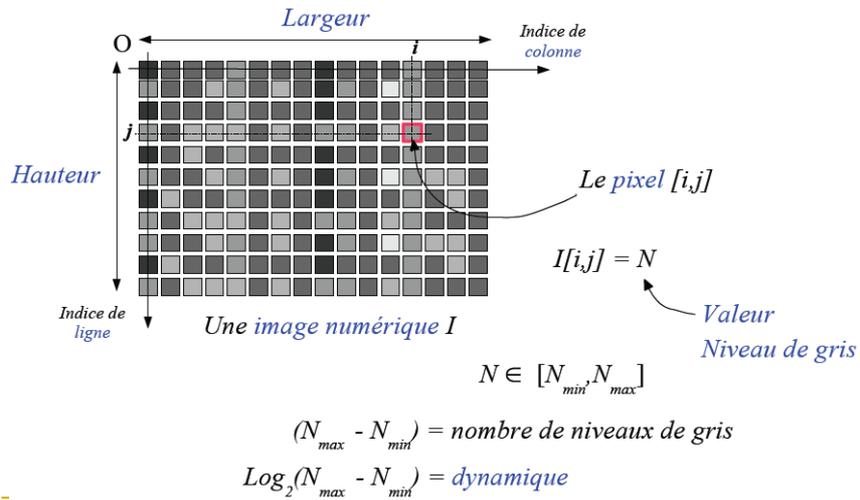
- 1. **Rappel**
- 2. Problématique du filtrage
- 3. Filtres de lissage
- 4. Filtres dérivateurs
- 5. Transformée de Hough
- 6. Restauration : initiation

1. Rappel

- **L'échantillonnage** est le procédé de discrétisation spatiale d'une image consistant à associer à chaque zone rectangulaire $R(x,y)$ d'une image continue une unique valeur $I(x,y)$.



1. Rappel



1. Outils fondamentaux (Rappel)

■ La convolution :

- Soit I une image numérique et h une fonction de $[x_1, x_2] \times [y_1, y_2]$ à valeurs réelles.
- La convolution discrète de I par h est définie par :

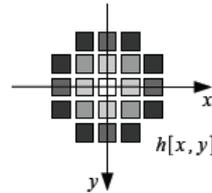
$$(I * h)[x, y] = \sum_{i=x_1}^{x_2} \sum_{j=y_1}^{y_2} h(i, j) \cdot I[x - i, y - j]$$

1. Outils fondamentaux (Rappel)

■ La convolution :

□ Propriétés de la convolution :

- Commutativité ;
- Associativité ;
- Distributivité.

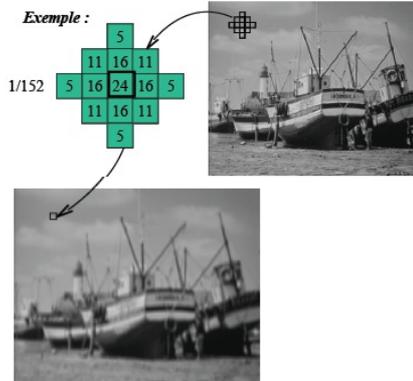


Les nouvelles valeurs du pixel sont calculées par *produit scalaire* entre le noyau de convolution et le *voisinage* correspondant du pixel.

1. Outils fondamentaux (Rappel)

■ La convolution :

Pour calculer une convolution, on remplace la valeur de chaque pixel par la valeur du produit scalaire entre les valeurs du noyau de convolution et les valeurs du voisinage du pixel considéré (par rapport à l'origine (0,0) du noyau de convolution).



1. Outils fondamentaux (Rappel)

■ La transformée de Fourier

- TF discrète bidimensionnelle directe

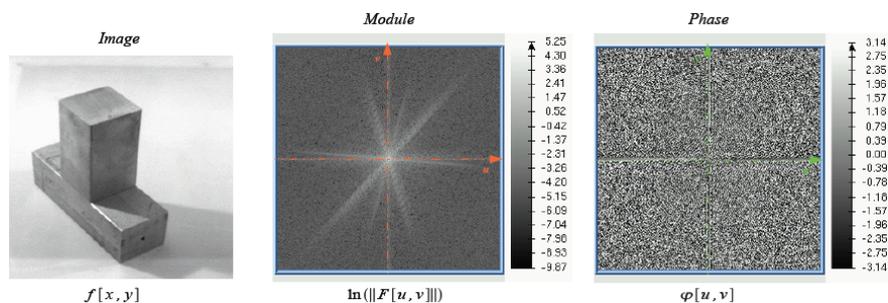
$$F[u, v] = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} f[x, y] e^{-\frac{2i\pi(ux+vy)}{wh}}$$

- TF discrète bidimensionnelle inverse

$$f[x, y] = \frac{1}{wh} \sum_{u=0}^{w-1} \sum_{v=0}^{h-1} F[u, v] e^{\frac{2i\pi(ux+vy)}{wh}}$$

1. Outils fondamentaux (Rappel)

■ La transformée de Fourier



Plan

- 1. Rappel
- 2. Problématique du filtrage
- 3. Filtres de lissage
- 4. Filtres dérivateurs
- 5. Transformée de Hough
- 6. Restauration : initiation

2. Filtrage d' image

- Le filtrage d' image a pour but :
 - D' atténuer l' effet du bruit d' acquisition sur une image.
 - Extraire des caractéristiques de l' image (contours).
 - Améliorer le contraste d' une image.

2. Filtrage d' image

- En termes de filtrage, on distinguera 2 grands types de filtre :
 - Les filtres de lissage (atténuation du bruit)
 - Les filtres dérivateurs (extraction des contours, rehaussement de contraste)

Plan

- 1. Rappel
- 2. Problématique du filtrage
- 3. **Filtres de lissage**
- 4. Filtres dérivateurs
- 5. Transformée de Hough
- 6. Restauration : initiation

3. Filtre de lissage

- On s'intéresse ici aux techniques d'*amélioration* des images numériques, pour augmenter la qualité de leur rendu visuel, ou pour faciliter leur analyse.
- On cherche donc à atténuer, sinon supprimer une certaine *dégradation*.
- On distinguera ici :
 - les dégradations liées au *bruit* : $g(x) = f(x)+b(x)$ ou $g(x) = f(x)b(x)$ liées au capteur, à la quantification, à la transmission... On les traite en tirant parti des informations locales par le *filtrage*.
 - les dégradations *convolutives* : $g(x) = f(x)*b(x)$ liées à un mouvement du capteur ou un défaut de mise au point. On les traite en inversant un opérateur linéaire, donc supposé connu : ce sont les techniques dites de *restauration*.

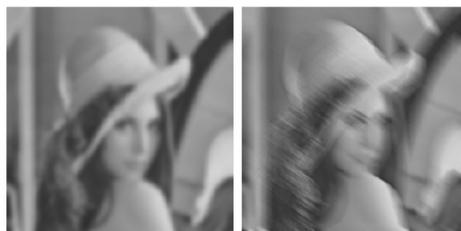
3. Filtre de lissage

- Exemples :



bruit additif

bruit multiplicatif



flou de mise au point

flou de bougé

3. Filtre de lissage

- **Les filtres de lissage** sont des opérateurs qui *éliminent* des éléments *perturbateurs / non significatifs* dans les images numériques,
 - soit pour *améliorer* leur visualisation,
 - soit pour les *simplifier* en but d'un traitement postérieur.

Bruit
gaussien



Bruit
impulsionnel

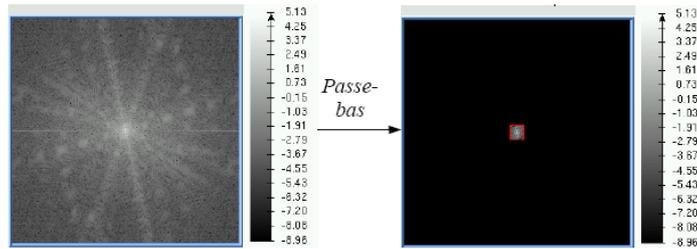
3. Filtre de lissage

- *On distingue 3 grands types de filtres de lissage*
 - (1) *Filtrage dans le domaine de Fourier*
 - (2) *Filtrage par convolution*
 - (3) *Filtres non linéaires*

3. Filtre de lissage

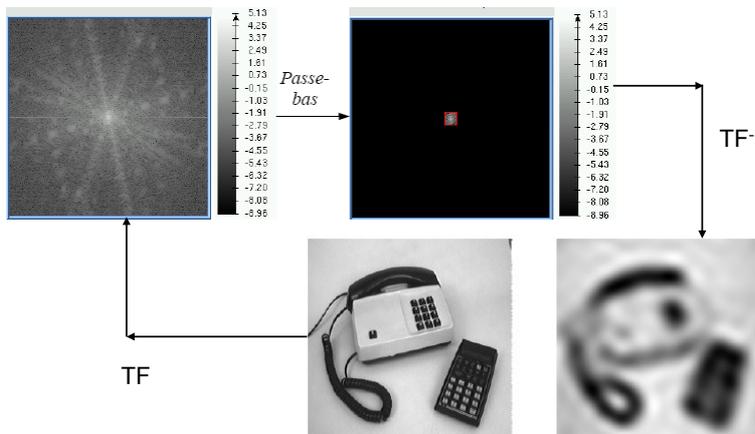
■ Filtrage passe-bas dans le domaine de Fourier

- Le filtrage passe-bas est la multiplication dans le domaine fréquentiel par une *fonction porte* (fonction indicatrice d'un intervalle $[-u_{max}, u_{max}] \times [-v_{max}, v_{max}]$).



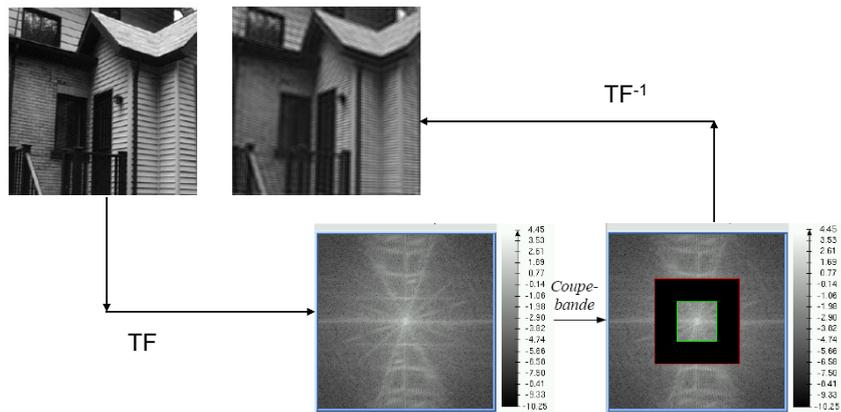
3. Filtre de lissage

■ Filtrage passe-bas dans le domaine de Fourier



3. Filtre de lissage

- Filtrage coupe-bande dans le domaine de Fourier



3. Filtre de lissage

- Filtrage par convolution

- La **multiplication** dans le domaine fréquentiel correspond à la **convolution** dans le domaine spatial.
- Un grand nombre de filtres de lissage peut être obtenu à partir de noyaux de convolution symétriques et normalisés (de somme égale à 1).

3. Filtre de lissage

- **Filtrage par convolution**

Les trois types de filtres les plus utilisés sont :

- Les filtres moyeneurs $h(x, y) = \frac{1}{\lambda^2}$
- Les filtres gaussiens $h(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$
- Les filtres exponentiels $h(x, y) = \frac{\gamma^2}{4} e^{-\gamma(|x|+|y|)}$

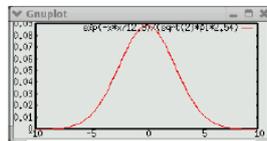
3. Filtre de lissage

- **Filtrage par convolution : réponse impulsionnelle**



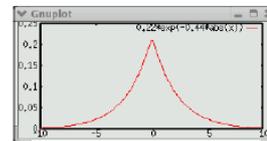
$$\frac{1}{25} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Filtre moyeneur (5x5)



$$\frac{1}{864} \begin{pmatrix} 11 & 23 & 29 & 23 & 11 \\ 23 & 48 & 62 & 48 & 23 \\ 29 & 62 & 80 & 62 & 29 \\ 23 & 48 & 62 & 48 & 23 \\ 11 & 23 & 29 & 23 & 11 \end{pmatrix}$$

Filtre gaussien ($\sigma = 1,41$)

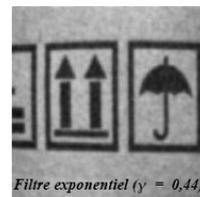
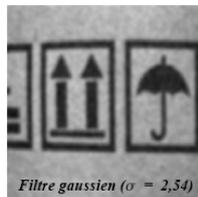
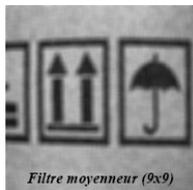
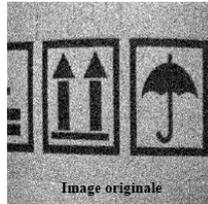


$$\frac{1}{80} \begin{pmatrix} 1 & 1 & 3 & 1 & 1 \\ 1 & 3 & 7 & 3 & 1 \\ 3 & 7 & 16 & 7 & 3 \\ 1 & 3 & 7 & 3 & 1 \\ 1 & 1 & 3 & 1 & 1 \end{pmatrix}$$

Filtre exponentiel ($\gamma = 0,8$)

3. Filtre de lissage

- Filtrage par convolution : exemples



3. Filtre de lissage

- Implantation des filtres de lissage linéaires

- En traitement d'images, les volumes de données traités sont bien sûr très importants.
- La prise en compte du temps de calcul reste un élément majeur dans les algorithmes en dépit des progrès technologiques exponentiels des microprocesseurs.
- L'implantation des filtres linéaires, en particulier ceux dont le support est grand, voire infini, est un problème incontournable.

3. Filtre de lissage

■ Implantation des filtres de lissage linéaires

- Les méthodes les plus courantes sont :
 - Multiplication dans le domaine de Fourier (La convolution devient une multiplication)
 - Convolution par noyau tronqué
 - Noyaux séparables
 - Implantation récursive des filtres RII

3. Filtre de lissage

■ Inconvénients des filtres de lissage linéaires

- Par convolution : effet de bord
- Dégradation des contours de l' image
- Inefficace sur les bruits de type impulsionnel (« salt and pepper »)

3. Filtre de lissage

■ Filtres non linéaires

- Deux aspects du lissage sont concernés par le filtrage non linéaire :
 - *Le bruit impulsionnel* : les filtres linéaires éliminent mal les valeurs aberrantes.
 - *L'intégrité des frontières* : on souhaiterait éliminer le bruit sans rendre flou les frontières des objets.

3. Filtre de lissage

■ Filtres non linéaires

- Les principaux filtres non linéaires sont :
 - Les filtres d'ordre (médian en particulier)
 - Les filtres morphologiques

3. Filtre de lissage

■ Filtres d'ordre

- Les filtres d'ordres procèdent en remplaçant les valeurs de chaque pixel par la valeur qui occupe *un certain rang* lorsqu'on trie les valeurs observées dans *un certain voisinage* du pixel.

3. Filtre de lissage

■ Filtres d'ordre : principe

- Les valeurs dans le voisinage de (x,y) :

$$V(x,y) = [a_1, a_2, \dots, a_n]$$

- Permutation des valeurs dans l'ordre croissant

$$V_p(x,y) = [b_1, b_2, \dots, b_k]$$

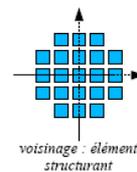
- Le filtre d'ordre de rang k est alors défini par :

$$\rho_k(x,y) = b_k$$

3. Filtre de lissage

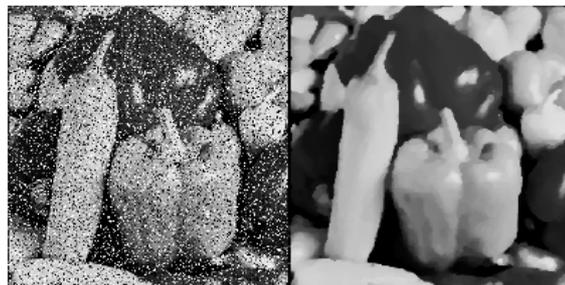
- Filtres d'ordre : exemples

- Pour $k=N/2$, on parle de filtre médian
- Pour $k=1$, on parle d'érosion morphologique
- Pour $k=N$, on parle de dilatation morphologique
- L'élément structurant de voisinage n'a pas nécessité à être carré



3. Filtre de lissage

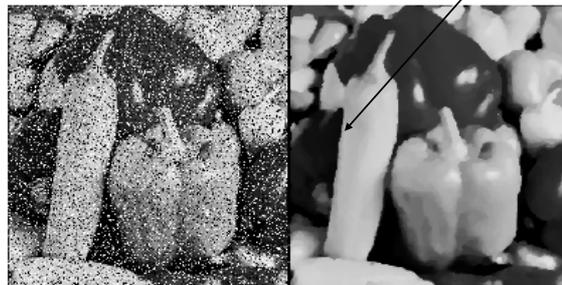
- Filtres d'ordre : filtre médian



ex : bruit impulsionnel traité par un filtre médian

3. Filtre de lissage

- Filtres d'ordre : filtre médian



Effet de bloc

ex : bruit impulsionnel traité par un filtre médian

3. Filtre de lissage

- Filtres d'ordre : érosion et dilatation



Erosion

Dilatation

3. Filtre de lissage

- **Filtres morphologiques**

- Il existe deux grands types de filtres morphologiques :
 - Le filtre d'ouverture
 - Le filtre de fermeture

3. Filtre de lissage

- **Ouverture et fermeture**

- L'*ouverture* morphologique s'obtient par la composition d'une érosion suivie d'une dilatation.
- La *fermeture* morphologique est l'opération duale de l'ouverture : Elle est égale à la composition d'une dilatation suivie d'une érosion.

3. Filtre de lissage

- Ouverture et fermeture

- Visuellement :

- L'ouverture élimine les petites composantes, et ouvre les petits isthmes.
- La fermeture bouche les petites trous, et ferme les petits détroits.

3. Filtre de lissage

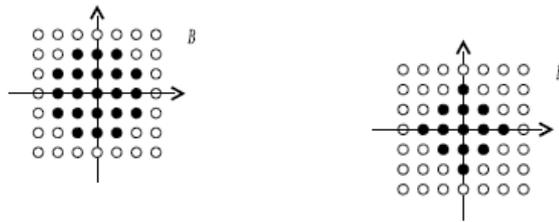
- Ouverture et fermeture ?



3. Filtre de lissage

- Ouverture et fermeture ?

- Le choix du voisinage est important



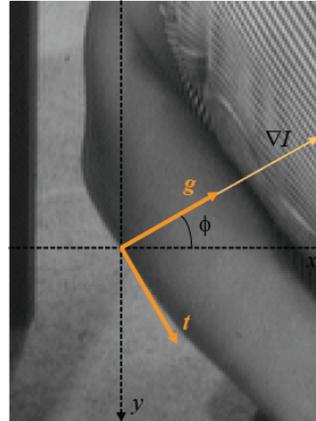
Plan

- 1. Rappel
- 2. Problématique du filtrage
- 3. Filtres de lissage
- 4. Filtres dérivateurs
- 5. Transformée de Hough
- 6. Restauration : initiation

4. Filtres dérivateurs

■ Le modèle différentiel

- Au premier ordre, on peut ainsi associer à chaque point (x,y) un repère propre (t,g) , où le vecteur t donne la direction de l'*isophote* (ligne de variation minimale) et g la direction orthogonale, celle du *gradient*.



4. Filtres dérivateurs

- Les *variations locales d'intensité* constituent une source primordiale d'information en traitement d'images.
- Elles sont mesurées par le *gradient*, fonction *vectorielle* des pixels $[i,j]$:

$$\nabla f[i, j] = \left(\frac{\partial f}{\partial x}[i, j], \frac{\partial f}{\partial y}[i, j] \right)$$

4. Filtres dérivateurs

- D'autres grandeurs différentielles sont utilisées en traitement d'images, comme le *laplacien*, fonction *scalaire* de $[i,j]$:

$$\Delta f[i,j] = \frac{\partial^2 f}{\partial x^2}[i,j] + \frac{\partial^2 f}{\partial y^2}[i,j]$$

- ou encore le *hessien*, fonction *matricielle* de $[i,j]$:

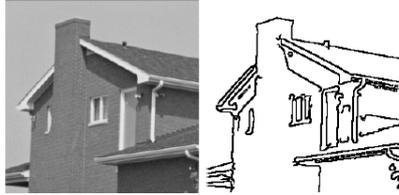
$$H_f[i,j] = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2}[i,j] & \frac{\partial^2 f}{\partial x \partial y}[i,j] \\ \frac{\partial^2 f}{\partial x \partial y}[i,j] & \frac{\partial^2 f}{\partial y^2}[i,j] \end{pmatrix}$$

4. Filtres dérivateurs

- Le problème du calcul des filtres dérivateurs dans les images numériques est l'*approximation* de ces grandeurs différentielles dans notre espace discret
- On s'intéresse aussi à leur *utilisation* dans le cadre du rehaussement, et de la détection de contours,...

4. Filtres dérivateurs

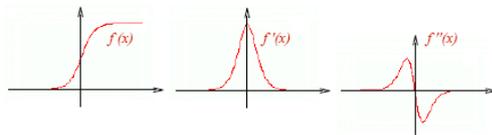
■ Exemples :



Les contours (image de droite) constituent une simplification de l'image utile dans de nombreuses applications. Dans les approches linéaires, ils sont en général obtenus à partir des maxima locaux de la dérivée première, ou des passages par zéro de la dérivée seconde :



réhaussement de contraste obtenu par combinaison linéaire avec le laplacien.



4. Filtres dérivateurs

■ Exemples :

- ❑ L'implantation de ces filtres peut se faire comme précédemment soit dans le domaine fréquentiel (filtrage passe-haut par exemple)
- ❑ Soit par convolution directe.

4. Filtres dérivateurs

■ Implantation par convolution

□ Dérivées directionnelles d'ordre 1

- Les approximations les plus simples des dérivées directionnelles se font par différences finies calculées par convolution avec des noyaux très simples :

$$\frac{\partial f}{\partial x} \longrightarrow [-1,1] \qquad \frac{\partial f}{\partial y} \longrightarrow \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

4. Filtres dérivateurs

■ Implantation par convolution

□ Dérivées directionnelles d'ordre 1

- On utilise plus souvent :

$$\frac{\partial f}{\partial x} \longrightarrow [-1,0,1] \qquad \frac{\partial f}{\partial y} \longrightarrow \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

Ces noyaux permettent en effet d'obtenir des contours plus épais mais bien centrés

4. Filtres dérivateurs

■ Implantation par convolution

- Dérivées directionnelles d'ordre 1
 - Ces opérations étant très sensibles au bruit, on les combine en général avec un filtre lisseur dans la direction orthogonale à celle de dérivation
 - par exemple par le noyau suivant (ou sa transposée) :

$$[1,2,1]$$

4. Filtres dérivateurs

■ Implantation par convolution

- Dérivées directionnelles d'ordre 1
 - Le calcul des dérivées directionnelles en x et en y revient finalement à la convolution avec les noyaux suivants :

$$\begin{aligned} f_x(i, j) &= (f * h_x)[i, j] \\ f_y(i, j) &= (f * h_y)[i, j] \end{aligned} \quad \text{avec} \quad h_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad h_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

- Ce filtre est connu sous le nom de **filtre de Sobel**

4. Filtres dérivateurs

- Implantation par convolution

- Dérivées directionnelles d'ordre 1

- On peut ensuite calculer la norme du gradient :

$$\|\nabla f(i, j)\|_2 = \sqrt{f_x(i, j)^2 + f_y(i, j)^2}$$

$$\|\nabla f(i, j)\|_1 = |f_x(i, j) + f_y(i, j)|$$

- Et son orientation : $\arg(\nabla f(i, j)) = \arctan\left(\frac{f_y(i, j)}{f_x(i, j)}\right)$

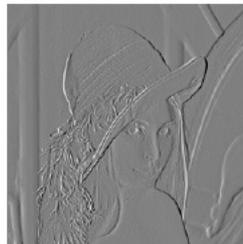
4. Filtres dérivateurs

- Implantation par convolution

- Dérivées directionnelles d'ordre 1



Original



Noyau [-1 1]



Noyau [-1 0 1]

4. Filtres dérivateurs

- Implantation par convolution

- Dérivées directionnelles d'ordre 1



Gradient horizontal (Sobel)



Gradient vertical (Sobel)



Module du gradient de Sobel

4. Filtres dérivateurs

- Implantation par convolution

- Dérivées directionnelles d'ordre 2

- On utilise le plus souvent :

$$\frac{\partial^2 f}{\partial x^2} \longrightarrow [1, -2, 1] \quad \frac{\partial^2 f}{\partial y^2} \longrightarrow \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$$

4. Filtres dérivateurs

- Implantation par convolution

- Dérivées directionnelles d'ordre 2

- Le Laplacien peut donc être estimé par le noyau suivant

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad \text{4-connexité} \quad \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad \text{8-connexité}$$

4. Filtres dérivateurs

- Implantation par convolution

- Dérivées directionnelles d'ordre 2



4. Filtres dérivateurs

■ Rehaussement de contraste

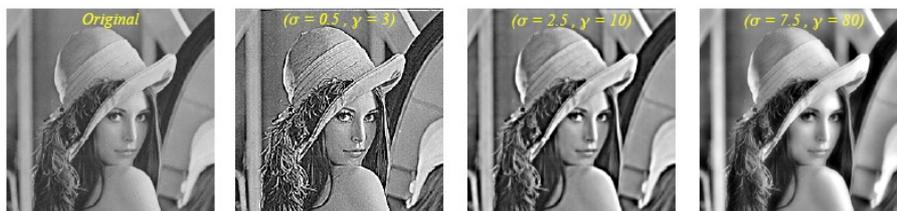
- L'opération de rehaussement de contraste à base de filtre dérivateur s'obtient de la manière suivante :

$$R_f(i, j) = f(i, j) - \gamma \cdot \Delta f(i, j)$$

- En soustrayant le Laplacien à l'image originale, on augmente le contraste.

4. Filtres dérivateurs

■ Rehaussement de contraste



4. Filtres dérivateurs

■ Filtrage optimal :

- ❑ Les dérivations présentées consistent à convoluer l'image par des masques de petites dimensions.
- ❑ Ces approches sont donc dépendantes de la taille des objets traités, elles sont aussi très sensible au bruit.
- ❑ Un autre type d'approche repose sur la définition de critères **d'optimalité de la détection de contours**; ces critères débouchant sur des filtres *optimaux*.

4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny

- ❑ On se place dans le cas monodimensionnel.
- ❑ On suppose que la détection est effectuée en convoluant le signal par un filtre de réponse impulsionnel h , les contours étant caractérisés par les extrema de la sortie du filtre.
- ❑ Les contours envisagés ici sont des contours de types marche et le bruit est supposé blanc (de moyenne nulle).

4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny

- Les critères d'optimalités sont :
 - **détection** : le contour doit être détecté, il faut minimiser les fausses réponses ;
 - **localisation** : le contour doit être localisé avec précision, il s'agit de minimiser la distance entre les points détectés et le vrai contour ;
 - **réponse unique** : il s'agit de minimiser le nombre de réponse pour un seul contour.

4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny

- La minimisation de ces critères dans le contexte du modèle proposé débouche sur une équation différentielle dont la solution est de la forme :

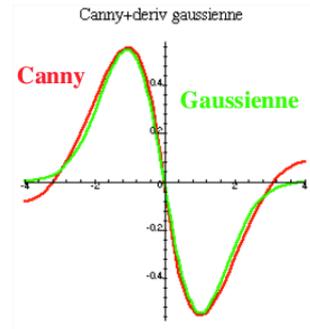
$$h(x) = a_1 e^{\alpha x} \cos(\omega x) + a_2 e^{\alpha x} \sin(\omega x) + \\ + a_3 e^{-\alpha x} \cos(\omega x) + a_4 e^{-\alpha x} \sin(\omega x).$$

4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny

- La réponse impulsionnelle h est très souvent approximée par une dérivée de Gaussienne :

$$h(x) \approx -xe^{-\frac{x^2}{2\sigma^2}}$$



4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny

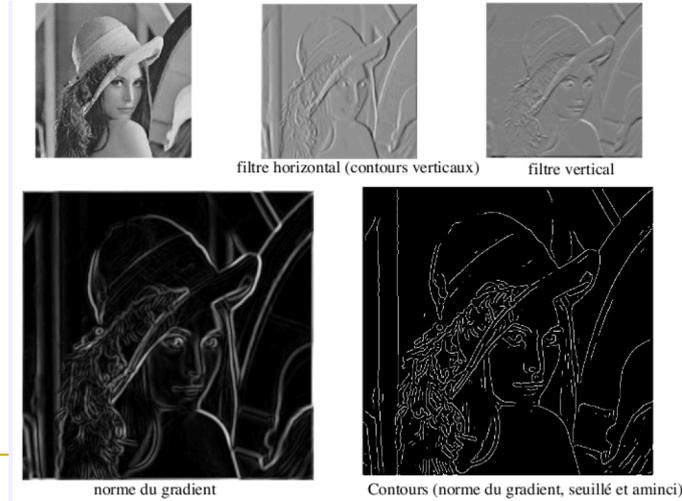
- En 2D : application de 2 filtres croisés

$$h_x(x, y) \approx -xe^{-\frac{x^2}{2\sigma^2}} e^{-\frac{y^2}{2\sigma^2}}$$

$$h_y(x, y) \approx -ye^{-\frac{x^2}{2\sigma^2}} e^{-\frac{y^2}{2\sigma^2}}$$

4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny



67

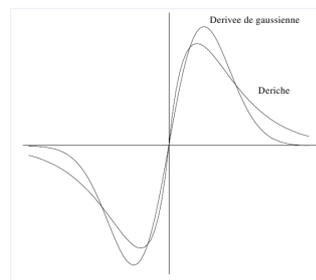
4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny-Deriche

- Modèle du filtre de Canny avec Réponse Impulsionnelle infinie (RII)

$$h(x) = -ce^{-\alpha|x|} \sin(\omega x)$$

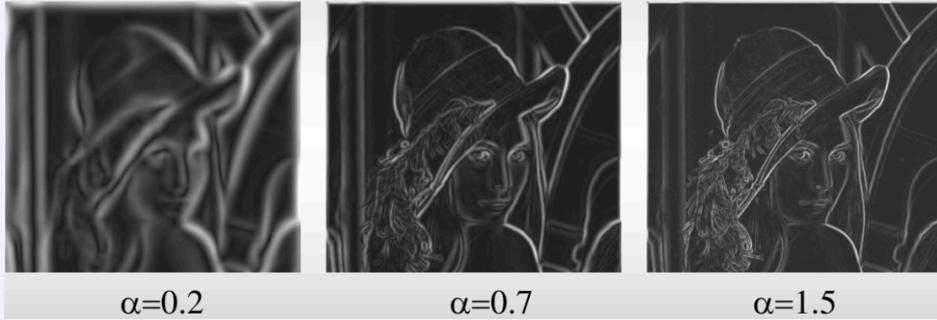
$$\text{avec } c = \frac{(1 - e^{-\alpha^2})}{e^{-\alpha}} \quad (\text{normalisation})$$



- Implantation récursive simple en 2D

4. Filtres dérivateurs

■ Filtrage optimal : critère de Canny-Deriche



- α joue le rôle d'un facteur d'échelle
- α faible : bonne localisation mais peu de robustesse au bruit
- α élevé : bonne robustesse mais mauvaise localisation

4. Filtres dérivateurs

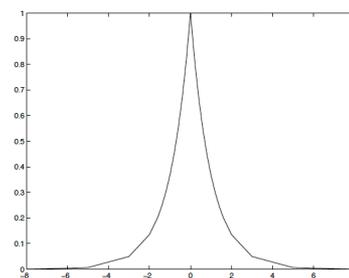
■ Filtrage optimal : autres critères

- Shen-Castan

$$h(x) = c \cdot \text{sign}(x) e^{-\alpha|x|}$$

- Spacek, Petrou

- ...



α joue sur la
largeur du filtre

Plan

- 1. Rappel
- 2. Problématique du filtrage
- 3. Filtres de lissage
- 4. Filtres dérivateurs
- 5. Transformée de Hough
- 6. Restauration initiation

5. Transformée de Hough

5.1 Introduction

- La transformée de Hough a été développée par Paul Hough en 1962, et a été brevetée par IBM.
- Dans les dernières décennies, la transformée de Hough est devenu un outils standard dans le domaine de la vision artificielle.
- Elle permet la détection de droites, de cercles ou d'ellipses de façon traditionnelle .
- Elle peut aussi être étendue a des cas de description d'objets plus complexes.

5. Transformée de Hough

5.2 Principe (détection de droite)

- Une droite est caractérisée par deux paramètres a et b tels que $y = ax+b$.
- La transformée de Hough va permettre de faire un mapping entre les coordonnées d'un pixel et l'espace des paramètres (a,b) .
- Pour cela deux tactiques possibles. Soit utiliser la transformation de 1 à m , soit de m à 1.

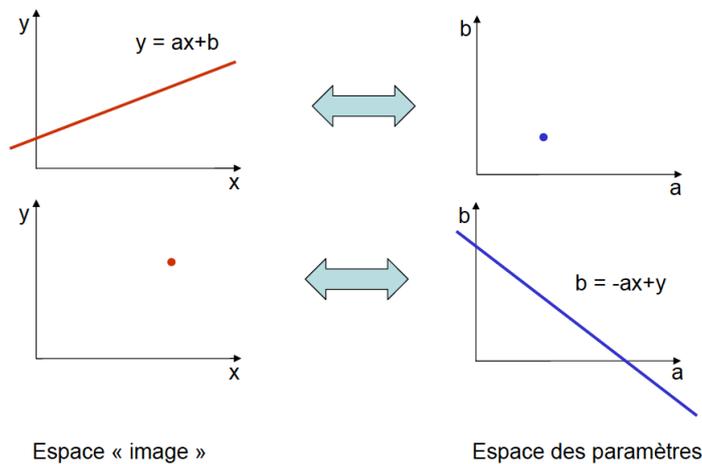
5. Transformée de Hough

5.2 Principe (détection de droite)

- En effet on peut lire l'équation d'une droite de deux manières :
 - Soit par la lecture traditionnelle qui nous indique que $y=ax+b$: tous les points (x,y) vérifiant cette équation seront associés au même couple (a,b) dans l'espace des paramètres (En l'occurrence, il nous faudra simplement un bipoint pour définir complètement notre droite).
 - Soit on peut lire $b = -ax + y$, alors un couple (x,y) de l'image sera projeté en une droite de l'espace de paramètres (a,b) .

5. Transformée de Hough

5.2 Principe (détection de droite)



Aymeric Histace

75

5. Transformée de Hough

5.2 Principe (détection de droite)

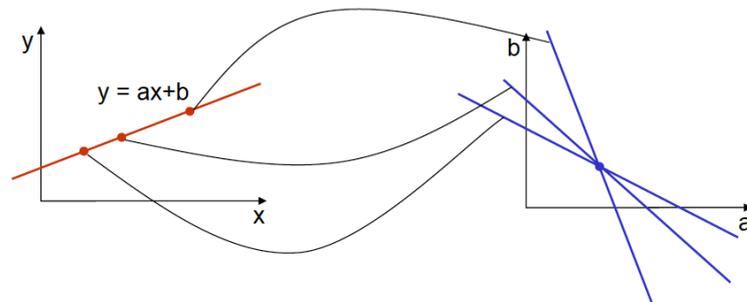
- Ensuite pour étudier l'existence de droites il faut se pencher sur l'espace des paramètres.
- Celui-ci, si tout c'est bien passé, doit présenter des regroupements traduisant l'existence de l'objet recherché.
- Une étude des différents maximum doit être alors effectuée. Il faut savoir à partir de quel moment un regroupement de point est significatif.

Aymeric Histace

76

5. Transformée de Hough

5.2 Principe (détection de droite)



Les images de tous les points d'une droite se coupent en (a,b) dans le domaine de Hough

5. Transformée de Hough

5.2 Principe (détection de droite)

- Cette étape est en général délicate, le bruit causé par les autres éléments de l'image peuvent perturber la détection des objets recherchés.
- D'autre part si les objets sont trop nombreux on peut aussi avoir du mal à les repérer.

5. Transformée de Hough

5.2 En pratique (détection de droite)

- 1. Appliquer une détection de contours à l'image
- 2. Discrétiser le plan des paramètres (a,b)
- 3. Initialiser un accumulateur
- 4. Pour chaque point sur un contour
 - 1. Déterminer sa droite image dans l'espace des paramètres
 - 2. Incrémenter l'accumulateur sur les points de cette droite
- 5. Recherche des maxima, i.e. paramètres des droites présentes.

5. Transformée de Hough

5.2 En pratique (détection de droite)

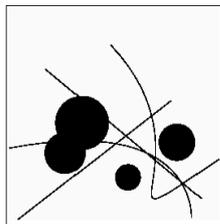
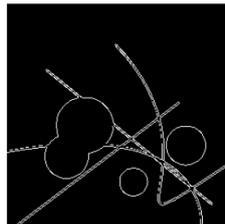


Image originale



Détection de contours
(MATLAB: edge)



Transformée de Hough
Recherche de maximum



Le max correspond aux paramètres (a,b) de la droite

5. Transformée de Hough

5.2 En pratique (détection de droite)

■ Défauts :

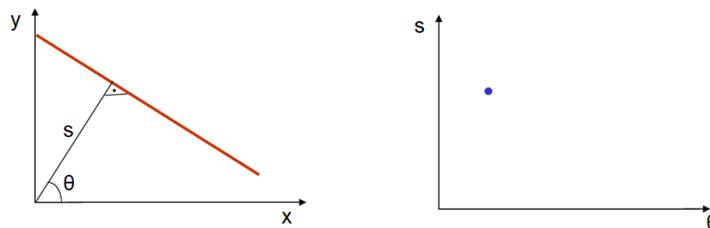
- L'espace des paramètres doit être borné et discrétisé pour une implantation réelle
- Une droite verticale ne peut pas être représentée (a tend vers l'infini)

■ Solution :

- Utiliser un autre couple de paramètres que (a,b)

5. Transformée de Hough

5.2 L'espace (s,θ) (détection de droite)



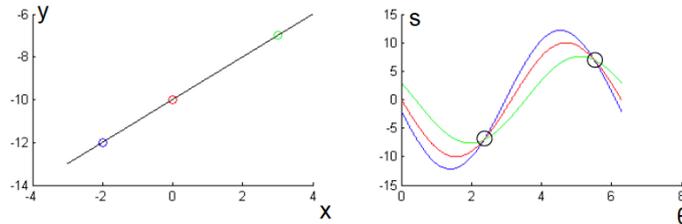
$$x = s \cos(\theta)$$

$$y = s \sin(\theta)$$

$$x \cos(\theta) + y \sin(\theta) - s = 0$$

5. Transformée de Hough

5.2 L'espace (s, θ)



$$(x_1, y_1) \longrightarrow s = x_1 \cos(\theta) + y_1 \sin(\theta)$$

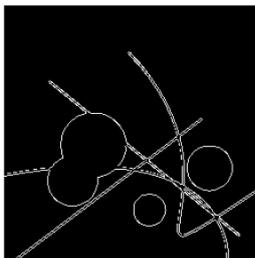
Les sinusôïdes correspondant aux points d'une même droite se coupent au point (s, θ) paramétrisant cette droite

Aymeric Histace

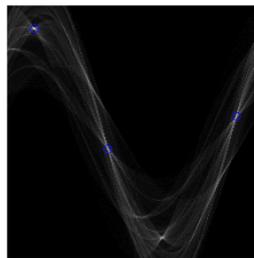
83

5. Transformée de Hough

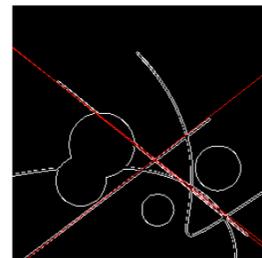
5.2 L'espace (s, θ) (droite)



Détection de bords



Transformée de Hough (s, θ)
Maxima locaux



Droites correspondantes

Aymeric Histace

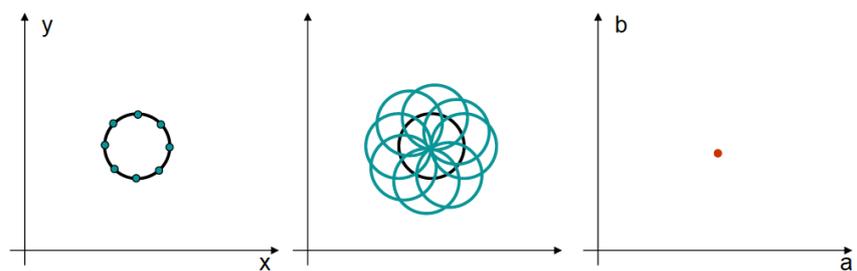
84

5. Transformée de Hough

5.3 Détection de cercle

La paramétrisation d'un cercle de centre (a,b) et de rayon R peut être donnée par

$$(x-a)^2 + (y-b)^2 - R^2 = 0$$

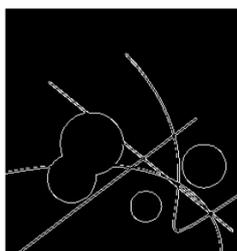


Aymeric Histace

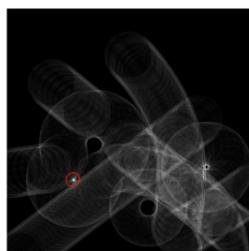
85

5. Transformée de Hough

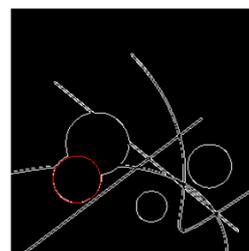
5.3 Détection de cercle



Détection de bords



Transformée de Hough (a,b)
Maxima locaux



Cercle
correspondant

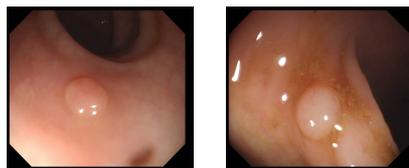
Aymeric Histace

86

5. Transformée de Hough

5.3 Détection de cercle :

Application à la détection de polype en vidéocoloscopie (dépistage du cancer colorectal)



Exemples d'images acquises

Aymeric Histace

87

5. Transformée de Hough

5.3 Détection de cercle :

Application à la détection de polype en vidéocoloscopie (dépistage du cancer colorectal)



Hough



Détection

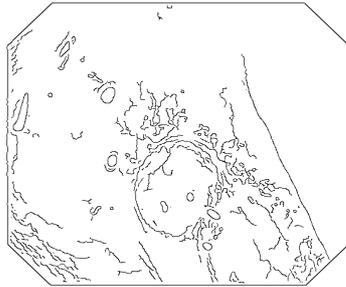
Aymeric Histace

88

5. Transformée de Hough

5.3 Détection de cercle :

Application à la détection de polype en vidéocoloscopie (dépistage du cancer colorectal)



Hough



Détection

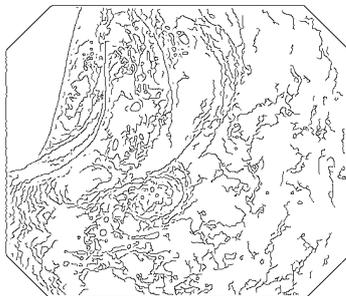
Aymeric Histace

89

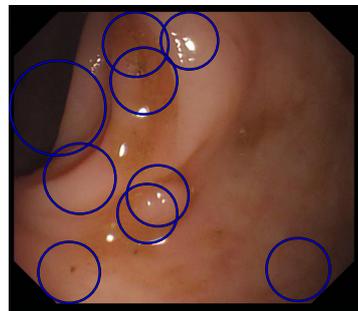
5. Transformée de Hough

5.3 Détection de cercle :

Application à la détection de polype en vidéocoloscopie (dépistage du cancer colorectal)



Hough



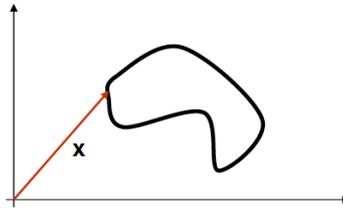
Détection

Aymeric Histace

90

5. Transformée de Hough

5.4 Généralisation



Paramétrisation: $f(\mathbf{x}, \mathbf{a}) = 0 \quad \forall \mathbf{x} \in \text{la courbe}$

La dimension de l'accumulateur (espace des paramètres)
est $\dim(\mathbf{a})$

5. Transformée de Hough

5.4 Généralisation

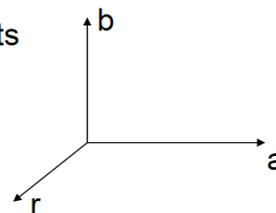
La dimension du domaine transformé est égal aux
nombres de paramètres nécessaires pour décrire la
courbe

=> Le temps de calcul et la mémoire utilisée
deviennent vite conséquents

Exemple: Cercles de rayons différents

$$f(\mathbf{x}, \mathbf{a}) = (x-a)^2 + (y-b)^2 - r^2$$

$$\mathbf{x} = (x, y) \quad \mathbf{a} = (a, b, r)$$



Plan

- 1. Rappel
- 2. Problématique du filtrage
- 3. Filtres de lissage
- 4. Filtres dérivateurs
- 5. Transformée de Hough
- 6. **Restauration initiation**

6. Restauration : initiation

- **Problématique :**
 - On se place dans le cadre où l' image est dégradée de manière *convolutive* : $i(x,y) = b(x,y)*h(x,y)$.
 - Ces dégradation sont liées à un mouvement du capteur ou un défaut de mise au point.
 - On les traite en inversant un opérateur linéaire, donc supposé connu.

6. Restauration : initiation

■ Problématique :

- Dans le domaine spectrale : $I(u,v)=B(u,v).H(u,v)$
- On cherche donc une fonction de transfert $H'(u,v)$ telle que :

$$o(x, y) = i(x, y) * h'(x, y)$$

- Avec $o(x, y) \approx b(x, y)$
- Soit $O(u, v) = I(u, v).H'(u, v) = B(u, v).H(u, v).H'(u, v)$

6. Restauration : initiation

■ Problématique :

- On en conclut donc que $H(u,v).H'(u,v)=1$
- Et que $h'(x,y)$ est telle que :

$$H'(u, v) = \frac{1}{H(u, v)}$$

- **Problèmes :** Solution instable si $H(u,v)=0$

6. Restauration : initiation

■ Solution n°1

- On construit $H'(u,v)$ de la manière suivante :

$$H'(u,v) = \frac{1}{H(u,v) + C}$$

- Avec C une constante positive telle que $H(u,v) \gg C$
- Si $H(u,v)$ tend vers 0 alors $H'(u,v)$ tend vers $1/C$ (saturation)

6. Restauration : initiation

■ Solution n°1

- Résultat



- **Problème** : Artefacts HF très visibles (amplification en $1/C$)

6. Restauration : initiation

■ Solution n°2

- On construit $H'(u,v)$ de la manière suivante :

$$H'(u,v) = \frac{H(u,v)}{H^2(u,v) + C}$$

- Avec C une constante positive telle que $H^2(u,v) \gg C$
- Si $H(u,v)$ tend vers 0 alors $H'(u,v)$ tend vers 0

6. Restauration : initiation

■ Solution n°2

- Résultat



- Proche du filtrage optimal

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- On construit $H'(u,v)$ de façon à ce que :

$$H'(u,v) = \frac{H(u,v)}{H^2(u,v) + \frac{\Phi_{nn}}{\Phi_{bb}}}$$

- $\Phi_{ab}(u,v)$ étant la TF de la corrélation entre $a(x,y)$ et $b(x,y)$
- $n(x,y)$ représentant un bruit additif détériorant l' image

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Rappel : fonction de corrélation entre $a(x,y)$ et $b(x,y)$

$$\phi_{ab}(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} a(\xi - x, \eta - y) b(\xi, \eta) d\xi d\eta$$

- $\Phi_{aa}(x,y)$: autocorrélation

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- **Construction de $H'(u,v)$** : on cherche à minimiser une fonction énergétique caractérisant la qualité de restauration :

$$E = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (o(x,y) - d(x,y))^2 dx dy$$

- Avec $d(x,y)$ l' image désirée et $o(x,y) = i(x,y) * h'(x,y)$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Ainsi :

$$E = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (o^2(x,y) - 2o(x,y)d(x,y) + d^2(x,y)) dx dy$$

- Or par définition

$$o^2(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} i(x-\xi, y-\eta) h'(\xi, \eta) d\xi d\eta \times \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} i(x-\alpha, y-\beta) h'(\alpha, \beta) d\alpha d\beta$$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

□ Et donc :

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} o^2(x, y) dx dy = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi_{ii}(\xi - \alpha, \eta - \beta) h'(\xi, \eta) h'(\alpha, \beta) d\xi d\eta d\alpha d\beta$$

□ De plus

$$o(x, y) d(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} i(x - \xi, y - \eta) h'(\xi, \eta) d(x, y) d\xi d\eta$$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

□ Donc :

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} o(x, y) d(x, y) dx dy = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi_{id}(\xi, \eta) h'(\xi, \eta) d\xi d\eta$$

□ Or

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} d^2(x, y) dx dy = \phi_{dd}(0, 0)$$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Enfin :

$$E = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi_{ii}(\xi - \alpha, \eta - \beta) h'(\xi, \eta) h'(\alpha, \beta) d\xi d\eta d\alpha d\beta - 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi_{id}(\xi, \eta) h'(\xi, \eta) d\xi d\eta + \phi_{dd}(0, 0)$$

- Il s'agit du critère à minimiser (voir cours approche variationnel pour déterminer $h'(x, y)$)

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Solution (Euler-Lagrange):

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi_{ii}(\xi - \alpha, \eta - \beta) h'(\alpha, \beta) d\alpha d\beta - \phi_{id}(\xi, \eta) = 0$$

- Soit :

$$\phi_{id} = \phi_{ii} * h' \quad \text{ou} \quad \boxed{\Phi_{id} = H' \Phi_{ii}} \quad \text{Domaine fréquentielle}$$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Solution : Cas du bruit additif

$$i(x, y) = b(x, y) + n(x, y). \quad \text{Avec } d(x, y) = b(x, y)$$

- Donc

$$\Phi_{id} = \Phi_{bb} + \Phi_{nb}$$

$$\Phi_{ii} = \Phi_{bb} + \Phi_{bn} + \Phi_{nb} + \Phi_{nn}$$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Hypothèse : bruit non corrélé aux données

$$H' = \frac{\Phi_{id}}{\Phi_{ii}} = \frac{\Phi_{bb}}{\Phi_{bb} + \Phi_{nn}} = \frac{1}{1 + \Phi_{nn}/\Phi_{bb}}.$$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Solution : Cas de détériorations convolutive et additive

$$i = b * h + n \quad \text{Avec } d(x,y) = b(x,y)$$

- Donc

$$\Phi_{id} = H\Phi_{bb} + \Phi_{nb}$$

$$\Phi_{ii} = H^2\Phi_{bb} + H(\Phi_{nb} + \Phi_{bn}) + \Phi_{nn}$$

6. Restauration : initiation

■ Filtrage optimal : Filtre de Wiener

- Même hypothèse sur le bruit

$$H' = \frac{\Phi_{id}}{\Phi_{ii}} = \frac{H\Phi_{bb}}{H^2\Phi_{bb} + \Phi_{nn}}$$

6. Restauration : initiation

- **Filtrage optimal : Filtre de Wiener**

- Résultat (flou)



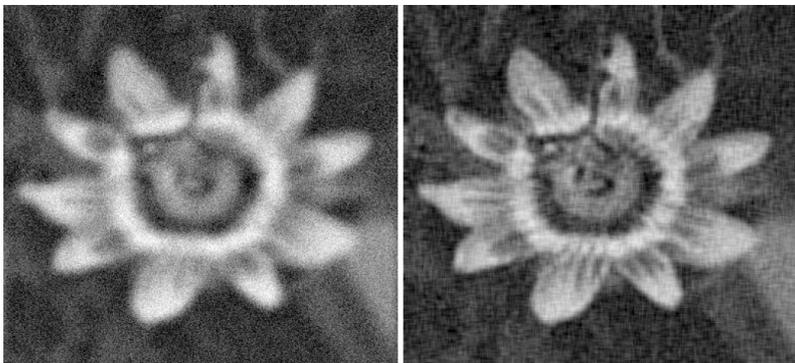
Aymeric Histace

113

6. Restauration : initiation

- **Filtrage optimal : Filtre de Wiener**

- Résultat (flou+bruit)



Aymeric Histace

114

Bibliographie

- J.P. Cocquerez et S. Philipp « *Analyse d'images : filtrage et segmentation* » Masson 1995.
- R.C. Gonzalez et Woods « *Digital Image Processing - 2d edition* » Addison Wesley 2002.
- A. Rosenfeld et A.C. Kak « *Digital picture processing* » Academic Press - London 1982.
- H. Maître (ss la direction de) « *Le traitement des images* » Hermes Lavoisier IC2 2003.
- J.R. Parker « *Algorithms for Image Processing and Computer Vision* » Wiley & Sons 1997.
- S. Bres, J.M. Jolion, F. Lebourgeois « *Traitement et analyse des images numériques* » Hermes Lavoisier 2003
- I.T. Young, J.J. Gerbrands et L.J. Van Vliet « *Fundamentals of Image Processing* » Université de Delft.
<http://www.ph.tn.tudelft.nl/~lucas/publications/1995/FIP95TYJGLV/FIP2.2.pdf>
- D. Lingrand « *Introduction au Traitement d'images* » Vuibert 2004.