

Introduction à l'imagerie numérique

3I022-fev2018

Méthodes de base en traitement d'images

Licence d'informatique



Février 2018

Méthodes de base de traitement des images numériques

- ▶ Opérations entre images
- ▶ Opérations géométriques
- ▶ Opérations sur les pixels
- ▶ Filtrage spatial et amélioration d'images

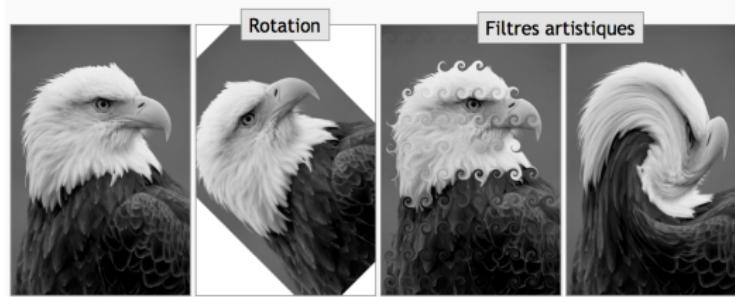
Opérations entre images

- ▶ Opérations arithmétiques simples (images de même taille) : somme, différence, division, multiplication, moyenne, opérations logiques, ...
- ▶ Opérations non linéaires : max, min, ...
- ▶ Opération de normalisation souvent nécessaire



Opérations géométriques

- ▶ Modifient les relations spatiales (positions) entre les pixels
- ▶ Indépendantes de la valeur du pixel
- ▶ Formulation générale d'une transformation : $X' = M \times X$, où X est un vecteur contenant les coordonnées (x, y) d'un pixel et M une matrice de transformation
- ▶ Exemples :
 - ▶ Translation et rotation : voir `man rot` ; changement d'échelle : voir `man izoom`; ...
 - ▶ Application courante : *morphing*



Opérations ponctuelles sur les valeurs des pixels

- ▶ Modifient les valeurs (niveaux de gris) des pixels
- ▶ Peuvent se faire à partir de l'histogramme
- ▶ Formulation générale d'une transformation linéaire :
$$I'(x, y) = a \times I(x, y) + b$$
- ▶ $I'(x, y)$ doit être un entier et compris entre 0 et $L - 1$
- ▶ La valeur a modifie la dynamique des pixels
- ▶ La valeur b modifie la luminosité

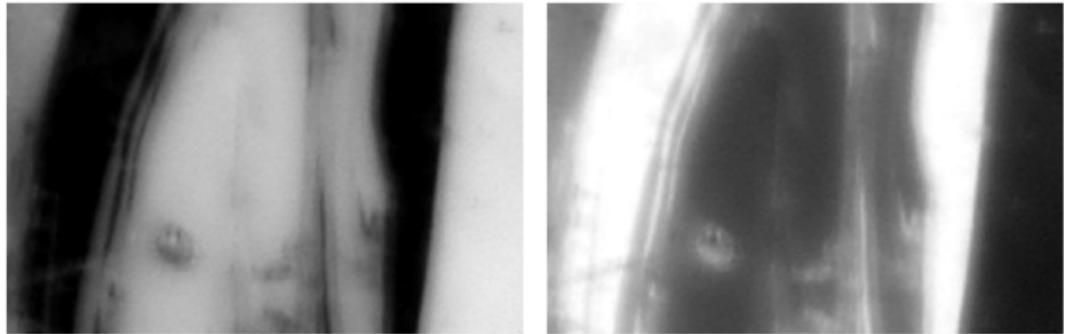
Exemple ($a = 2$ et $L = 256$ niveaux de gris)

- ▶ $I(x, y) = 10 \rightarrow I'(x, y) = 20$ (*toujours noir*)
- ▶ $I(x, y) = 127 \rightarrow I'(x, y) = 254$ (*gris moyen devient blanc*)

```
|| sc -n a image | bi -n b > output
```

Opérations directes sur les valeurs des pixels : négation

- ▶ Image en négatif : inversement de l'échelle des niveaux de gris de 0 à ($L-1$)
- ▶ $I'(x, y) = L - I(x, y)$



```
|| logic -inv image > output
```

Opérations directes sur les valeurs des pixels : seuillage

- ▶ Seuillage (*thresholding*) : traitement ramenant l'image à deux ou quelques niveaux d'intensité
- ▶ Binarisation : traitement ramenant l'image à deux niveaux, on l'appelle aussi seuillage binaire

```
for x = 0 à M - 1 do
    for y = 0 à N - 1 do
        if I(x, y) < S then
            I(x, y) = A
        else
            I(x, y) = B
        end if
    end for
end for
```



- ▶ Commandes de seuillage, voir : `man arith`.

Méthodes d'amélioration d'images (*enhancement*)

- ▶ But : rendre les images plus aptes à l'interprétation humaine ou à celle de la machine
 - ▶ Vision subjective
 - ▶ Aucune théorie générale : spécificité de l'image
 - ▶ On parle de prétraitement d'images
- ▶ Méthodes agissant sur les niveaux de gris
 - ▶ Rehaussement de contraste
 - ▶ Modifications d'histogramme
- ▶ Méthodes de voisinage pour atténuer le bruit
 - ▶ Filtrage spatial (filtre moyenneur, filtre median)
 - ▶ Filtrage fréquentiel
- ▶ Méthodes de voisinage pour mettre en évidence les contours
- ▶ Ici $k = I(x, y)$ est la valeur du pixel de coordonnées (x, y)

Méthodes agissant sur les niveaux de gris

- ▶ Image trop claire ou trop foncée
- ▶ Nécessité de modifier ses niveaux de gris afin de rendre visibles certains détails
- ▶ Méthodes générales : s'appliquent à tous les pixels (possibilité de les appliquer sur des zones de l'image)



Normalisation d'images

- ▶ On part d'un intervalle de variation des niveaux de gris $[k_{\min}, k_{\max}]$ (k_{\min} peut être négatif, et $k_{\max} > L - 1$)
- ▶ On doit arriver à un intervalle $[A, B]$
- ▶ Soit k un niveau de gris de $[k_{\min}, k_{\max}]$ et k' le niveau de gris correspondant dans $[A, B]$, alors on a :

$$k' = (k - k_{\min}) \times \frac{B - A}{k_{\max} - k_{\min}} + A$$

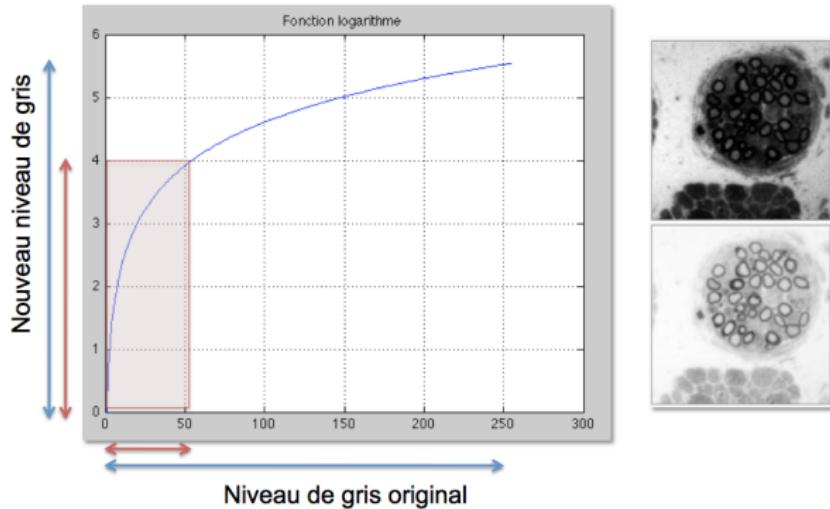
- ▶ On va revenir dessus plus tard

Exemple ($k_{\min} = 25$, $k_{\max} = 50$, $A = 10$, $B = 255$)

- ▶ $k = 25 \rightarrow k' = 10$
- ▶ $k = 50 \rightarrow k' = 255$
- ▶ $k = 45 \rightarrow k' = 206$

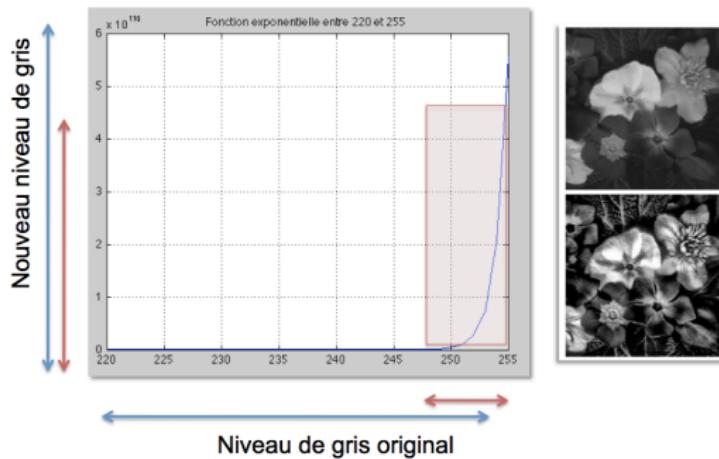
Rehaussement de contraste logarithmique

- ▶ $k' = \log(k)$
- ▶ L'intervalle des intensités sombres est augmenté : éclaircissement global de l'image
- ▶ Utilisé pour traiter des images trop sombres



Rehaussement de contraste exponentiel

- ▶ $k' = e^k$
- ▶ L'intervalle des intensités claires est augmenté : assombrissement global de l'image
- ▶ Utilisé pour traiter des images trop claires



Problème avec les rehaussements logarithmique et exponentiel

- ▶ Les valeurs des niveaux de gris transformés n'appartiennent pas à l'intervalle voulu (dynamique L)
- ▶ Nécessité d'effectuer une transformation sinon l'image sera illisible
- ▶ Pour le rehaussement de contraste logarithmique :

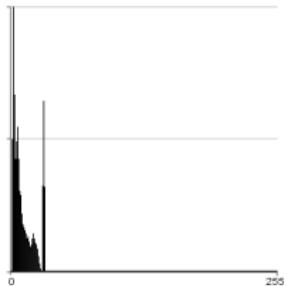
$$k' = \frac{\log(k)}{\log(L-1)} \times (L-1), k \neq 0$$

- ▶ Pour le rehaussement de contraste exponentiel :

$$k' = \frac{e^k}{e^{L-1}} \times (L-1)$$

Modifications d'histogramme : étirement

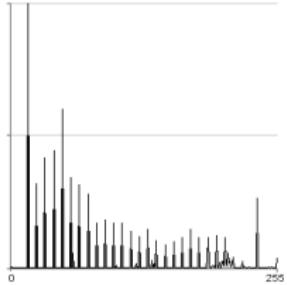
- ▶ Élargissement de l'intervalle de répartition des intensités des pixels
- ▶ Meilleure répartition des valeurs de niveaux de gris
- ▶ Transformation linéaire : $[a, b] \rightarrow [A, B]$
(peut se faire dans n'importe quel autre intervalle)



$$k'_e = \frac{L-1}{b-a}(k - a)$$

- ▶ Rappel, d'une manière générale :

$$k'_e = (k - k_{\min}) \times \frac{B - A}{k_{\max} - k_{\min}} + A$$



Exemple d'étirement d'histogramme

Exemple (pour $A = 0$, $B = 10$, $k_{\min} = 0$ et $k_{\max} = 5$)

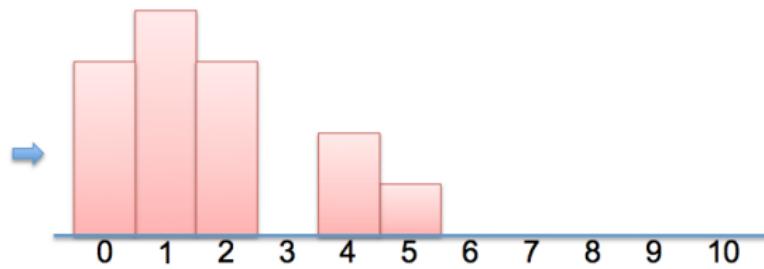
- ▶ Histogramme :

5	6	5	0	3	1
---	---	---	---	---	---
- ▶ $k = 0 : k'_e = (k - k_{\min}) \times \frac{B-A}{k_{\max}-k_{\min}} + A = \frac{10-0}{5-0}(0 - 0) + 0 = 0$
- ▶ $k = 1 : k'_e = (k - k_{\min}) \times \frac{B-A}{k_{\max}-k_{\min}} + A = \frac{10-0}{5-0}(1 - 0) + 0 = 2$
- ▶ $k = 2 : k'_e = (k - k_{\min}) \times \frac{B-A}{k_{\max}-k_{\min}} + A = \frac{10-0}{5-0}(2 - 0) + 0 = 4$
- ▶ $k = 3 : k'_e = (k - k_{\min}) \times \frac{B-A}{k_{\max}-k_{\min}} + A = \frac{10-0}{5-0}(3 - 0) + 0 = 6$
- ▶ $k = 4 : k'_e = (k - k_{\min}) \times \frac{B-A}{k_{\max}-k_{\min}} + A = \frac{10-0}{5-0}(4 - 0) + 0 = 8$
- ▶ $k = 5 : k'_e = (k - k_{\min}) \times \frac{B-A}{k_{\max}-k_{\min}} + A = \frac{10-0}{5-0}(5 - 0) + 0 = 10$

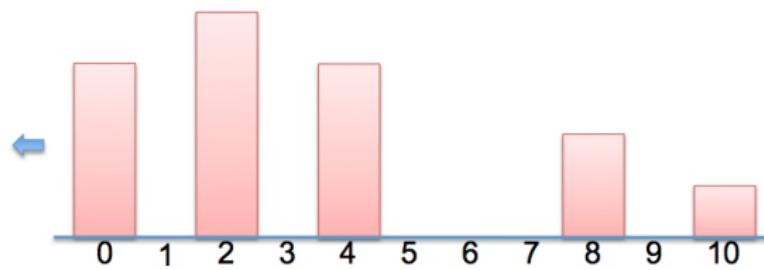
Exemple d'étirement d'histogramme

Exemple (pour $A = 0$, $B = 10$, $k_{\min} = 0$ et $k_{\max} = 5$)

0	1	1	2	4
2	1	0	0	2
5	2	0	0	4
1	1	2	4	1



0	2	2	4	8
4	2	0	0	4
10	4	0	0	8
2	2	4	8	2



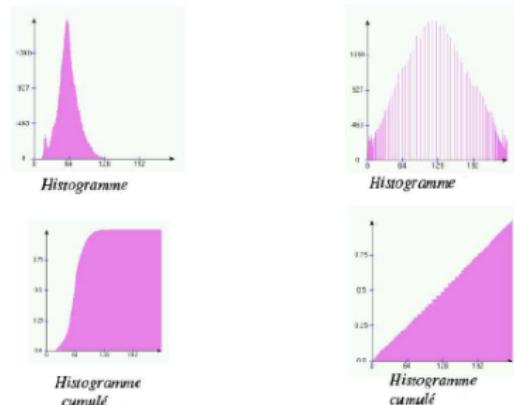
Modifications d'histogramme : égalisation

- ▶ Homogénéisation de la répartition des intensités des pixels, en linéarisant l'histogramme cumulé

$$k'_{eg} = \text{Int} \left(\frac{L - 1}{N \times M} H_c(k) \right)$$

où Int est l'entier le plus proche, NM la taille de l'image et L sa dynamique.
 $H_c(k)$ est l'histogramme cumulé en k

- ▶ Amplification des fluctuations dans les zones où elles sont faibles
- ▶ Étalement des détails concentrés dans un petit intervalle de niveaux de gris

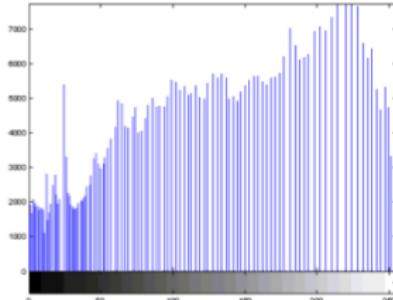
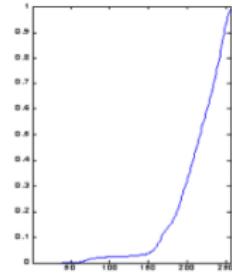
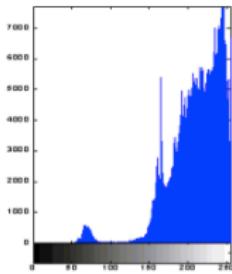


Égalisation d'histogramme

Avant égalisation



Après égalisation



Exemple d'égalisation d'histogramme

Exemple (pour $A = 0$, $B = 10$, $k_{\min} = 0$ et $k_{\max} = 5$)

- ▶ Histogramme :

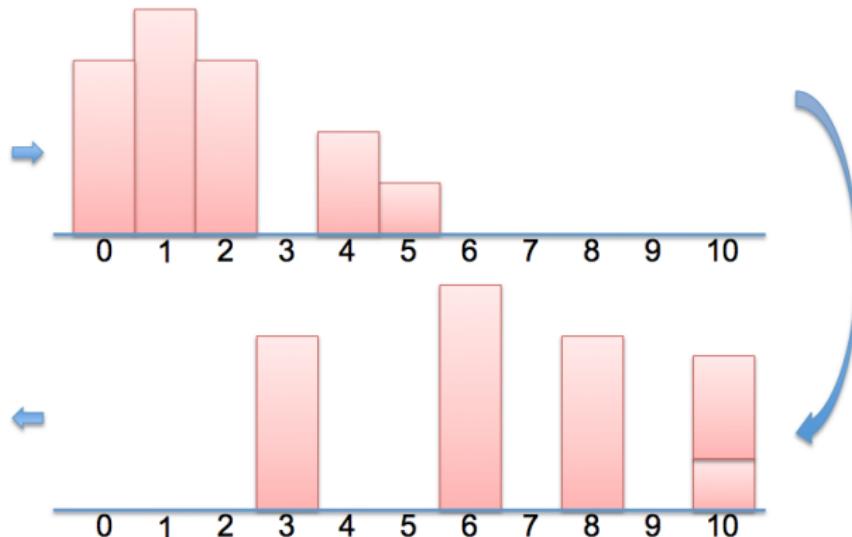
5	6	5	0	3	1
---	---	---	---	---	---
- ▶ Histogramme cumulé :

5	11	16	16	19	20
---	----	----	----	----	----
- ▶ $k = 0 : k'_{eg} = \text{Int} \left(\frac{L-1}{N \times M} H_c(k) \right) = \text{Int} \left(\frac{10}{20} \times 5 \right) \approx 3$
- ▶ $k = 1 : k'_{eg} = \text{Int} \left(\frac{L-1}{N \times M} H_c(k) \right) = \text{Int} \left(\frac{10}{20} \times 11 \right) \approx 6$
- ▶ $k = 2 : k'_{eg} = \text{Int} \left(\frac{L-1}{N \times M} H_c(k) \right) = \text{Int} \left(\frac{10}{20} \times 16 \right) = 8$
- ▶ $k = 3 : k'_{eg} = \text{Int} \left(\frac{L-1}{N \times M} H_c(k) \right) = \text{Int} \left(\frac{10}{20} \times 16 \right) = 8$
- ▶ $k = 4 : k'_{eg} = \text{Int} \left(\frac{L-1}{N \times M} H_c(k) \right) = \text{Int} \left(\frac{10}{20} \times 19 \right) \approx 10$
- ▶ $k = 5 : k'_{eg} = \text{Int} \left(\frac{L-1}{N \times M} H_c(k) \right) = \text{Int} \left(\frac{10}{20} \times 20 \right) = 10$

Exemple d'égalisation d'histogramme

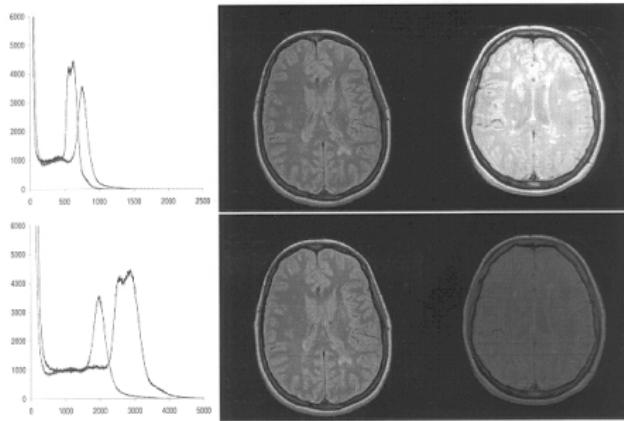
Exemple (pour $A = 0$, $B = 10$, $k_{\min} = 0$ et $k_{\max} = 5$)

0	1	1	2	4
2	1	0	0	2
5	2	0	0	4
1	1	2	4	1



Modifications d'histogramme : spécification

- ▶ Donner à un histogramme une forme particulière (spécifier la répartition des niveaux de gris)
- ▶ Favoriser certaines plages d'intensité dans lesquelles on cherche des détails
- ▶ Faire en sorte qu'un ensemble d'images aient les mêmes (ou presque) caractéristiques statistiques



Dans Inrimage

- ▶ Normalisation : commande `norma`,
- ▶ Rehaussement :
 - ▶ logarithmique : commande `lo`,
 - ▶ exponentiel : commande `exp`,
- ▶ Calcul d'histogramme : commande `his`. Exemple :

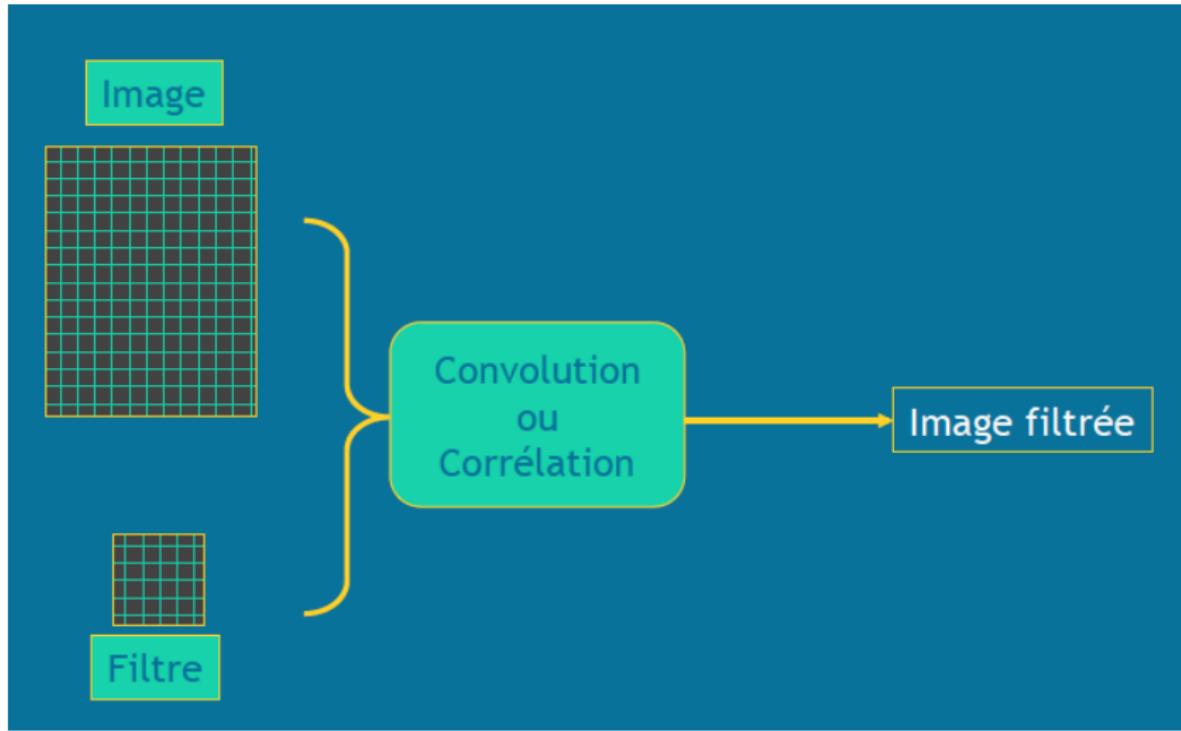
```
|| his lena.inr | par
|| < -F=Inrimage -hdr=1 -x 256 -y 1 -rdecm -o 4
|| his lena.inr | tpr -l 1 -c | nl | xgraph
```

- ▶ Égalisation d'histogramme : commande `eg`.

Filtrage spatial d'images

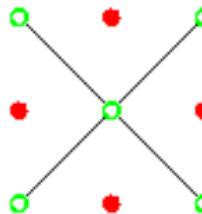
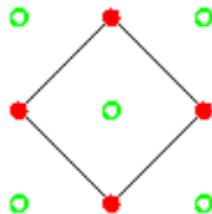
- ▶ Définitions
- ▶ Filtrage par convolution
- ▶ Filtrage par corrélation
- ▶ Types de filtres
- ▶ Élimination du bruit dans les images par filtrage linéaire ou non

Filtrage spatial d'images : principe



Voisinage des pixels

- ▶ Notion fondamentale en traitement d'images
 - ▶ Suivi de contour, accroissement de régions, ...
 - ▶ Deux pixels voisins peuvent être agrégés
- ▶ On parle de connexité d'un pixel
 - ▶ Connexité d'ordre 4 (*4-Connectivity*) : on considère les 4 voisins (N, S, E ,O) du pixel
 - ▶ Connexité d'ordre 8 (*8-Connectivity*) : on considère les 8 voisins (N, S,E ,O, NO, SO, NE, SE) du pixel

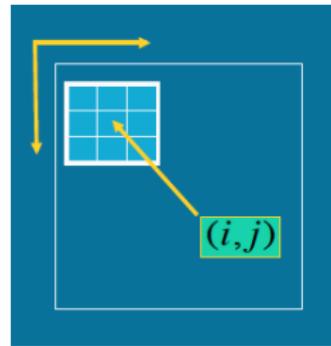


Filtrage spatial d'images : définitions

- ▶ Technique permettant de modifier ou améliorer les images
- ▶ Opération de voisinage qui effectue une combinaison linéaire (ou non) de pixels de l'image I , produisant une nouvelle image J :

$$J(x, y) = T(I)(x, y)$$

- ▶ T est un opérateur sur I défini en chaque pixel $I(x, y)$ et sur son voisinage
 - ▶ Le plus souvent, le domaine d'application de T est un carré
 - ▶ On appelle masque (ou filtre) cet opérateur



Filtrage spatial linéaire

- ▶ Le filtrage **linéaire** d'une image I peut se faire par une opération appelée convolution
 - ▶ Matrice de poids : noyau de convolution ou filtre, appelé h
 - ▶ Opération notée : $I * h$
- ▶ Principe de calcul de la convolution au pixel p de l'image I :
 1. Faire une symétrie du noyau par rapport à son centre
 2. Centrer le filtre sur p en le superposant à l'image
 3. Effectuer la somme pondérée entre les pixels de l'image et les coefficients du filtre
 4. Le pixel p dans l'image but (filtrée) aura comme valeur cette somme pondérée
- ▶ Convolution avec Inrimage : commande `conv.`

Convolution

- ▶ On considère un masque carré h de taille $d \times d$, vu comme une fenêtre “glissante” sur l’image
- ▶ Le produit de convolution d’une image I de taille $M \times N$ avec un filtre h au pixel (x, y) est donné par :

$$J(x, y) = (I * h)(x, y) = \sum_m \sum_n I(x - m, y - n)h(m, n)$$

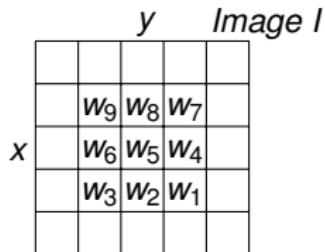
- ▶ En général, h est un masque carré de taille d impaire. On le suppose **centré** ($h(i, j), i, j \in \{-\frac{d-1}{2}, \dots, \frac{d-1}{2}\}$) et on a alors :

$$J(x, y) = (I * h)(x, y) = \sum_{m=-\frac{d-1}{2}}^{\frac{d-1}{2}} \sum_{n=-\frac{d-1}{2}}^{\frac{d-1}{2}} I(x - m, y - n)h(m, n)$$

Filtrage spatial linéaire

Exemple

$$h = \begin{pmatrix} w_1 & w_2 & w_3 \\ w_4 & w_5 & w_6 \\ w_7 & w_8 & w_9 \end{pmatrix}$$

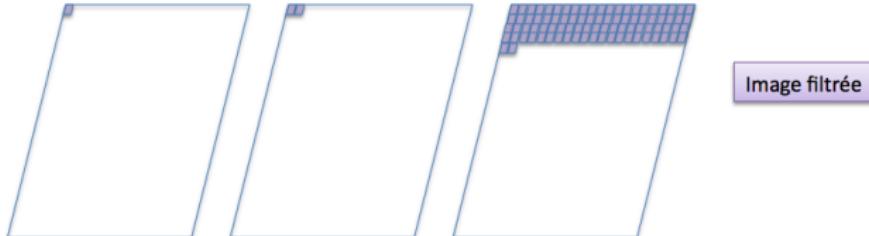
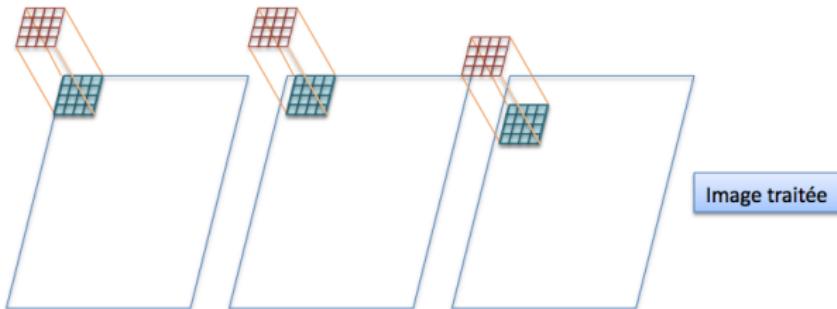


- ▶ La convolution au pixel (x, y) de I par le noyau h est donnée par :

$$\begin{aligned} J(x, y) &= w_9 I(x-1, y-1) + w_8 I(x-1, y) + w_7 I(x-1, y+1) \\ &+ w_6 I(x, y-1) + w_5 I(x, y) + w_4 I(x, y+1) \\ &+ w_3 I(x+1, y-1) + w_2 I(x+1, y) + w_1 I(x+1, y+1) \end{aligned}$$

Filtrage spatial linéaire

- Le calcul se fait sur chaque pixel de l'image : principe de la "fenêtre glissante"



Le problème des bords

- ▶ Que faire lorsque le filtre recouvre des zones n'appartenant pas à l'image ?
- ▶ Convolution linéaire : on considère que l'image est entourée de 0 (noir) ou on répète les bords !
- ▶ Convolution circulante : on considère que l'image est entourée d'elle-même (périodicité)
- ▶ Une autre solution : ne pas traiter les bords de l'image

Le filtrage passe-bas

- ▶ Caractéristiques du masque :
 - ▶ Coefficient central supérieur ou égal aux autres
 - ▶ Coefficients positifs
 - ▶ Pour conserver la moyenne originale de l'image normalise les coefficients du filtre ($\sum w_i = 1$)
- ▶ Effets du filtrage :
 - ▶ Le pixel central devient égal à une moyenne pondérée des pixels voisins
 - ▶ Les régions présentant de grandes zones de pixels homogènes
 - ▶ Peu changées, voire étalée par le filtrage
 - ▶ Peuvent ensuite être examinées de plus près
 - ▶ Les régions présentant des variations brusques d'intensité
 - ▶ Le moyennage local atténue ces variations (phénomène de flou)
 - ▶ Réduction du bruit, également des frontières

Le filtrage passe-bas

Exemple (le filtre moyenneur de taille de taille $d \times d$)

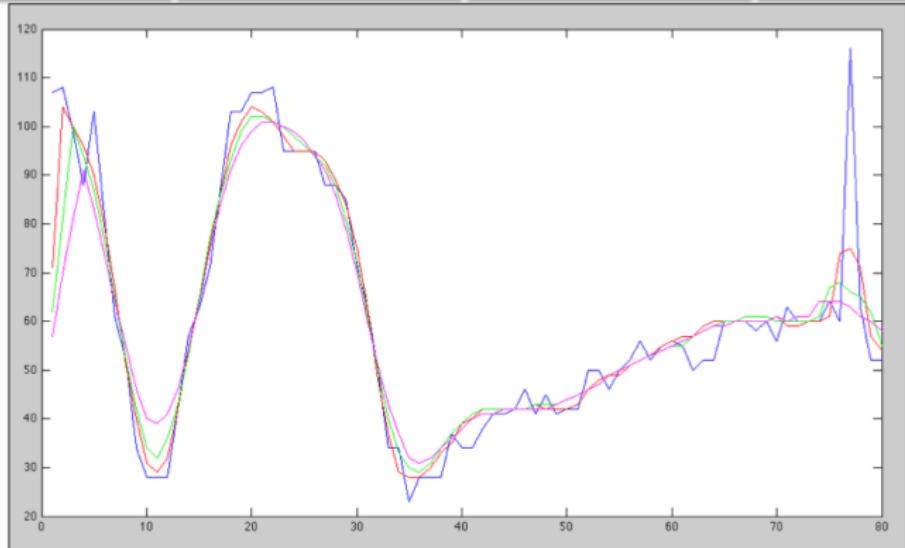
$$h = \frac{1}{d^2} \begin{pmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{pmatrix}$$

où h est une matrice de taille $d \times d$:

```
||  for d in 3 5 7 ; do fmoy lena.inr -x $d -y $d | xvis -nu; done
```



Le filtrage passe-bas : regardons de plus près



Le filtrage passe-haut

- ▶ Caractéristiques du masque :
 - ▶ Coefficient central positif et élevé
 - ▶ Autres coefficients plus petits, souvent négatifs ou nuls : atténuent l'effet des pixels alentour par rapport au pixel central
 - ▶ La somme des coefficients est nulle
- ▶ Effets du filtrage :
 - ▶ Les régions présentant de grandes zones de pixels homogènes
 - ▶ Perte de l'information d'intensité
 - ▶ Mieux délimitées en terme de régions
 - ▶ Les régions présentant des variations brusques d'intensité
 - ▶ Accentuées par le fort poids du pixel central du masque
 - ▶ Mise en évidence des frontières et autres changements brusques d'intensité

Le filtrage passe-haut

Exemple (le filtre laplacien sur une image naturelle)

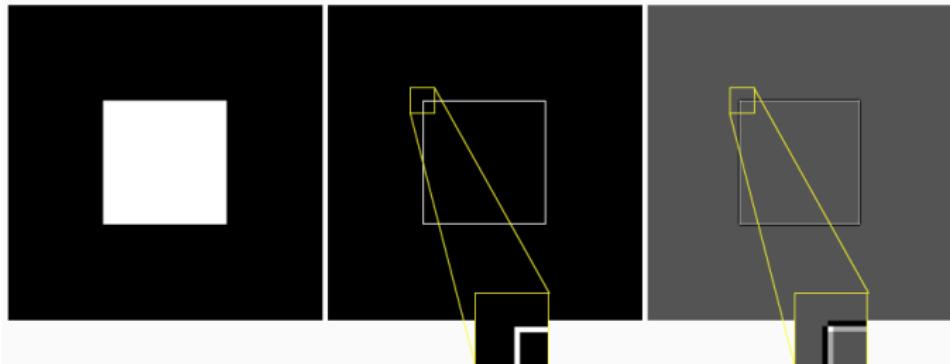
$$h = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

```
|| echo 0 -1 0 -1 4 -1 0 -1 0 | cim -x 3 -y 3 -r | conv lena.inr \
| xvis -nu
```



Le filtrage passe-haut

Exemple (le filtre laplacien sur une image synthétique)



0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	255	255	255	255
0	0	0	255	255	255	255
0	0	0	255	255	255	255
0	0	0	255	255	255	255

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	-255	-255	-255	-255	-255
0	0	-255	510	255	255	255	255
0	0	-255	255	0	0	0	0
0	0	-255	255	0	0	0	0
0	0	-255	255	0	0	0	0

Filtrage par corrélation

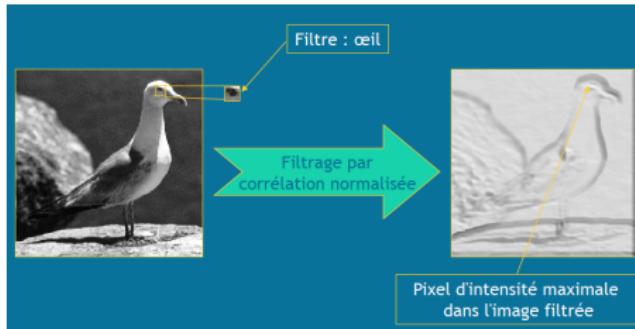
- ▶ L'opération de corrélation est très liée à celle de convolution
 - ▶ Matrice de poids : noyau de corrélation h
 - ▶ Indique une notion de similarité entre une portion de l'image et le noyau utilisé (corrélation normalisée)

$$J(x, y) = \sum_{m=-\frac{d-1}{2}}^{\frac{d-1}{2}} \sum_{n=-\frac{d-1}{2}}^{\frac{d-1}{2}} I(x+m, y+n)h(m, n)$$

- ▶ Principe de calcul de la corrélation au pixel p
 1. Centrer le filtre sur p en le superposant à l'image
 2. Effectuer la somme pondérée entre les pixels de l'image et les coefficients du filtre
 3. Le pixel p dans l'image but (filtrée) aura comme valeur cette somme pondérée
- ▶ Si le noyau h est symétrique, alors les opérations de convolution et corrélation sont équivalentes

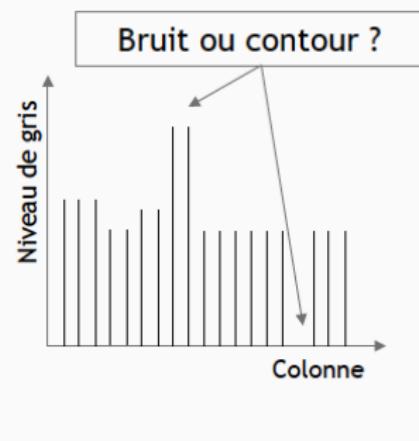
Filtrage par corrélation normalisée

- ▶ Application à la détection (*feature detection*) pour localiser la zone de l'image ressemblant le plus au filtre
- ▶ La corrélation normalisée est également une opération de filtrage linéaire
- ▶ On divise le résultat du calcul de la corrélation par la norme de la zone de l'image recouverte par le filtre
- ▶ Le pixel d'intensité maximale dans cette image filtrée détermine la position centrale la plus vraisemblable pour le filtre dans l'image



Réduction du bruit dans une image

- Bruit : phénomène parasite aléatoire (suivant une distribution de probabilité connue ou non) dont les origines sont diverses (capteur, acquisition, lumière, ...)

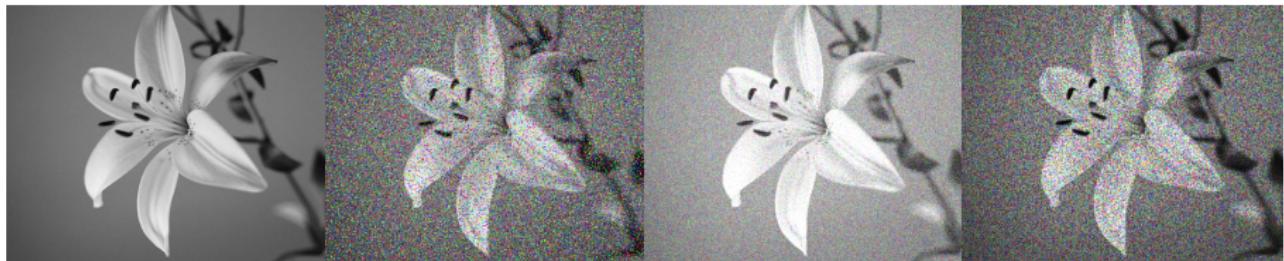


Qu'est-ce que le bruit ?

- ▶ Contexte d'acquisition :
 - ▶ sous- ou sur-illuminations ;
 - ▶ perturbation des capteurs (ex. ondes magnétiques pour les IRM, antenne de télévision liée à une carte tuner) ;
 - ▶ perturbations optiques : distorsions géométriques ou d'intensité.
- ▶ Échantillonnage :
 - ▶ bruit de Moiré (sous-échantillonnage) ;
 - ▶ bruit poivre et sel (pixels trop gros pour représenter des petits objets).
- ▶ Transmission des images : perturbations variées (corruption ou perte de données)

Quelques modèles de bruit

- ▶ Bruit gaussien (bruit additif)
- ▶ Bruit impulsionnel (bruit additif)
- ▶ Flou (bruit convolutif)
- ▶ Bruit multiplicatif (bruit multiplicatif)



Méthodes de réduction de bruit

- ▶ Un bruit se caractérise par des transitions brutales dans l'image (pixels de couleur aléatoire) : un bruit est une haute fréquence
- ▶ On le réduit en utilisant des filtres
- ▶ Les filtres linéaires, tels que :
 - ▶ le filtre moyenneur ;
 - ▶ le filtre gaussien.
- ▶ Les filtres non linéaires : le filtre médian

Le filtrage linéaire pour la réduction de bruit

- ▶ On considère que le bruit b est additif à l'image I :

$$I_b(x, y) = I(x, y) + b(x, y)$$

- ▶ On utilise des filtres dits passe-bas (on verra plus tard pourquoi) : beaucoup de filtres !
- ▶ En pratique ces filtres ne sont, en général, pas très bons : on dégrade l'image
- ▶ On doit trouver un compromis entre la quantité de bruit à enlever et la qualité de l'image restituée

Filtrage linéaire : filtre moyenneur

- ▶ Moyennage local de l'image : lissage
- ▶ Filtre (souvent) carré de taille impaire en x et y
- ▶ Fenêtre “glissante” sur l'image, centrée en chaque pixel traité : convolution linéaire

Exemple (filtre moyenneur de taille 3×3 sur une portion d'image)

$$\begin{matrix} 3 & 3 & 3 & 3 & 3 & 3 \\ 3 & 10 & 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 & 12 & 3 \\ 3 & 3 & 3 & 3 & 3 & 3 \\ 15 & 3 & 3 & 3 & 3 & 3 \end{matrix} * \frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} = \begin{matrix} 2 & 3 & 3 & 2 & 2 & 1 \\ 3 & 4 & 4 & 3 & 3 & 2 \\ 4 & 4 & 4 & 4 & 4 & 3 \\ 2 & 3 & 3 & 4 & 4 & 3 \\ 3 & 4 & 3 & 4 & 4 & 3 \\ 3 & 3 & 2 & 2 & 2 & 1 \end{matrix}$$

Filtrage linéaire : filtre moyenneur

- ▶ Plus le filtre est large, plus le bruit est atténué, mais plus la qualité de l'image est dégradée (phénomène de flou)



Filtrage linéaire : filtre gaussien

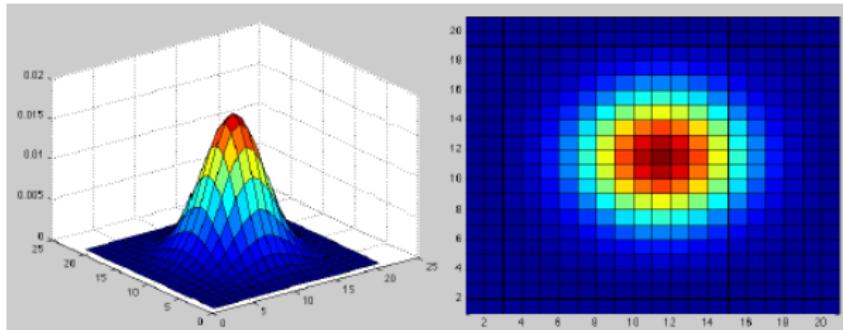
- ▶ Filtre passe-bas éliminant moins brutalement les hautes fréquences (lissage plus modéré)
- ▶ Il est donné par :

$$h(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$

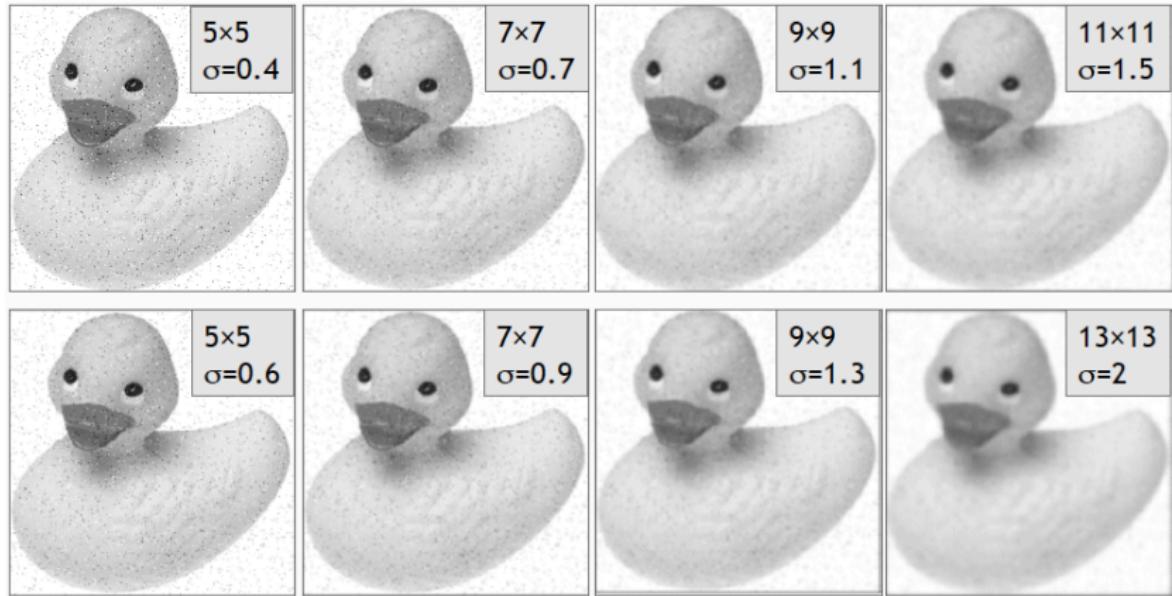
- ▶ La largeur (et donc l'efficacité) du filtre est donnée par sa variance σ^2
- ▶ Largeur du filtre de part et d'autre du point central : $\text{Int}^+(3\sigma)$, où Int^+ est la partie entière supérieure
- ▶ Largeur totale du filtre : $2 \times \text{Int}^+(3\sigma) + 1$

Filtrage linéaire : filtre gaussien

- ▶ Le noyau gaussien est défini par un ensemble de coefficients qui sont des échantillons de la gaussienne 2D
- ▶ Si σ est plus petit qu'un pixel le lissage n'a presque pas d'effet
- ▶ Plus σ est grand, plus on réduit le bruit, mais plus l'image filtrée est floue
- ▶ Si σ est choisi trop grand, tous les détails de l'image sont perdus



Filtrage linéaire : filtre gaussien



Filtrage linéaire : filtre gaussien

Exemple (pour $\sigma = 0.625$ ($\frac{1}{\sqrt{2\pi}\sigma} = 0.4$))

- ▶ Largeur du filtre de part et d'autre du point central :
 $\text{Int}^+(3\sigma) = 2$
- ▶ Largeur totale du filtre : $2 \times \text{Int}^+(3\sigma) + 1 = 5$
- ▶ On obtient le filtre suivant :

$$h = 0.4 \times 10^{-2} \times \begin{pmatrix} 0.03 & 0.16 & 5.98 & 0.16 & 0.03 \\ 0.16 & 7.7 & 27.8 & 7.7 & 0.16 \\ 5.98 & 27.8 & 100 & 27.8 & 5.98 \\ 0.16 & 7.7 & 27.8 & 7.7 & 0.16 \\ 0.03 & 0.16 & 5.98 & 0.16 & 0.03 \end{pmatrix}$$

Filtrage non linéaire : filtre médian

- ▶ Filtre non linéaire
- ▶ Méthode :
 - ▶ Déplacer une fenêtre de taille impaire sur le support image
 - ▶ Remplacer le pixel central (sur lequel est positionnée la fenêtre) par la valeur médiane des pixels inclus dans la fenêtre
- ▶ Soit une séquence discrète $\{a_1, a_2, \dots, a_N\}$ (N impair), a_i : est la valeur médiane de la séquence si :
 - ▶ il existe $\frac{N-1}{2}$ éléments de valeur inférieure et
 - ▶ il existe $\frac{N-1}{2}$ éléments de valeur supérieure.
- ▶ Très adapté au bruit type “poivre et sel” (faux blanc et noir dans l’image)
- ▶ Préserve les contours
- ▶ Réduit le bruit additif uniforme ou gaussien (lissage de l’image)
- ▶ Inrimage : commande `med`

Filtre médian

Exemple (filtre médian de taille 3×3 sur une portion d'image)

3	3	3	3	3	3	3	3	3	3	3	3	3
3	10	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	12	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
15	3	3	3	3	3	3	3	3	3	3	3	3

Filtre médian

