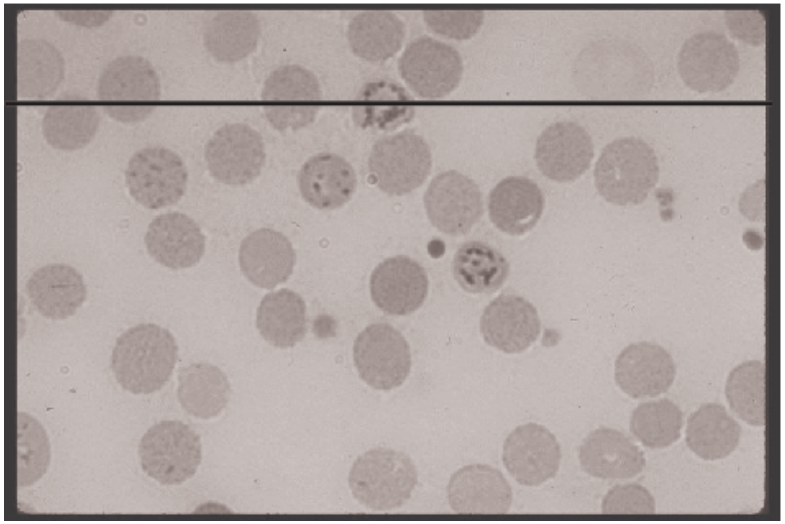
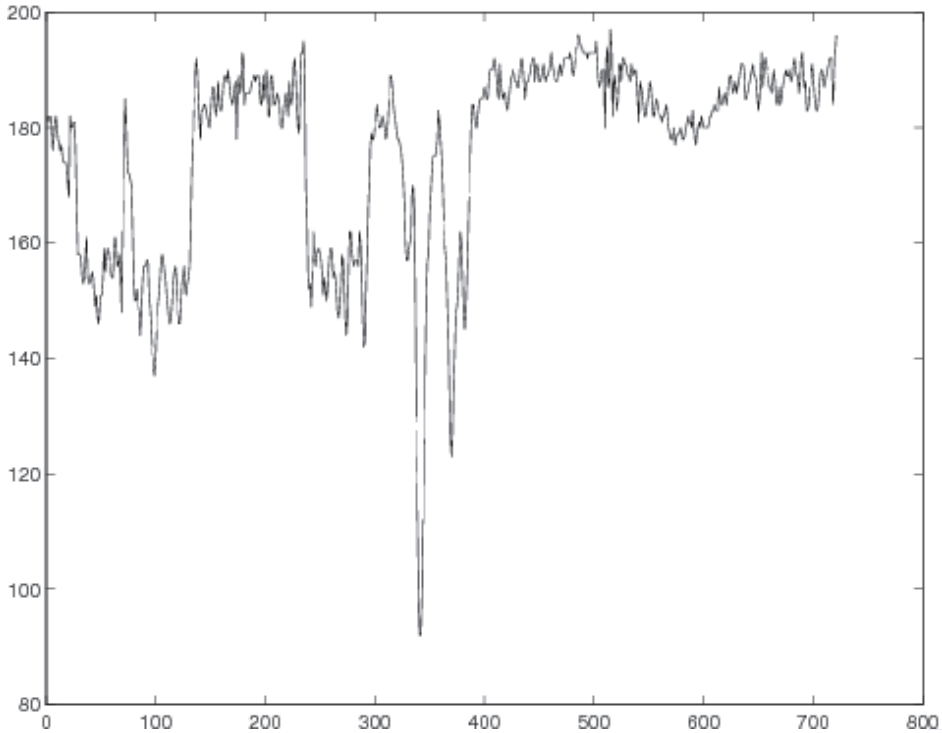
Filtrage

Marie-Odile Berger, INRIA Nancy Grand Est

La réalité d'une image numérique





Pourquoi restaurer/filtrer/améliorer une image?

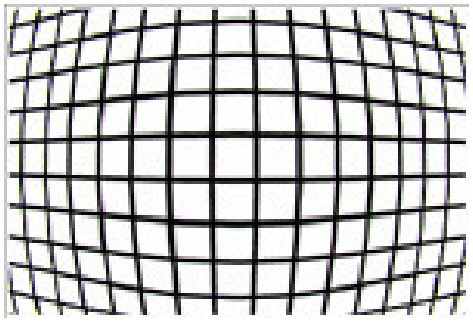
- Améliorer la qualité de l'image pour la rendre mieux interprétable
- A quoi est dû le bruit?
  - problème d'aquisition: flou de bouger, manque de lumière, perturbations magnétiques,
  - problème de quantification, distorsion de l'optique,
  - raisons extérieures: poussières sur l'objectif, rayure des films

La correction des distorsions

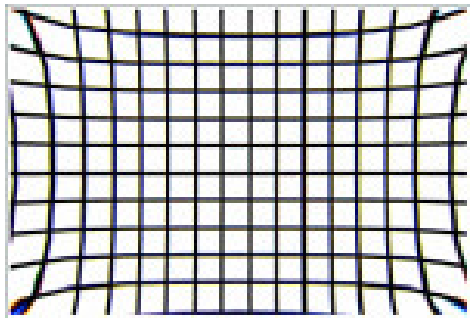
Origine des distorsions:

- lentilles non alignées, non positionnées perpendiculairement à l'axe principal $\rightarrow$ distorsion tangentielle
- lentilles non minces ou défauts de courbure $\rightarrow$ distorsions radiales.

le modèle sténopé n'est plus valable.



coussinet



barillet

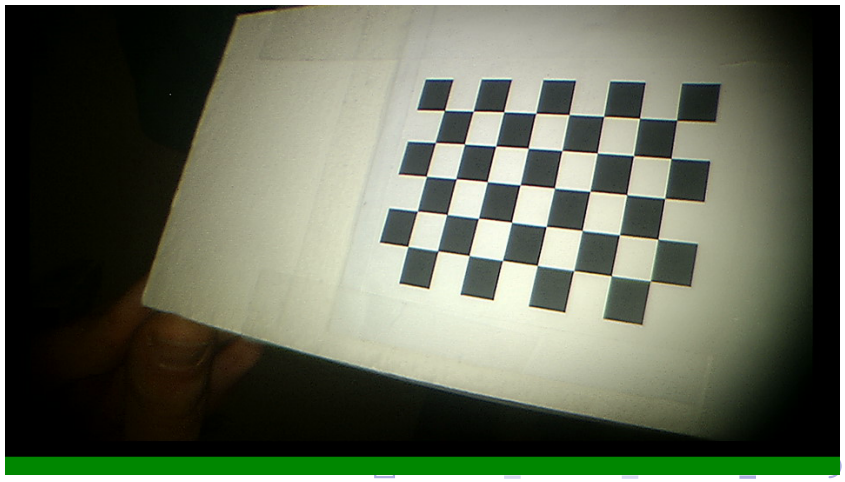
Prise en compte des distorsions

Pour des objectifs d'assez bonne qualité, la modélisation des distorsions radiales suffit souvent.
$\rightarrow$ on modélise la distorsion par une fonction analytique symétrique par rapport au centre optique.
Soit (u_p, u_v) le projeté du point sur l'image avec le modèle de projection centrale. Modèle de distorsion radiale:

$$\begin{aligned} u &= u_p + (u_p - u_0)(K_1 r^2 + K_2 r^4 \dots) \\ v &= v_p + (v_p - v_0)(K_3 r^2 + K_4 r^4 \dots) \end{aligned}$$

où $r = \sqrt{(u_p - u_0)^2 + (v_p - v_0)^2}$.

Les paramètres K_i sont calculés grâce à l'observation d'une mire



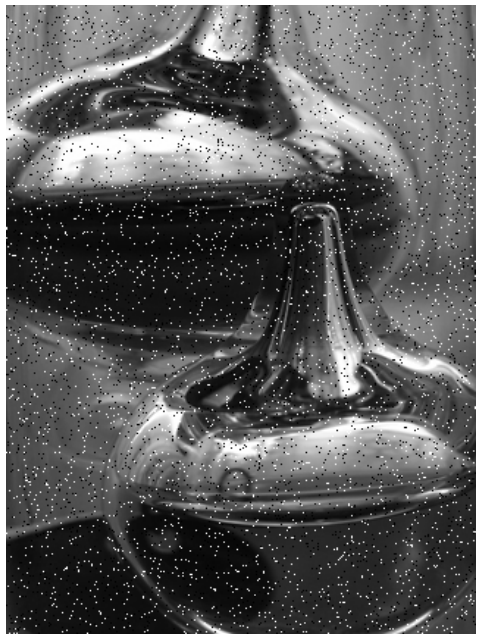
Dégradations courantes de l'images

bruit: information parasite s'ajoutant de façon aléatoire à la scène. particulièrement visible dans:

- Les zones où le rapport signal/bruit est faible (ex zones peu éclairées)
- les zones uniformes

Types de bruit:

- Le bruit impulsionnel (bruit sel et poivre):présence de pixels noirs et blancs répartis au hasard. Raisons: éléments du capteur CCD défaillants, poussières.
- burit de speckle : présence de grains, en particulier dans images ultrasonores
- bruit électronique, de grenaille, de quantification: souvent modélisé par un bruit gaussien



poivre et sel

Filtrage des images

- De nombreuses méthodes utilisent le voisinage d'un pixel pour
  - corriger éventuellement sa valeur et éliminer le bruit.
  - rehausser les indices et les rendre mieux détectables
- Les filtres peuvent être:
  - fréquentiel: élimination des hautes fréquences
  - spatiaux:
    - s'exprimer comme une convolution
    - non linéaires (filtre médian)
    - topologique

  Certains filtres sont spécifiques à un modèle mathématique de bruit donné (non abordé dans ce cours).

Part I

Les filtres fréquentiels

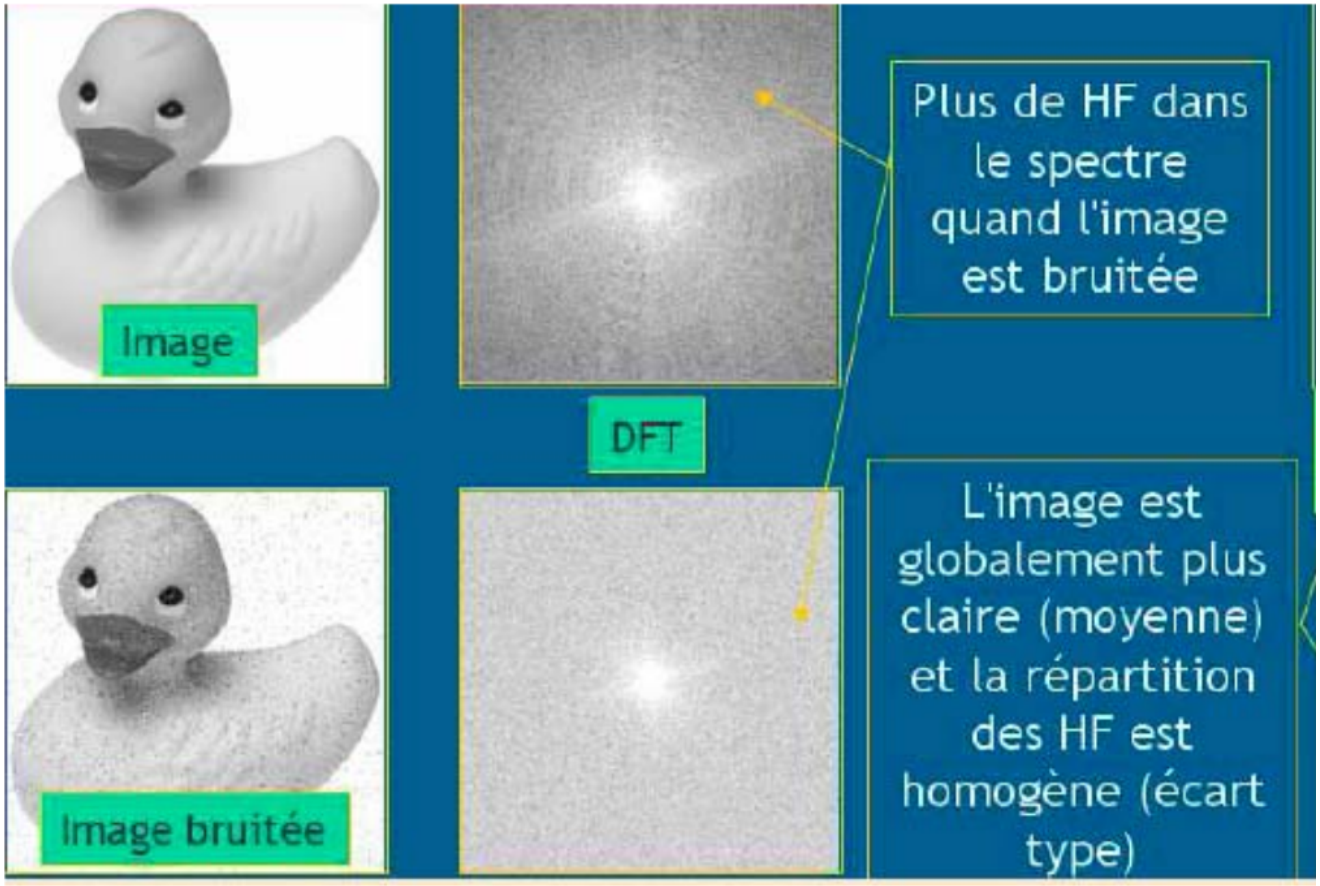
Filtres fréquentiels

- Idée générale: le bruit correspond à des hautes fréquences
- On travaille à partir d'une représentation fréquentielle de l'image, souvent la transformée de Fourier de l'image $f(x, y)$

$$F(u, v) = \int \int f(x, y) e^{-2\pi i(xu+yv)} dxdy$$

- principe: appliquer un filtre sur la représentation fréquentielle pour **éliminer les hautes fréquences** puis produire l'image filtrée par transformée de Fourier inverse.

Le bruit est une haute fréquence



Filtre fréquentiel

Exemple du filtre passe bas: on ne garde que les basses fréquences

- I=imread('venise.jpg');I=rgb2gray(I);[M,N]=size(I);
- F=fftshift(fft2(I)); H0=0;M2=round(M/2); N2=round(N/2);
- H0(M2-D0:M2+D0,N2-D0:N2+D0)=1;
- G=F.*H0;
- g=ifft2(G);

Tester ce filtre porteFourier sur l'image venise.jpg avec diverses valeurs de D0.

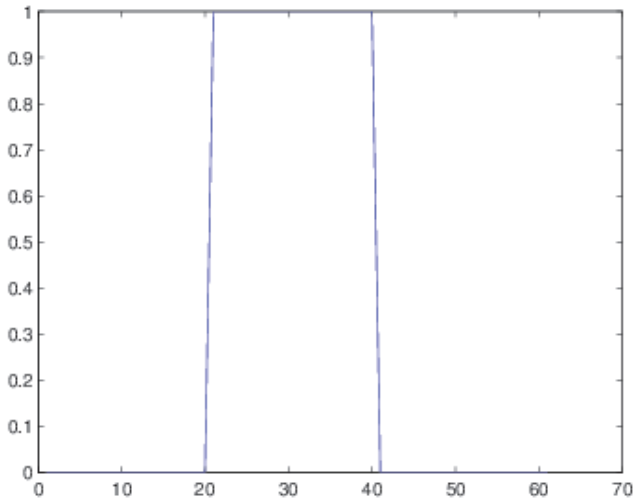
Filtre fréquentiels: résultats

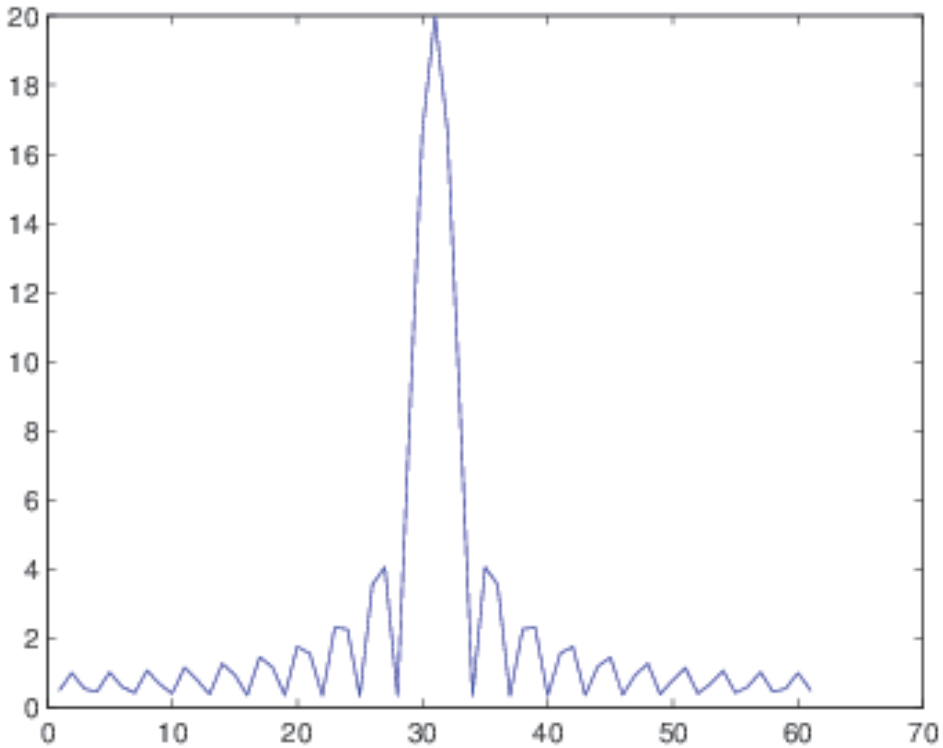
coupure à 50

à 100

Filtre fréquentiels: résultats

Problème: on enlève du bruit mais aussi des détails
raison: FFT de la fonction porte: existence de coefficients non nuls pour toutes les fréquences





Part II

Filtrage spatial

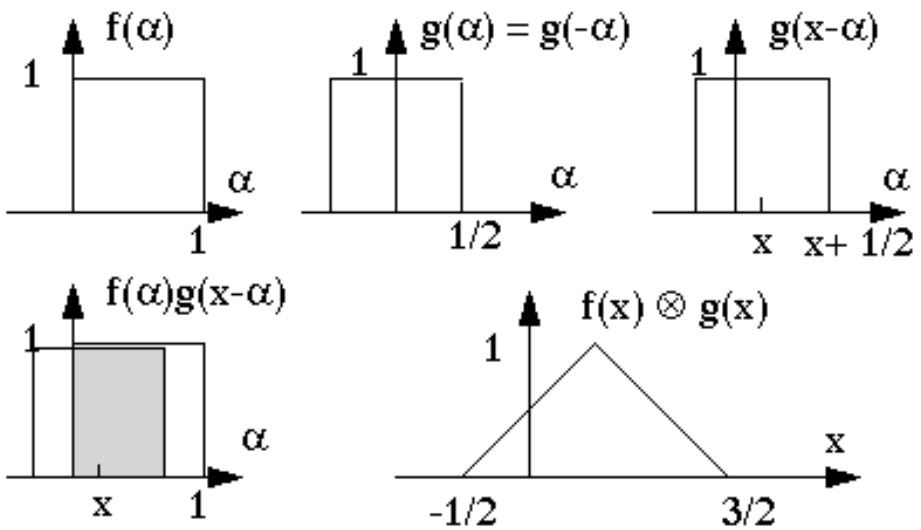
Filtre linéaires par convolution

But: remplacer un pixel par une moyenne (à définir) de ses voisins

$$f * g(x) = \int f(\alpha) g(x - \alpha) d\alpha$$

Dans le domaine discret, soit h un noyau de taille $[-k, k]x[-k, k]$.
L'image filtrée est donnée par

$$I_f(u, v) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} f(i, j) * I(u - i, v - j)$$



Filtre linéaires par convolution

Filtre le plus simple: le filtre moyenneur

1	1	1
1	1	1
1	1	1

Implantation:

h= 1/3 *ones(1:3,1:3);

Inew=conv2(double(I),double(h);

Inew=imfilter(I,h);

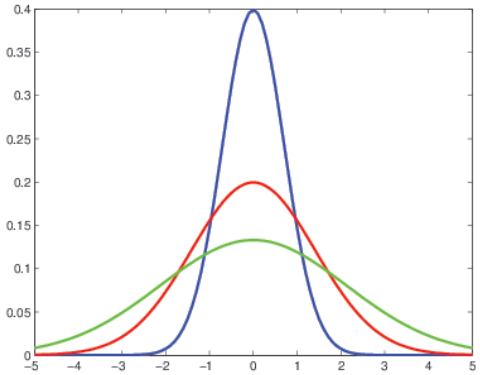
demie taille1

Flou important

et 2

Lissage: le filtre Gaussien

- L'opérateur gaussien de moyenne nulle est d'écart type σ est donné en dimension 1 par



$$G_\sigma = \frac{1}{\sqrt{2\pi}\sigma} exp^{-x^2/2\sigma^2}$$

- la convolution $u_\sigma = G_\sigma * u$ d'une image u par une gaussienne G_σ permet de lisser l'image à différentes échelles (initiale, 3,5,10).

Lissage: le filtre Gaussien

initial

$\sigma = 3$

H=fspecial('gaussian',10,3);% $\sigma = 3$, taille du filtre 10
imfilter(I,H)

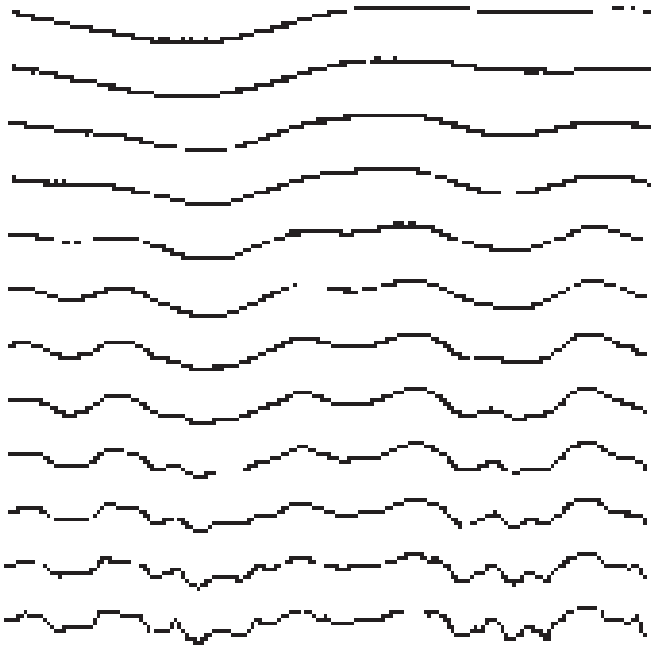
Lissage: le filtre Gaussien

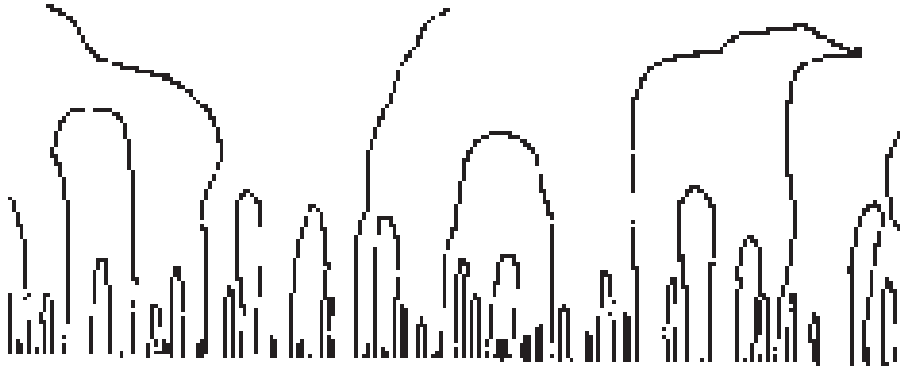
$\sigma = 5$

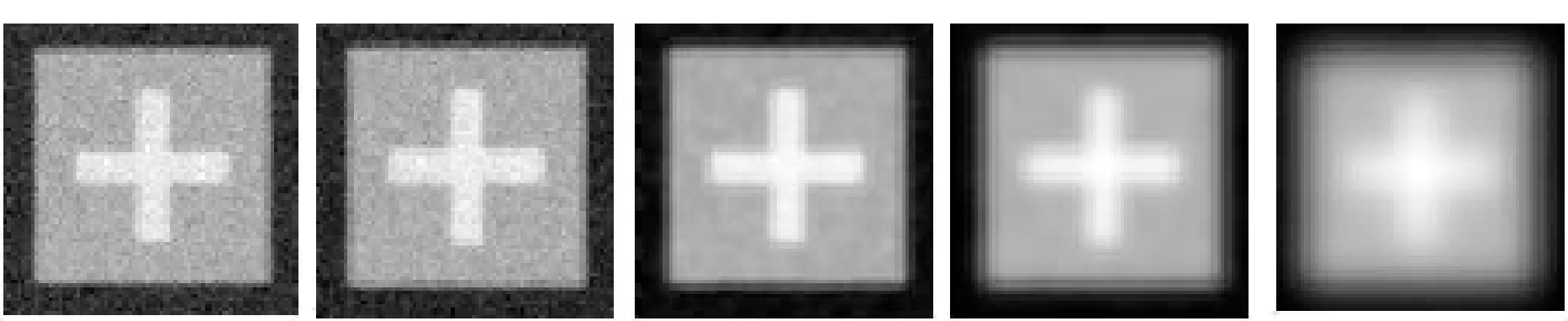
$\sigma = 10$

Lissage et extraction de contours

Lisser une image (ici filtre gaussien), entraine une délocalisation des points de contours...







Les filtres anistropes

- Constat: les problèmes viennent du fait qu'on lisse de la même manière dans toutes les directions indépendemment de la structure de l'image.
- Utiliser des filtres anisotropes lissant selon les caractéristiques de l'images
  - filtre bilatéral
  - filtre respectant les contours (Perona Malik)

Le filtre bilatéral

Voir [Tomasi98]

- **objectif**: préserver les discontinuités et **lisser les zones de photométrie semblable**
- Dans un filtre classique, seule la distance au point est considérée pour pondérer les valeurs: $\sum_{p \in Voisins(u)} w(|p-u|)I(p)$
- Dans un filtre bilatéral, on introduit un poids dépendant de la similarité des niveaux de gris:
  $\sum_{p \in Voisins(u)} w_s(|p-u|)w_p(|I(p)-I(u)|I(p)$
  $w_p(x) = e^{-x^2/2\sigma_p}, w_s(x) = e^{-x^2/2\sigma_s},$

Le filtre bilatéral

image initiale

bilateral

image initiale

gaussien

Filtres bilateraux: effet d'artiste

[Winnemoller, SIGGRAPH 2006]

Equation de la chaleur et filtre gaussien

- Rappels:

$$\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$$
$$div(u) = \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y}$$
$$\Delta u = div(\nabla u)$$

- L'équation de propagation de la chaleur:

$$\frac{\partial u(t,x)}{\partial t} = \Delta u(t,x),$$
$$u(0,x) = u_0(x)$$

- Propriété fondamentale:
  $I * G_\sigma$ est solution de l'équation de la chaleur à $t = \sigma^2/2$.

Schéma numérique

- en dimension 1, $\frac{\partial u(t,x)}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ avec la condition aux limites $u(0,x) = I$
- soit δt le pas de discrétisation. $\frac{\partial u(t,x)}{\partial t} \approx (u(t+\delta t, x) - u(t,x))/\delta t$
- Discrétisation spatiale avec un pas de 1 (pixel):
  $\frac{\partial u}{\partial x} \approx u(x+1) - u(x) \approx u(x) - u(x-1) \approx (u(x+1) - u(x-1))/2$
- Premier pas de temps:

$$(u(\delta t, x) - u(0,x))/\delta t = (u(x+1) - u(x-1))/2$$
$$\rightarrow u(\delta t, x) = u(0,x) + \delta t * (u(x+1) - u(x-1))/2$$

- itérations suivantes $u(t+\delta t, x) = u(t,x) + \delta t(u(x+1) - u(x-1))/2$

Filtrage anistropique: le filtre de Perona-Malik

- Modifier l'équation de la chaleur en n'autorisant la diffusion que lorsque le gradient est faible (pas de contour)
- $$\frac{\partial u}{\partial t} = \Delta u = div(\nabla u)$$

  devient
- $$\frac{\partial u}{\partial t} = div(c(|\nabla u|)\nabla u)$$

  ou c est une fonction décroissante avec $c(0) = 1$ et $lim_{x\to\infty} c(x) = 0$, par exemple $c(x) = \frac{1}{\sqrt{1+x^2}}$
- si $c = 1$, on retrouve l' équation de la chaleur $\rightarrow$ diffusion,
- si c est faible, la diffusion est stoppée ce qui préserve les bords.

Filtrage anistropique: résultats avec Perona-Malik

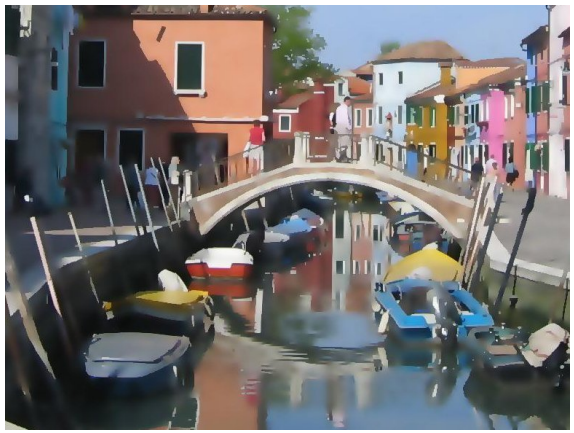
Initiale

50 it

100 it

Zoom

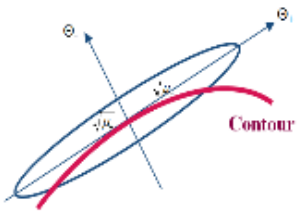
50 it



100 it

Filtrage anisotropique: améliorations ultérieures

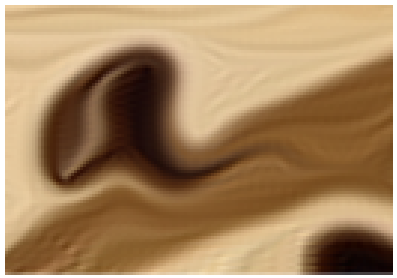
- Le filtre de Perona réduit la diffusion dans les zones à fort gradient mais il ne tient pas compte de la structure locale
- Améliorations ulterieures permettant des lissages directionels: on lisse tangentiellement au contour mais pas au travers du contour (ex Alvares, Weickert97) Toujours avec un schéma de type EDP.
- utilisation d'un tenseur de diffusion au lieu d'une diffusion scalaire



Tenseur

I originale

filtre Perona



filtre weickert

Filtre médian

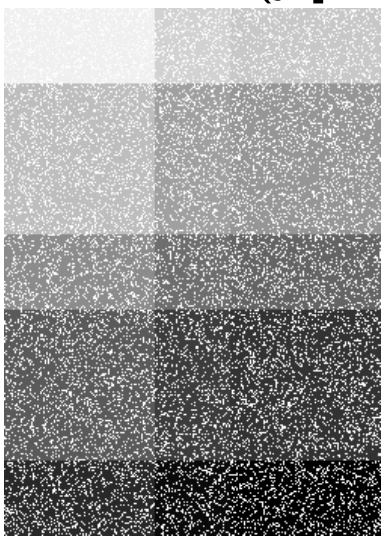
Définition de la **médiane** :
Soient n valeurs réelles $x_1, ..., x_n$, avec n impair. On les ordonne par ordre croissant: $x_{i_1}, ..., x_{i_n}$. La médiane est la valeur x_m avec $m = (n+1)/2$.
Prop: la médiane est robuste à la présence de 50% de données erronées, contrairement à la moyenne.
Filtre médian avec une fenêtre de demie-taille h:
$I_{new}(i,j) = mediane_{k \in [-h,h]} I(i-k : i+k, j-k : j+k)$

Propriétés du filtre médian

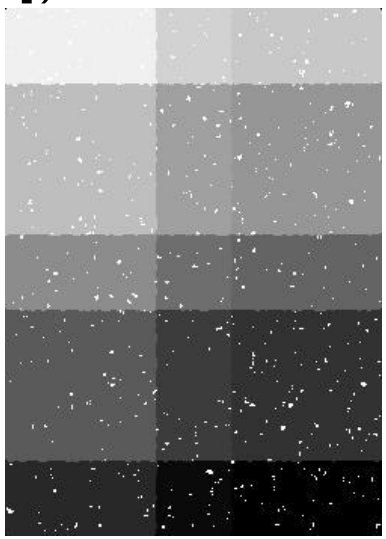
- Il ne crée pas de nouveaux niveaux de gris (ne résoud pas les problèmes de dynamique)
- il génère peu de flou (sauf au niveau de courbes ou coins)
- bien adapté au bruit poivre et sel
- nécessite de classer les valeurs dans chaque voisinage

Filtre médian:résultats

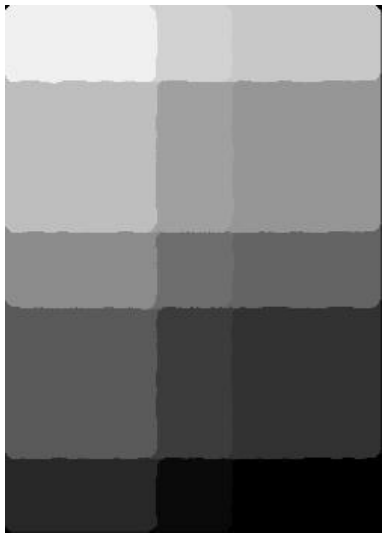
k=medfilt2(j,[10,10])



initiales



taille=3



taille = 10

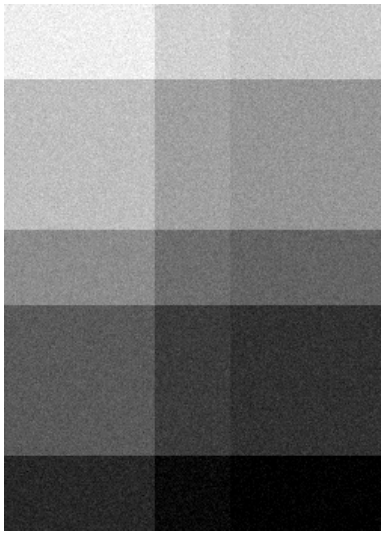
bruit impulsionnel

Q: Pourquoi a-t-on un arrondissement des coins?

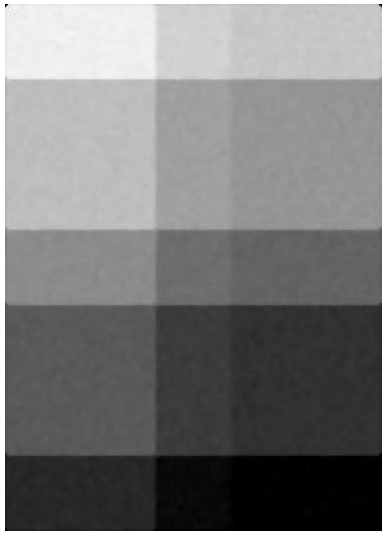
Filtre médian:résultats

k=medfilt2(j,[10,10]). exemple tiré de:

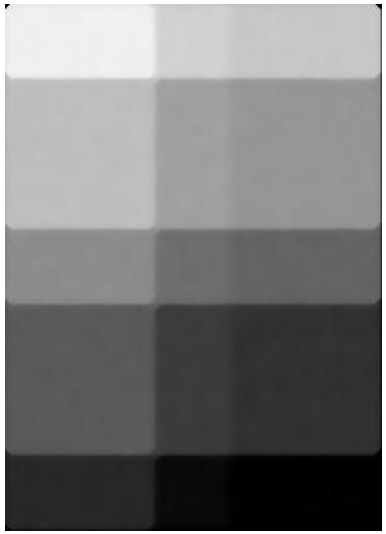
http://www.tsi.telecom-paristech.fr/pages/enseignement/ressources/beti/bruit_mult/test-apres.html



initiales



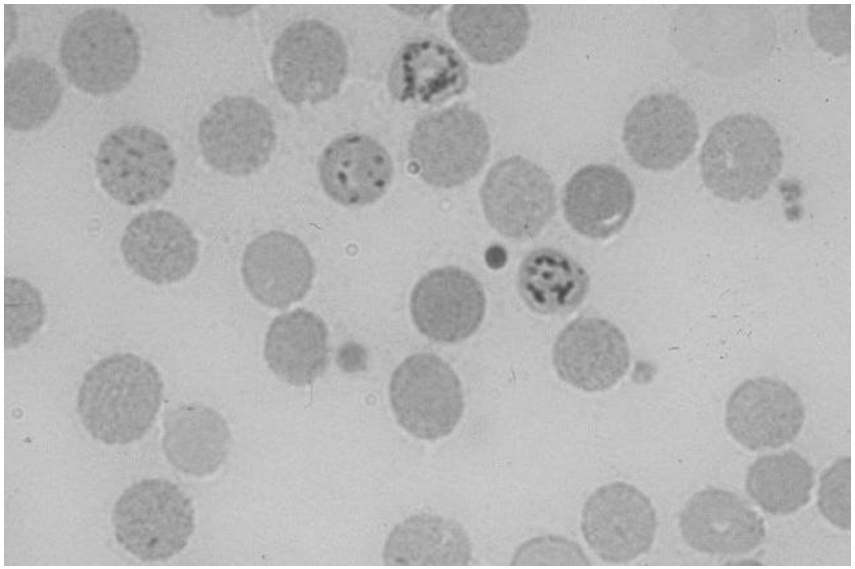
taille=3



taille = 10

bruit gaussien additif

TP: comparer les filtrages



Objectif: utiliser l'image cellule.jpg. On souhaite obtenir une image binarisée délimitant bien toutes les cellules. Vous pouvez utiliser l'outil *imtool* pour voir les niveaux de gris des cellules

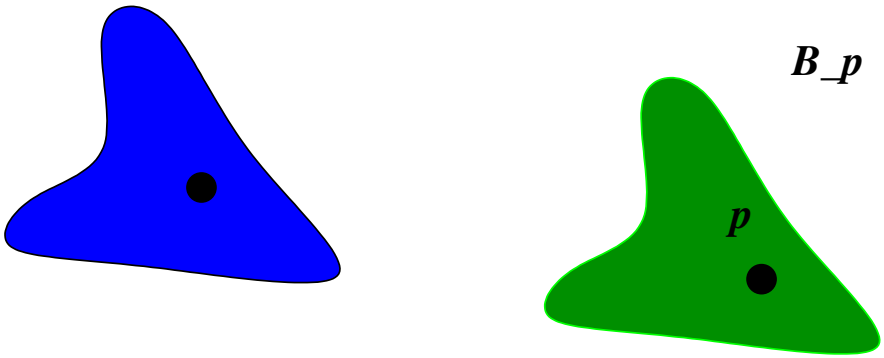
- Testez les filtres bas niveaux disponibles: gaussien, medfilter2, bfilter2 (cartoon). Essayez d'obtenir une image binarisee (im2bw) ne laissant voir que les cellules
- Note 1: utiliser *graythresh* pour définir au mieux le seuil de binarisation (méthode d'Otsu: minimisation de la variance intra classe). Vous pouvez aussi regarder l'histogramme
- Note 2: pensez à placer CodeExterne dans votre path pour voir accès à bfilter2 (prend en entrée des images de type double codées entre 0 et 1).
- Note 3: Mettre des titres au dessus de vos images: figure('Name','mon titre');
- Pourquoi n'arrive-t'on pas à récupérer la cellule la plus claire en haut à droite?
- solution: tp_cellule

Morphologie mathématique

- Techniques de filtrage et d'analyse basée sur des théories ensemblistes et algébriques [Matheron & Serra 65]
- idée de base: comparer localement les formes avec un élément dit structurant qui sert à analyser la forme et permet de filtrer l'image
- La morphomat initialement définie pour des images binaires a été ensuite étendue aux images à niveau de gris

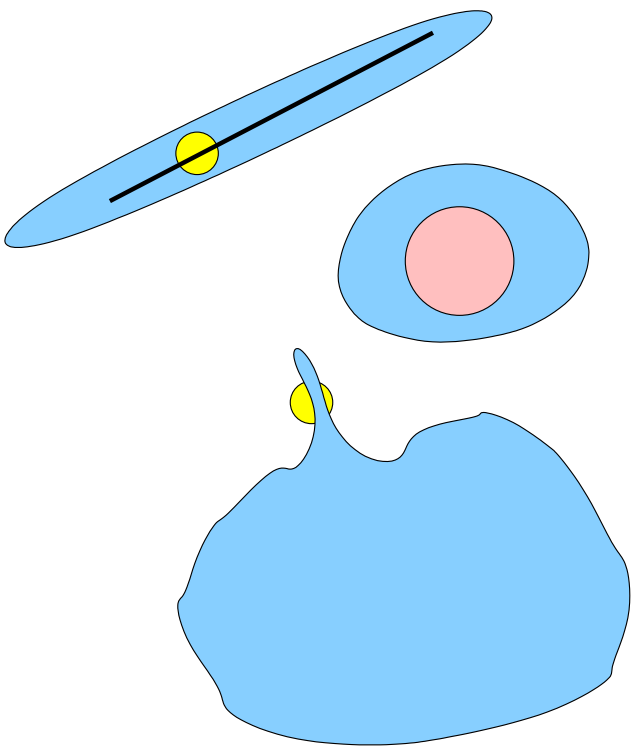
Eléments structurants

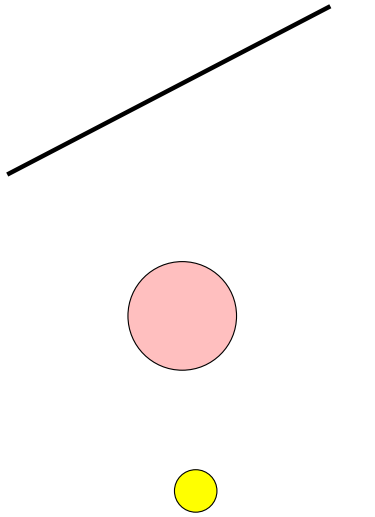
- Structure qui va permettre l'analyse locale de la forme
- souvent de forme régulière (disque, carré, ligne) mais on peut prendre n'importe quoi en fonction de ses besoins
- il est repéré par son origine (ou centre)
- La fenêtre d'analyse associée à un pixel p sera le translaté de B positionné sur p,et est noté B_p.



L'analyse morphologique des images

Filtrer les images en les comparant localement à un élément structurant et répondre à des questions telles que l'élément est-il contenu dans la forme?





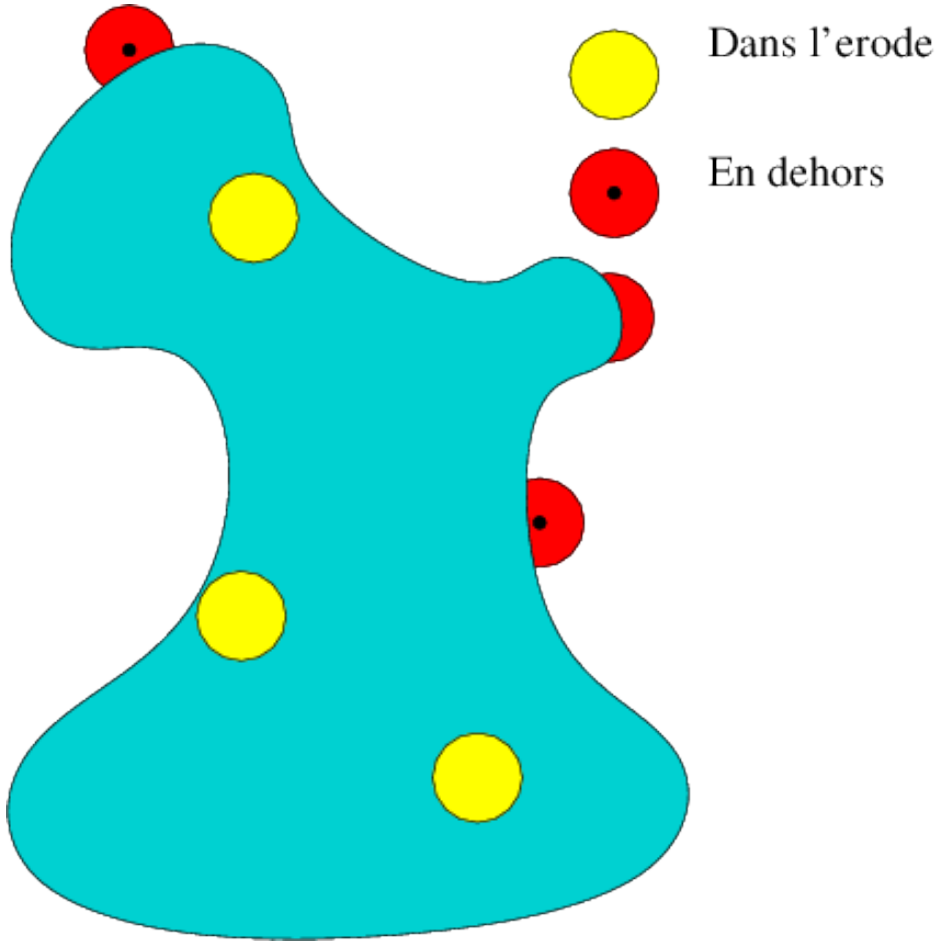
element structurant

Permet de filtrer selon la taille, la direction...

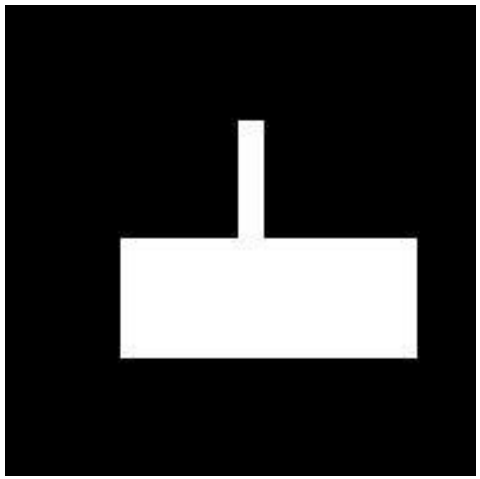
Erosion morphologique binaire

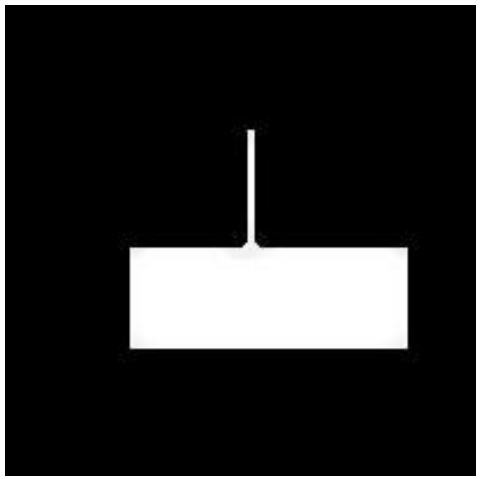
Soit X un ensemble et B l'élément structurant.
L'érosion de X par B est l'ensemble des points tels que si B est centré sur ce point, alors B est entièrement contenu dans X

$$Erode_B(X) = \{x : B_x \subset X\}$$

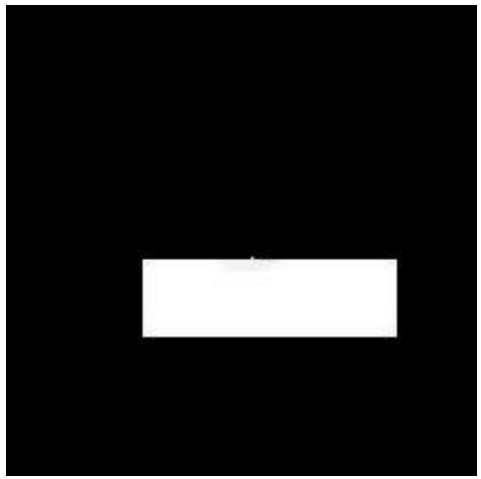


Erosion

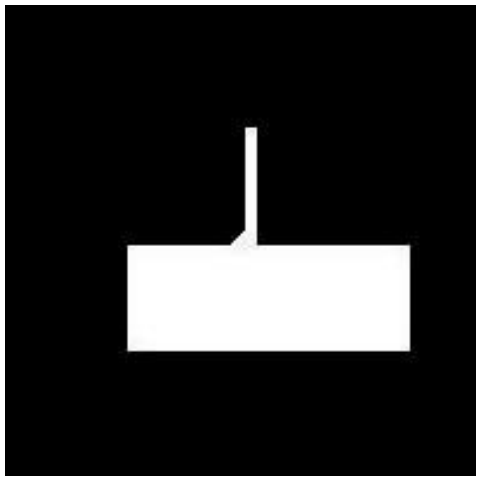




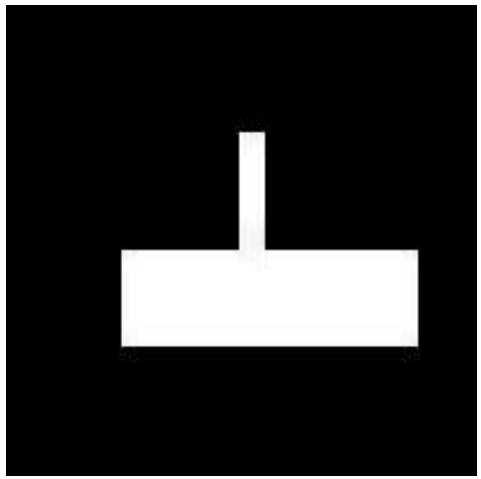
imerode(i,strel('disk',5))



imerode(i,strel('disk',10));



iimerode(i,strel('line',10,45));

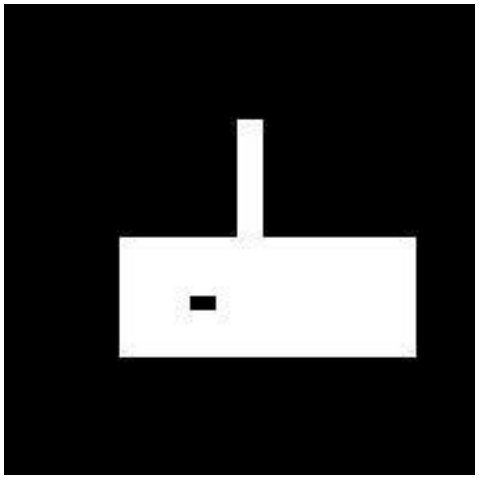


imerode(i,strel('line',10,90));

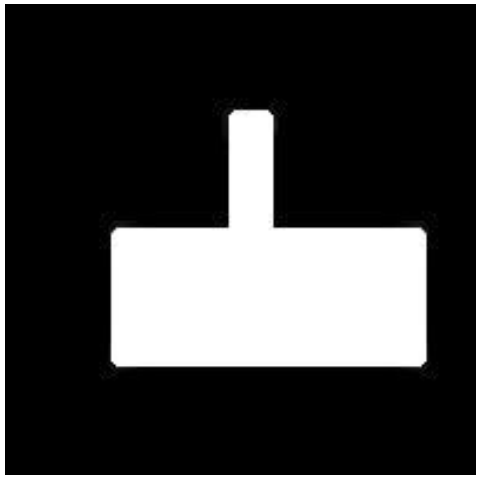
L'érosion fait disparaitre les petits objets et amincit les formes

La dilatation binaire

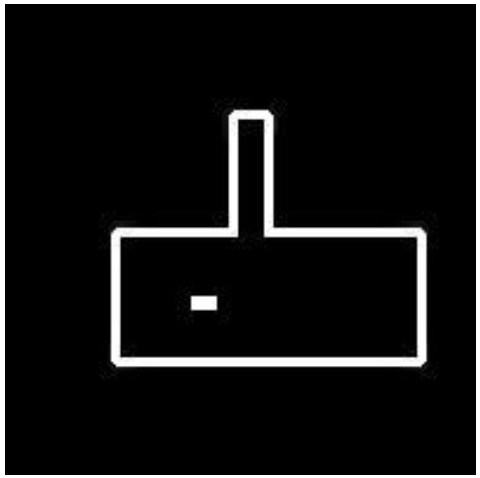
$Dilate_B(X)$ est l'ensemble des points tels que lorsque B est centré sur un de ces points, il y a une intersection non vide entre X et B.
$Dilate_B(X) = \cup\{B_x, x \in X\}$



original



imdilate(i,strel('disk',5))



difference

La dilatation fait disparaitre les petits trous et fait grossir l'objet.

Ouverture et fermeture

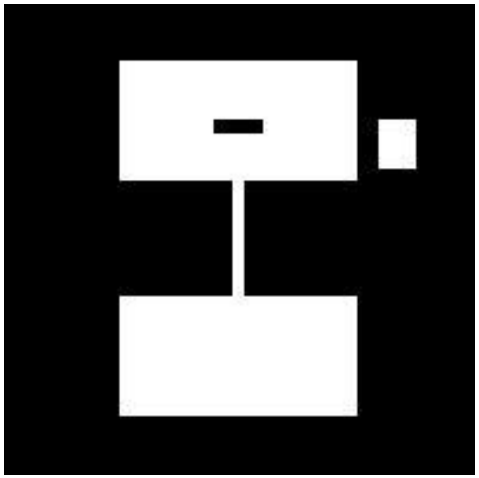
L' ouverture par B est la composition de l'érosion par B suivie de la dilatation par B :

$$Ouv_B(X) = DilateB_(Erode_B(X)),$$

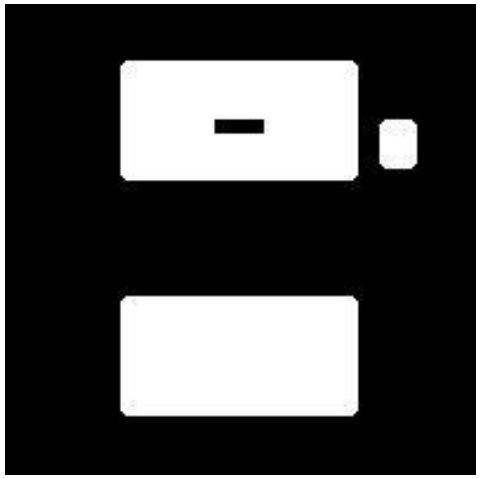
La fermerture par B est la composition de la dilatation par B suivie de l'érosion par B :

$$Ferm_B(X) = Erode_B(Dilate_B(X))$$

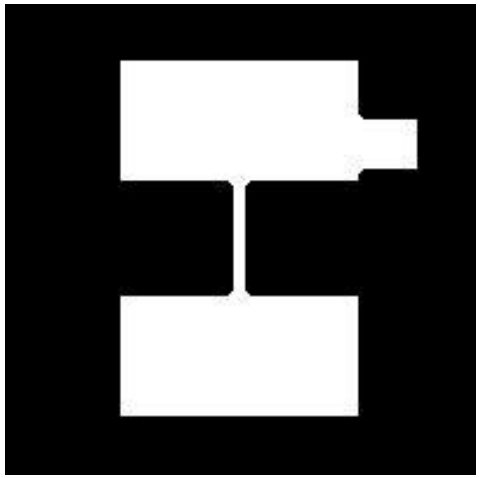
Ouverture et fermeture



original



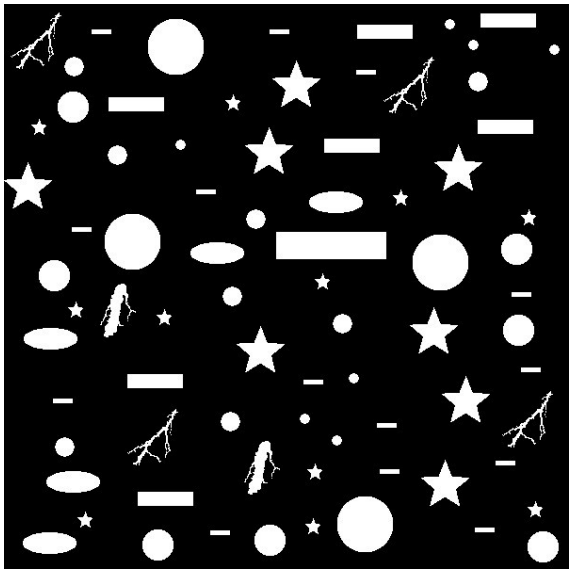
imopen(i,strel('disk',7))



imclose(i,strel('disk',7))

- une ouveture supprime les petites particules, sépare les formes là où elles sont étroites.
- une fermeture fait disparaitre les trous de petite taille et connecte les parties proches.

La Morphologie mathématique: TP granulométrie



- chargez l'image symboles.jpg
- Quel est le nombre de symboles présents dans cette images (fonction bweuler)?
- éliminer tous les symboles de taille $<$ à 100 pixels (bwareaopen)
- pour $i = 1 : 3500$, calculer le nombre N(i) de pixels appartenant à une structure de taille $> i$. Visualiser cette courbe (plot)
- en déduire un moyen d'afficher une image ne contenant qu'un seul type de symbole.
- solution: tpSymbole

La Morphologie mathématique: TP

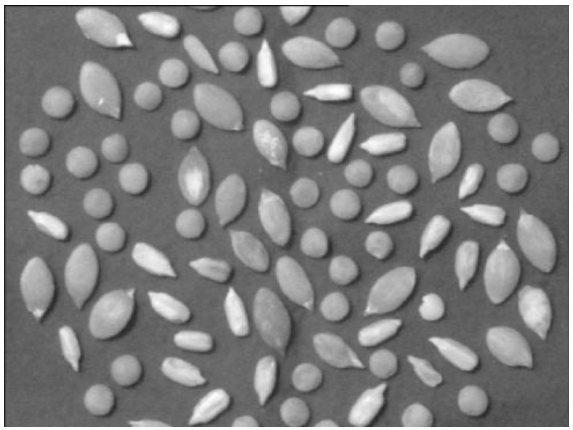


Image tirée de

`http://www.irit.fr/ Philippe.Joly/`

- Chargez l'image triGrain.jpg
- En considérant l'histogramme de cette image, trouvez un seuil permettant de la binariser correctement (fonction *im2bw*).
- utiliser un operateur morphologique pour enlever les bruit restant.
- Trois types de grains composent cette image. Comment isoler les grosses graines?
- Proposer une méthode permettant de distinguer les 2 types de graines les plus petites. Quelles sont les difficutés?
- NB: vous pouvez utiliser imtool qui permet de faire des mesures sur l'images. La fonction *bweuler* permet de compter le nombre de composantes connexes (et donc de graines) présentes dans l'image.