Traitement mathématiques des images (III) Filtrage — Convolution

bruno.colombel@univ-amu.fr

DUT Informatique
IUT d'Aix-Marseille
Site d'Arles

2019 — 2020

Bruit

Produit de convolution

Lissage (Filtrage spatial)

Filtrage fréquentiel

Sommaire

Le bruit

Dans toute image numérique, les valeurs de gris ou de couleur observées présentent une incertitude

Le bruit

Dans toute image numérique, les valeurs de gris ou de couleur observées présentent une incertitude

- Incertitudes liées aux capteurs

Le bruit

Dans toute image numérique, les valeurs de gris ou de couleur observées présentent une incertitude

- Incertitudes liées aux capteurs
- photons parasites

Le bruit

Dans toute image numérique, les valeurs de gris ou de couleur observées présentent une incertitude

- Incertitudes liées aux capteurs
- photons parasites
  - négligeables si la scène est bien éclairée

Le bruit

Dans toute image numérique, les valeurs de gris ou de couleur observées présentent une incertitude

- Incertitudes liées aux capteurs
- photons parasites
  - négligeables si la scène est bien éclairée
  - les zones sombres sont « bruitées »

Le bruit

Le bruit est dans la majorité des cas dû aux conditions d'acquisition :

Le bruit

Le bruit est dans la majorité des cas dû aux conditions d'acquisition :

- erreurs de mesure ou de manipulation

Le bruit

Le bruit est dans la majorité des cas dû aux conditions d'acquisition :

- erreurs de mesure ou de manipulation
- perturbations par le milieu ambiant
  - images médicales (radiographie, échographie) : information bruitée par la traversée des tissus

Le bruit

Le bruit est dans la majorité des cas dû aux conditions d'acquisition :

- erreurs de mesure ou de manipulation
- perturbations par le milieu ambiant
  - images médicales (radiographie, échographie) : information bruitée par la traversée des tissus
- mauvaises conditions d'acquisition :
  - nuages pour les photos aériennes, manque de luminosité, etc.

Le bruit

Le bruit est dans la majorité des cas dû aux conditions d'acquisition :

- erreurs de mesure ou de manipulation
- perturbations par le milieu ambiant
  - images médicales (radiographie, échographie) : information bruitée par la traversée des tissus
- mauvaises conditions d'acquisition :
  - nuages pour les photos aériennes, manque de luminosité, etc.

Mais aussi :

- vieillissement des supports

Le bruit

Le bruit est dans la majorité des cas dû aux conditions d'acquisition :

- erreurs de mesure ou de manipulation
- perturbations par le milieu ambiant
  - images médicales (radiographie, échographie) : information bruitée par la traversée des tissus
- mauvaises conditions d'acquisition :
  - nuages pour les photos aériennes, manque de luminosité, etc.

Mais aussi :

- vieillissement des supports
- traitement numérique, etc.

Bruit thermique

- dû à l'agitation naturelle des électrons, qui augmente avec la température du capteur

Bruit thermique

- dû à l'agitation naturelle des électrons, qui augmente avec la température du capteur
- peut être réduit efficacement par refroidissement du capteur

Bruit thermique

- dû à l'agitation naturelle des électrons, qui augmente avec la température du capteur
- peut être réduit efficacement par refroidissement du capteur
- Sur les appareils photo grand public, : filtre infrarouge juste devant le capteur

Bruit « Poivre et sel »

Pixels noirs et blancs répartis au hasard

Bruit « Poivre et sel »

Pixels noirs et blancs répartis au hasard

- erreurs de transmission de données

Bruit « Poivre et sel »

Pixels noirs et blancs répartis au hasard

- erreurs de transmission de données
- défaillance d'éléments de capteur

Bruit « Poivre et sel »

Pixels noirs et blancs répartis au hasard

- erreurs de transmission de données
- défaillance d'éléments de capteur
- particules fines (poussières) sur le capteur

Bruit de grenaille

- bruit électronique

Bruit de grenaille

- bruit électronique
- fluctuations statistiques perceptibles (éclairage faible)

Bruit de grenaille

- bruit électronique
- fluctuations statistiques perceptibles (éclairage faible)
- tend à modéliser le bruit d'acquisition

Bruit de grenaille

- bruit électronique
- fluctuations statistiques perceptibles (éclairage faible)
- tend à modéliser le bruit d'acquisition

Principale source de bruit dans les images prises par les appareils photo numériques actuels

Modélisation du bruit

Définition

On appelle « rapport signal sur bruit » (Signal to Noise Ratio : SNR), d'un signal donné x entaché d'un bruit b sur un intervalle de temps fini I la quantité :

$$SNR = 20 \cdot \log_{10}\left(\frac{\|x\|}{\|b\|}\right)$$

où $\|\cdot\|$ désigne la norme euclidienne du signal discret.

Modélisation du bruit

Définition

On appelle « rapport signal sur bruit » (Signal to Noise Ratio : SNR), d'un signal donné x entaché d'un bruit b sur un intervalle de temps fini I la quantité :

$$SNR = 20 \cdot \log_{10}\left(\frac{\|x\|}{\|b\|}\right)$$

où $\|\cdot\|$ désigne la norme euclidienne du signal discret.

- Il est exprimé en décibels (dB)

Définition

On appelle « rapport signal sur bruit » (Signal to Noise Ratio : SNR), d'un signal donné x entaché d'un bruit b sur un intervalle de temps fini I la quantité :

$$SNR = 20 \cdot \log_{10}\left(\frac{\|x\|}{\|b\|}\right)$$

où $\|\cdot\|$ désigne la norme euclidienne du signal discret.

- Il est exprimé en décibels (dB)
- Souvent obtenu expérimentalement

Modélisation du bruit

Le bruit est de nature et d'origine variées et donc se modélise de différentes façons :

Modélisation du bruit

Le bruit est de nature et d'origine variées et donc se modélise de différentes façons :

1. **Bruit additif gaussien d'écart-type** σ

Modélisation du bruit

Le bruit est de nature et d'origine variées et donc se modélise de différentes façons :

1. **Bruit additif gaussien d'écart-type σ**
   - si u représente l'image originale, l'image dégradée u_b s'obtient par :
     $$u_b = u + b_\sigma$$
     où b_σ suit la loi normale centrée d'écart-type σ

Modélisation du bruit

Le bruit est de nature et d'origine variées et donc se modélise de différentes façons :

1. **Bruit additif gaussien d'écart-type σ**
   - si u représente l'image originale, l'image dégradée u_b s'obtient par :
     $$u_b = u + b_\sigma$$
     où b_σ suit la loi normale centrée d'écart-type σ
   - l'écart-type σ souvent égal au SNR

Modélisation du bruit

Le bruit est de nature et d'origine variées et donc se modélise de différentes façons :

1. **Bruit additif gaussien d'écart-type σ**
   - si u représente l'image originale, l'image dégradée u_b s'obtient par :
     $$u_b = u + b_\sigma$$
     où b_σ suit la loi normale centrée d'écart-type σ
   - l'écart-type σ souvent égal au SNR
   - C'est le modèle le plus courant.

Modélisation du bruit

2. **Bruit multiplicatif**

Modélisation du bruit

2. **Bruit multiplicatif**
   - L'image dégradée u_b s'obtient par :

$$u_b = u \cdot (1 + b_\sigma)$$

2. **Bruit multiplicatif**
   - L'image dégradée u_b s'obtient par :

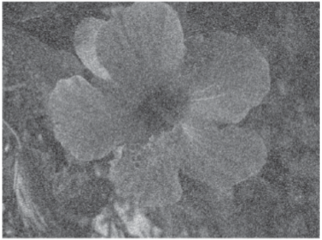
$$u_b = u \cdot (1 + b_\sigma)$$

   - bruit est proportionnel à l'intensité lumineuse

Exemples

(a) Image originale

(b) Bruit blanc gaussien $\sigma = 0.2$ ($\simeq$ 50 niveaux de gris)



(c) Bruit poivre et sel

(d) Bruit multiplicatif

Atténuation du bruit

1. **Traitement local** : Filtrage spatial

Atténuation du bruit

1. **Traitement local** : Filtrage spatial
   - Filtres linéaires : Produit de convolution

Atténuation du bruit

1. **Traitement local** : Filtrage spatial
    - Filtres linéaires : Produit de convolution
    - Filtres non linéaire

Atténuation du bruit

1. **Traitement local** : Filtrage spatial
   - Filtres linéaires : Produit de convolution
   - Filtres non linéaire
2. **traitement global** : Filtrage fréquentiel
   - Analyse de Fourier

Atténuation du bruit

1. **Traitement local** : Filtrage spatial
   - Filtres linéaires : Produit de convolution
   - Filtres non linéaire
2. **traitement global** : Filtrage fréquentiel
   - Analyse de Fourier

Le produit de convolution et le filtrage spatial servent aussi

- à la recherche des contours
- à la recherche des formes, etc

Il s'agit d'un des outils les plus importants du traitement des images

Sommaire

Produit de convolution 1D

Les techniques de traitement d'images sont adaptées de la théorie du traitement du signal

Produit de convolution 1D

Les techniques de traitement d'images sont adaptées de la théorie du traitement du signal

Application d'un filtre à un signal $x(t)$:

$$x(t) \rightarrow \boxed{\text{filtre}} \rightarrow y(t)$$

Produit de convolution 1D

Les techniques de traitement d'images sont adaptées de la théorie du traitement du signal

Application d'un filtre à un signal $x(t)$:

$$x(t) \rightarrow \boxed{\text{filtre}} \rightarrow y(t)$$

- Comment sont liés signaux d'entrée et de sortie ?

Produit de convolution 1D

Les techniques de traitement d'images sont adaptées de la théorie du traitement du signal

Application d'un filtre à un signal $x(t)$:

$$x(t) \rightarrow \boxed{\text{filtre}} \rightarrow y(t)$$

- Comment sont liés signaux d'entrée et de sortie ?
- Une grande classe de filtres sont les *filtres linéaires*

Filtres linéaires

Définition

Un filtre linéaire est formé d'un opérateur linéaire continu qui vérifie les deux propriétés suivantes :

Définition

Un filtre linéaire est formé d'un opérateur linéaire continu qui vérifie les deux propriétés suivantes :

1. **Linéarité**
   Si $x_1(t) \rightarrow \boxed{\text{f.l.}} \rightarrow y_1(t)$ et $x_2(t) \rightarrow \boxed{\text{f.l.}} \rightarrow y_2(t)$ alors :
   $$ax_1(t) + bx_2(t) \rightarrow \boxed{\text{f.l.}} \rightarrow ay_1(t) + by_2(t)$$

Définition

Un filtre linéaire est formé d'un opérateur linéaire continu qui vérifie les deux propriétés suivantes :

1. **Linéarité**
   Si $x_1(t) \to \boxed{\text{f.l.}} \to y_1(t)$ et $x_2(t) \to \boxed{\text{f.l.}} \to y_2(t)$ alors :
   $$ax_1(t) + bx_2(t) \to \boxed{\text{f.l.}} \to ay_1(t) + by_2(t)$$
2. **Invariance temporelle**
   Si $x_1(t) \to \boxed{\text{f.l.}} \to y_1(t)$ alors :
   $$x_1(t-\tau) \to \boxed{\text{f.l.}} \to y_1(t-\tau)$$

Produit de convolution 1D

Théorème (Produit de convolution)

Dans le cas d'un filtre linéaire :

$$x(t) \to \boxed{\text{f.l.}} \to y(t)$$

Alors :

$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t-\tau)\,\mathrm{d}\tau$$

Produit de convolution 1D

Théorème (Produit de convolution)

Dans le cas d'un filtre linéaire :

$$x(t) \rightarrow \boxed{\text{f.l.}} \rightarrow y(t)$$

Alors :

$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t - \tau) \, \mathrm{d}\tau$$

- *Produit de convolution de x et de h*

Produit de convolution 1D

Théorème (Produit de convolution)

Dans le cas d'un filtre linéaire :

$$x(t) \to \boxed{\text{f.l.}} \to y(t)$$

Alors :

$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t - \tau) \, \mathrm{d}\tau$$

- *Produit de convolution de x et de h*
- *On note $[x * h](t)$ ou $x(t) * h(t)$*

Produit de convolution 1D

Théorème (Produit de convolution)

Dans le cas d'un filtre linéaire :

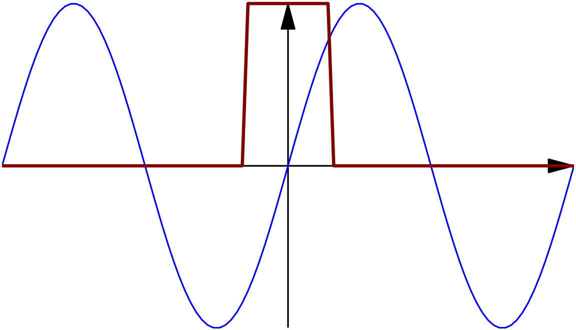
$$x(t) \rightarrow \boxed{\text{f.l.}} \rightarrow y(t)$$

Alors :

$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t-\tau)\,\mathrm{d}\tau$$

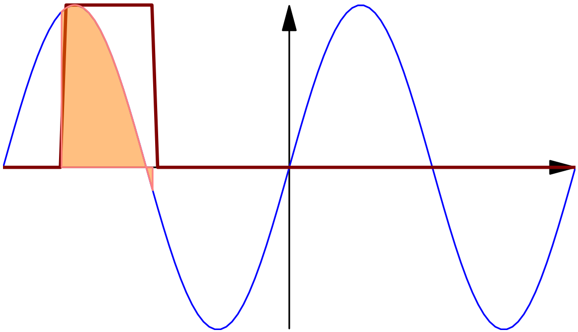
- *Produit de convolution de x et de h*
- *On note $[x * h](t)$ ou $x(t) * h(t)$*
- *h est la réponse impulsionelle du filtre*
- *h caractérise le filtre*

Produit de convolution 1D



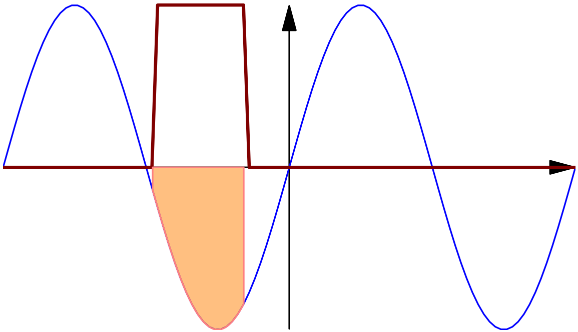
$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t-\tau)\,\mathrm{d}\tau$$

Produit de convolution 1D



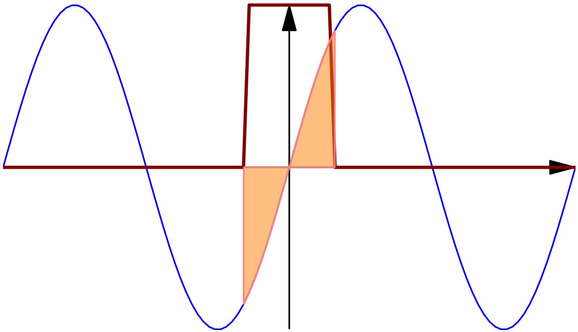
$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t-\tau)\,\mathrm{d}\tau \qquad \tau = -4$$

Produit de convolution 1D



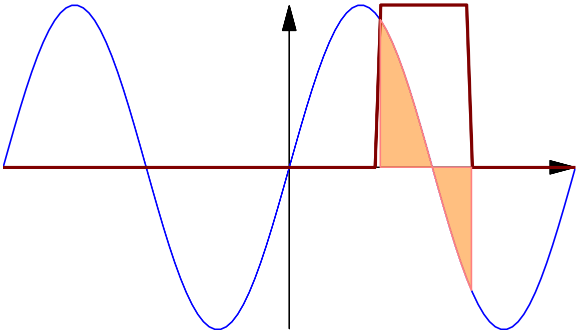
$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t-\tau)\,\mathrm{d}\tau \qquad \tau = -2$$

Produit de convolution 1D



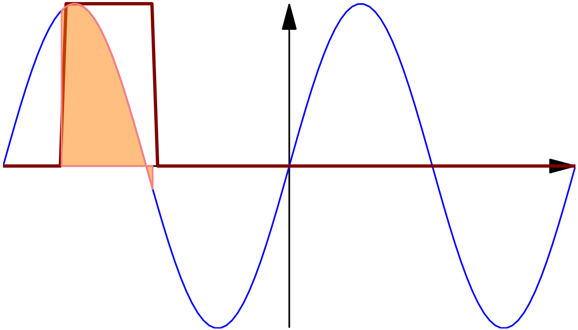
$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t-\tau)\,\mathrm{d}\tau \qquad \tau = 0$$

Produit de convolution 1D



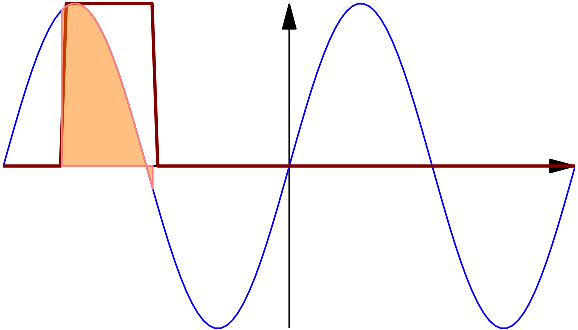
$$y(t) = \int_{-\infty}^{+\infty} x(\tau) \cdot h(t - \tau) \, \mathrm{d}\tau \qquad \tau = 3$$

Produit de convolution 1D



À chaque étape, cela revient à

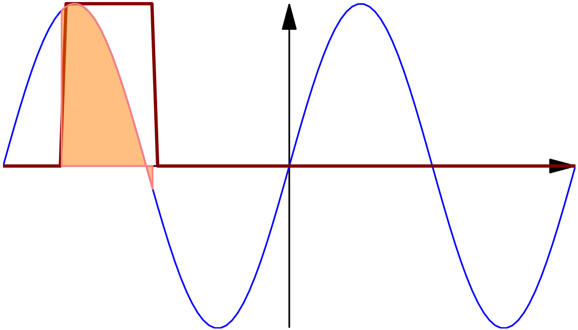
Produit de convolution 1D



À chaque étape, cela revient à

- Calculer la moyenne du signal $x(t)$ sur un intervalle de longueur 1

Produit de convolution 1D



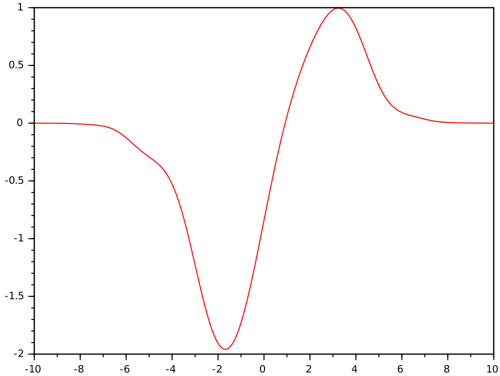
À chaque étape, cela revient à

- Calculer la moyenne du signal $x(t)$ sur un intervalle de longueur 1
- Lissage de la $x(t)$

Produit de convolution 1D

Un signal :

```
y0=(x-sin(0.2*x^2+1)).*exp(-x.^2/9);
```



Produit de convolution 1D

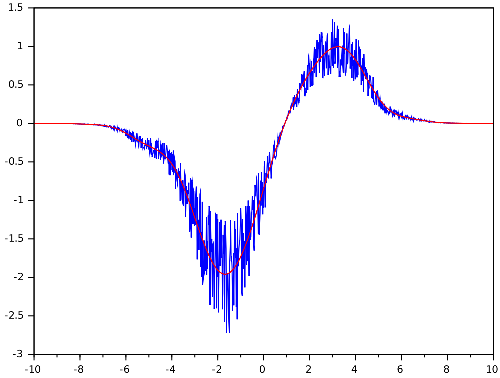
On le bruite :

```
y = y0.*(1+0.4*grand(y0,'unf',-1,1));
```

Produit de convolution 1D

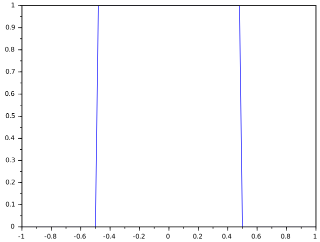
On le bruite :

```
y = y0.*(1+0.4*grand(y0,'unf',-1,1));
```



Produit de convolution 1D

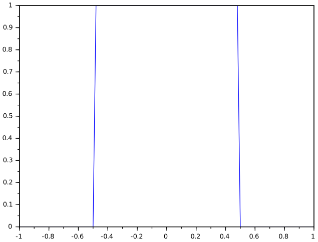
Filtrage par la fonction *rectangle* h_1 :

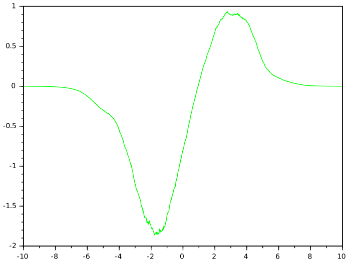



```
y1 = conv(h1, y, 'same');
plot(x, y1)
```

Produit de convolution 1D

Filtrage par la fonction *rectangle* h_1 :

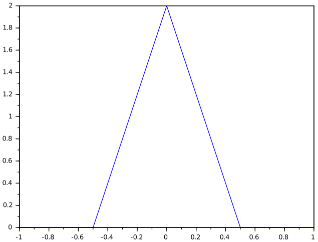


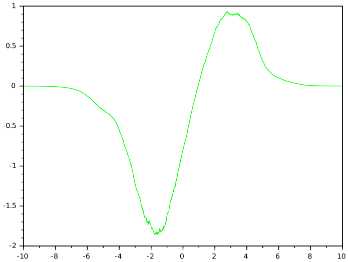



```
y1 = conv(h1, y, 'same');
plot(x, y1)
```

Produit de convolution 1D

Filtrage par la fonction *triangle* h_2 :

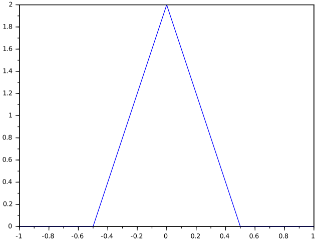


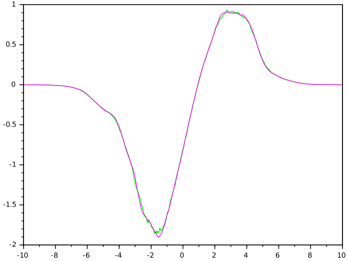



```
y1 = conv(h2, y, 'same');
plot(x, y2)
```

Produit de convolution 1D

Filtrage par la fonction *triangle* h_2 :

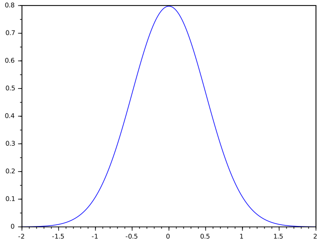


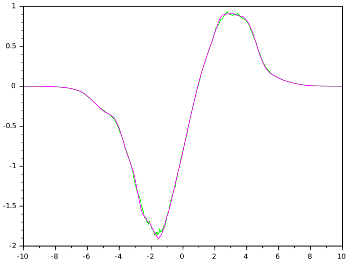



```
y1 = conv(h2, y, 'same');
plot(x, y2)
```

Produit de convolution 1D

Filtrage par une fonction *gaussienne* h_3 :

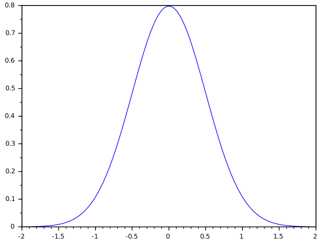


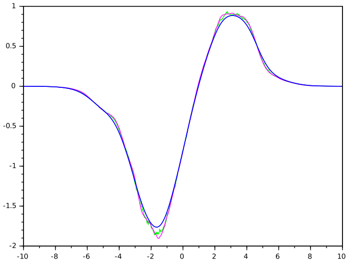



```
y3 = conv(h3, y, 'same');
plot(x, y3)
```

Produit de convolution 1D

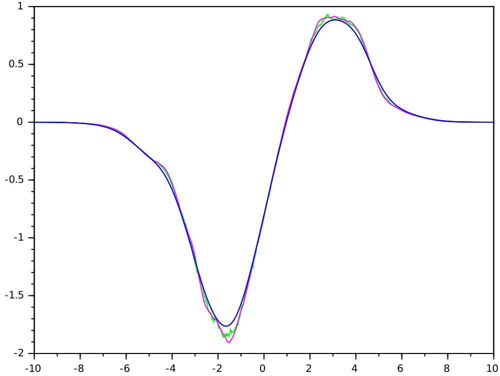
Filtrage par une fonction *gaussienne* h_3 :



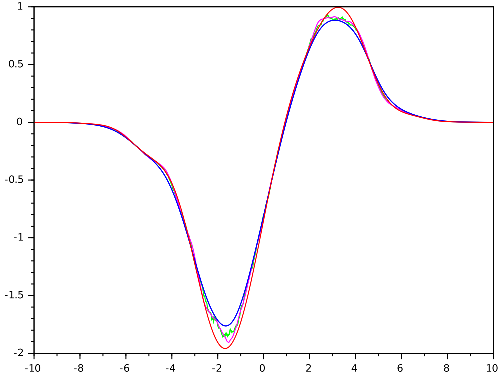



```
y3 = conv(h3, y, 'same');
plot(x, y3)
```

Produit de convolution 1D



Produit de convolution 1D



Produit de convolution 1D

Quelques remarques :

Produit de convolution 1D

Quelques remarques :

1. dans la pratique, les filtres sont non nuls uniquement sur un intervalle du type $[a; b]$. Il n'y aura donc
   - pas de problèmes de calcul des intégrales
   - pas de problèmes d'existence de produit de convolution.

Produit de convolution 1D

Quelques remarques :

1. dans la pratique, les filtres sont non nuls uniquement sur un intervalle du type $[a; b]$. Il n'y aura donc
   - pas de problèmes de calcul des intégrales
   - pas de problèmes d'existence de produit de convolution.
2. dans la cas d'un signal échantillonné (à temps discret) :

$$y_n = (x * h)(n) = \sum_{k=-\infty}^{+\infty} x_k h_{n-k}$$

Produit de convolution 2D

Le produit de convolution se généralise en 2 dimensions :

$$(I * h)(x; y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I(x; y) h(u - x; v - y) \, \mathrm{d}u \, \mathrm{d}v$$

Produit de convolution 2D

Le produit de convolution se généralise en 2 dimensions :

$$(I * h)(x; y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I(x; y) h(u - x; v - y) \, \mathrm{d}u \, \mathrm{d}v$$

Dans le cas discret :

$$(I * h)(x; y) = \sum_{-\infty}^{+\infty} \sum_{-\infty}^{+\infty} I(x; y) h(u - x; v - y)$$

Produit de convolution 2D

Une image est vue comme un signal 2D :

$$I(x; y) \rightarrow \boxed{\text{filtre linéaire } h} \rightarrow I_2(x; y)$$

$h(x; y)$ est appelé masque de convolution, noyau de convolution, filtre, fenêtre, kernel ...

Produit de convolution 2D

Une image est vue comme un signal 2D :

$$I(x;y) \rightarrow \boxed{\text{filtre linéaire } h} \rightarrow I_2(x;y)$$

$h(x;y)$ est appelé masque de convolution, noyau de convolution, filtre, fenêtre, kernel ...

$$I_2(x;y) = I_1 * h(x;y)$$

Produit de convolution 2D

Dans la pratique :

- noyaux dont le support est petit

Produit de convolution 2D

Dans la pratique :

- noyaux dont le support est petit
- convolution locale au voisinage de chaque pixel

Produit de convolution 2D

Dans la pratique :

- noyaux dont le support est petit
- convolution locale au voisinage de chaque pixel

Un filtre de convolution est une matrice généralement (mais pas toujours) de taille impaire et symétrique

- 3×3, 5×5, 7×7, ...

Produit de convolution 2D

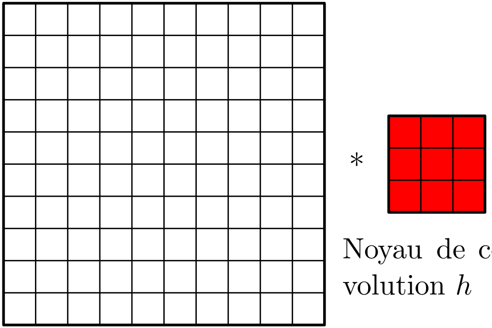
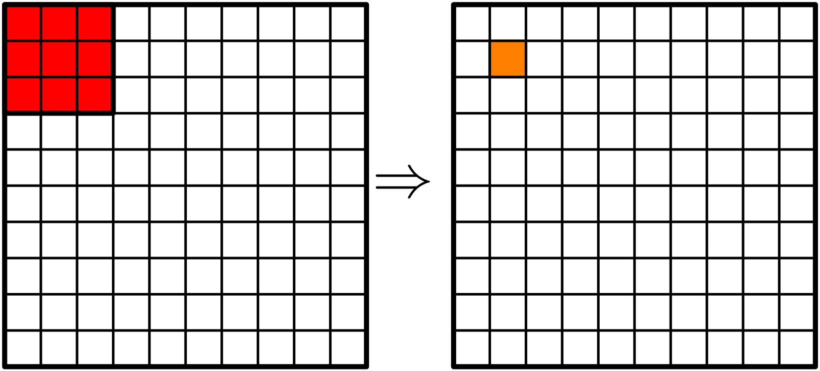


Image I

Noyau de convolution h

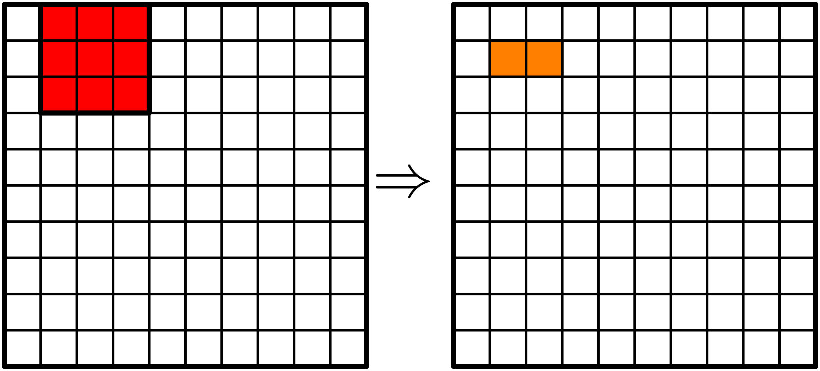
$$I'(x;y) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} I(x+i;y+j) \times h(i;j)$$

Produit de convolution 2D



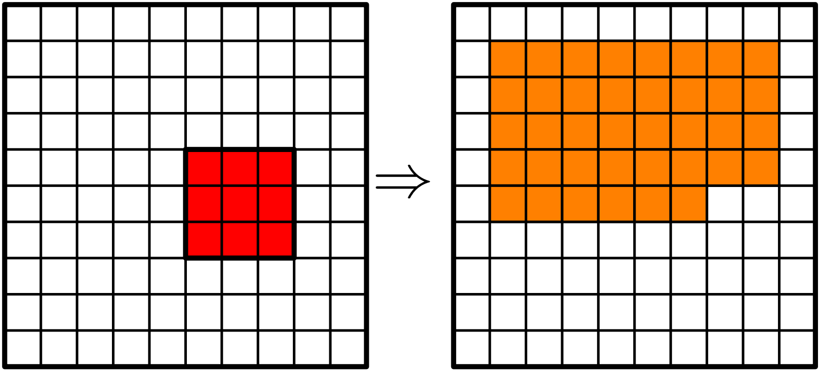
$$I'(1;1) = I(0;0)h(-1;-1) + I(1,0)h(0,-1) + I(2,0)h(1,-1) \\ + \cdots + I(2;2)h(1,1)$$

Produit de convolution 2D



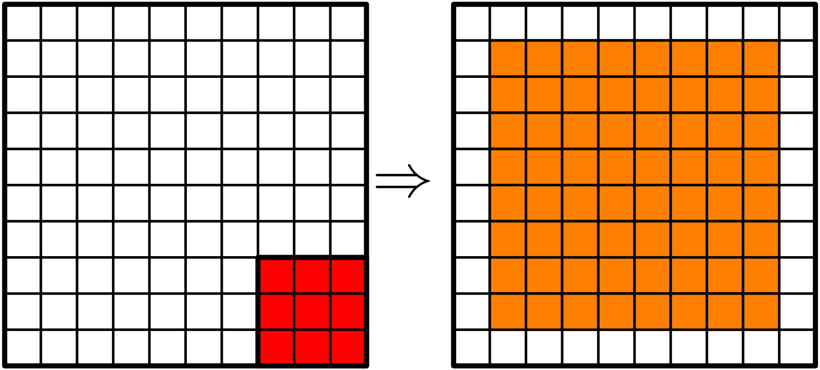
$$I'(2;1) = I(1;0)h(-1;-1) + I(2,0)h(0,-1) + I(3,0)h(1,-1) \\ + \cdots + I(3;2)h(1,1)$$

Produit de convolution 2D

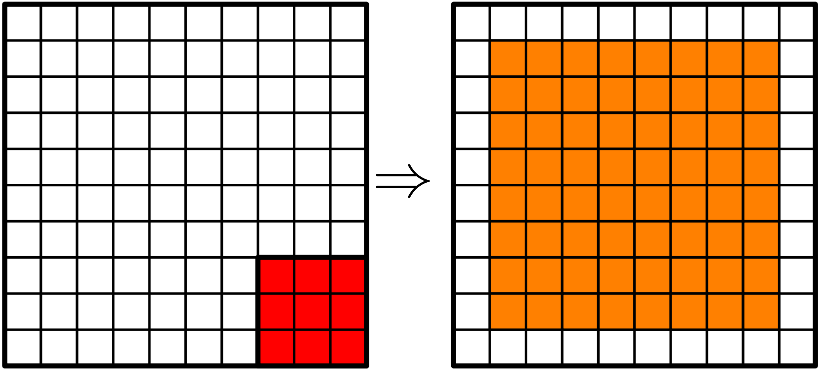


$$I'(i;j) = I(i-1;j-1)h(-1;-1) + I(i,j-1)h(0,-1) + \cdots + I(i+1;j+1)h(1,1)$$

Produit de convolution 2D



Produit de convolution 2D



Que faire des bords ?

Que faire des bords ?

Produit de convolution 2D

Que faire des bords ?

- Mettre à 0

Produit de convolution 2D

Que faire des bords ?

- Mettre à 0
- Convolution partielle

Produit de convolution 2D

Que faire des bords ?

- Mettre à 0
- Convolution partielle
- Miroir de l'image

Produit de convolution 2D

Deux types pour le filtrage spatial

Produit de convolution 2D

Deux types pour le filtrage spatial

1. Filtres passe-bas

Produit de convolution 2D

Deux types pour le filtrage spatial

1. Filtres passe-bas
   - Atténue le bruit et les détails

Produit de convolution 2D

Deux types pour le filtrage spatial

1. Filtres passe-bas
   - Atténue le bruit et les détails
   - lissage

Produit de convolution 2D

Deux types pour le filtrage spatial

1. Filtres passe-bas
   - Atténue le bruit et les détails
   - lissage
2. Filtres passe-haut

Deux types pour le filtrage spatial

1. Filtres passe-bas
   - Atténue le bruit et les détails
   - lissage
2. Filtres passe-haut
   - Accentue les détails et les contours

Produit de convolution 2D

Deux types pour le filtrage spatial

1. Filtres passe-bas
   - Atténue le bruit et les détails
   - lissage
2. Filtres passe-haut
   - Accentue les détails et les contours

Remarque

la somme des éléments du filtre est normalisé à 1

- conservation la moyenne de l'image (luminosité)

Filtre séparable

Définition

Un filtre 2D est dit séparable s'il est possible de décomposer le noyau de convolution h en

- un filtre horizontal
- suivi d'un filtre vertical

Filtre séparable

Définition

Un filtre 2D est dit séparable s'il est possible de décomposer le noyau de convolution h en

- un filtre horizontal
- suivi d'un filtre vertical

On peut alors traiter séparément les lignes et les colonnes de l'image ce qui est un gros avantage pour l'implémentation.

Filtre séparable

Théorème

un filtre h est séparable si :

Filtre séparable

Théorème

un filtre h est séparable si :

1. *sa matrice est le produit d'une matrice colonne par une matrice ligne*

Filtre séparable

Théorème

un filtre h est séparable si :

1. *sa matrice est le produit d'une matrice colonne par une matrice ligne*
2. *les lignes de la matrice de h sont proportionnelles engtre elles ainsi que ses colonnes*

Filtre séparable

Théorème

un filtre h est séparable si :

1. *sa matrice est le produit d'une matrice colonne par une matrice ligne*
2. *les lignes de la matrice de h sont proportionnelles engtre elles ainsi que ses colonnes*

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} \times \begin{pmatrix} a & b & c \end{pmatrix} = \begin{pmatrix} a\alpha & b\alpha & c\alpha \\ a\beta & b\beta & c\beta \\ a\gamma & b\gamma & c\gamma \end{pmatrix}$$

Sommaire

Filtre de moyenne

Définition

- Le filtre de moyenne est un filtre linéaire dont tous les coefficients sont égaux

Filtre de moyenne

Définition

- Le filtre de moyenne est un filtre linéaire dont tous les coefficients sont égaux
- Filtre de moyenne de dimension 3×3 :

$$\frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Filtre de moyenne

Définition

- Le filtre de moyenne est un filtre linéaire dont tous les coefficients sont égaux
- Filtre de moyenne de dimension 3×3 :

$$\frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

- Filtre de moyenne de dimension 5×5 :

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

Filtre de moyenne

- Filtre linéaire ;

Filtre de moyenne

- Filtre linéaire ;
- Filtre séparable ;

Filtre de moyenne

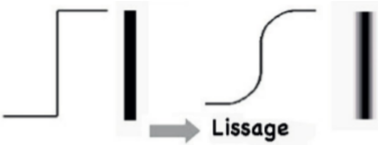
- Filtre linéaire ;
- Filtre séparable ;
- remplace la valeur d'une pixel par la moyenne des valeurs des pixels dans un certain voisinage

Filtre de moyenne

- Filtre linéaire ;
- Filtre séparable ;
- remplace la valeur d'une pixel par la moyenne des valeurs des pixels dans un certain voisinage
- Efficace pour les bruits additifs gaussiens

Filtre de moyenne

- Filtre linéaire ;
- Filtre séparable ;
- remplace la valeur d'une pixel par la moyenne des valeurs des pixels dans un certain voisinage
- Efficace pour les bruits additifs gaussiens
- Problème de floutage de contours :



Filtre gaussien

Définition

Un filtre gaussien est donné par discrétisation de la fonction gaussienne :

$$f(x; y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

sur un voisinage de $(0; 0)$

Ici σ est l'écart-type et la moyenne est nulle.

Filtre gaussien

Le filtre Gaussien 3×3 est défini par la matrice :

$$\begin{pmatrix} G(-1;-1) & G(0;-1) & G(1;-1) \\ G(-1;0) & G(0;0) & G(1;0) \\ G(-1;1) & G(0;1) & G(1;1) \end{pmatrix}$$

Que l'on normalise

Filtre gaussien

Le filtre Gaussien 3×3 est défini par la matrice :

$$\begin{pmatrix} G(-1;-1) & G(0;-1) & G(1;-1) \\ G(-1;0) & G(0;0) & G(1;0) \\ G(-1;1) & G(0;1) & G(1;1) \end{pmatrix}$$

Que l'on normalise
Pour $\sigma = 0,8$, on obtient environ :

$$\frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

Filtre gaussien

Le filtre gaussien de dimension 5×5 pour $\sigma = 1$ est environ :

$$\frac{1}{300}\begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 18 & 30 & 18 & 4 \\ 6 & 30 & 48 & 30 & 6 \\ 4 & 18 & 30 & 18 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix}$$

Filtre gaussien

Le filtre gaussien de dimension 5×5 pour $\sigma = 1$ est environ :

$$\frac{1}{300}\begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 18 & 30 & 18 & 4 \\ 6 & 30 & 48 & 30 & 6 \\ 4 & 18 & 30 & 18 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix}$$

- Filtre linéaire

Filtre gaussien

Le filtre gaussien de dimension 5×5 pour $\sigma = 1$ est environ :

$$\frac{1}{300}\begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 18 & 30 & 18 & 4 \\ 6 & 30 & 48 & 30 & 6 \\ 4 & 18 & 30 & 18 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix}$$

- Filtre linéaire
- Détails atténués (donc le bruit additif)

Filtre gaussien

Le filtre gaussien de dimension 5×5 pour $\sigma = 1$ est environ :

$$\frac{1}{300}\begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 18 & 30 & 18 & 4 \\ 6 & 30 & 48 & 30 & 6 \\ 4 & 18 & 30 & 18 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix}$$

- Filtre linéaire
- Détails atténués (donc le bruit additif)
- Contours érodés

Filtre médian

- Inconvénient du filtre de moyenne :

  sensible aux valeurs extrêmes

Filtre médian

- Inconvénient du filtre de moyenne :

  sensible aux valeurs extrêmes

- Lorsque l'on a des valeurs nulles ou égales à 255, on préfère prendre la médiane du voisinage plutôt que la moyenne.

Filtre médian

- Inconvénient du filtre de moyenne :

  sensible aux valeurs extrêmes

- Lorsque l'on a des valeurs nulles ou égales à 255, on préfère prendre la médiane du voisinage plutôt que la moyenne.

C'est le filtre médian !

Filtre médian

Exemple

Soit le voisinage :

$$\begin{pmatrix} 30 & 10 & 20 \\ 10 & 250 & 25 \\ 20 & 25 & 30 \end{pmatrix}$$

Filtre médian

Exemple

Soit le voisinage :

$$\begin{pmatrix} 30 & 10 & 20 \\ 10 & 250 & 25 \\ 20 & 25 & 30 \end{pmatrix}$$

- Valeurs dans l'ordre croissant :

$$10 \quad 10 \quad 20 \quad 20 \qquad \underset{\substack{\uparrow \\ \text{médiane}}}{25} \qquad 25 \quad 30 \quad 30 \quad \overset{\substack{\text{bruit} \\ \downarrow}}{250}$$

Sommaire

Inversion de la transformée de Fourier discrète

Soit $G(m;n) = TF(g(m;n))$ alors :

$$g(m;n) = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} G(i;j)\, \mathrm{e}^{2\,\mathrm{i}\,\pi\left(\frac{mi}{M} + \frac{nj}{N}\right)}$$

Inversion de la transformée de Fourier discrète

Soit $G(m;n) = TF(g(m;n))$ alors :

$$g(m;n) = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} G(i;j)\, \mathrm{e}^{2\,\mathrm{i}\,\pi\left(\frac{mi}{M} + \frac{nj}{N}\right)}$$

- Cela permet de reconstruire une image à partir d'un spectre complexe

Inversion de la transformée de Fourier discrète

Soit $G(m;n) = TF(g(m;n))$ alors :

$$g(m;n) = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} G(i;j)\, \mathrm{e}^{2\,\mathrm{i}\,\pi\left(\frac{mi}{M} + \frac{nj}{N}\right)}$$

- Cela permet de reconstruire une image à partir d'un spectre complexe
- Une image et son spectre complexe sont équivalentes
  - aucune perte d'information

Lien avec la convolution

- **Transformée d'un produit** :

$$\boxed{TF\left[x(t) \times y(t)\right] = X(f) * Y(f)}$$

Lien avec la convolution

- **Transformée d'un produit** :

$$\boxed{TF\left[x(t) \times y(t)\right] = X(f) * Y(f)}$$

- **Transformée d'une convolution** :

$$\boxed{TF\left[x(t) * y(t)\right] = X(f) \times Y(f)}$$

Lien avec la convolution

- **Transformée d'un produit** :

$$\boxed{TF\left[x(t) \times y(t)\right] = X(f) * Y(f)}$$

- **Transformée d'une convolution** :

$$\boxed{TF\left[x(t) * y(t)\right] = X(f) \times Y(f)}$$

- On a la diagramme suivant :

$$\begin{array}{ccc} I & \rightarrow & I * h \\ \downarrow & & \uparrow \\ TF(I) & \rightarrow & TF(I) \times H \end{array} \qquad \text{où } H = TF(h)$$

Lien avec la convolution

$$\begin{tikzcd} I \arrow[r] \arrow[d] & I * h \\ TF(I) \arrow[r] & TF(I) \times H \arrow[u] \end{tikzcd} \quad \text{où } H = TF(h)$$

Lien avec la convolution

$$\begin{array}{ccc} I & \rightarrow & I * h \\ \downarrow & & \uparrow \\ TF(I) & \rightarrow & TF(I) \times H \end{array} \qquad \text{où } H = TF(h)$$

- Dans le domaine spatial, le filtrage se fait par convolution.

Lien avec la convolution

$$\begin{array}{ccc} I & \rightarrow & I * h \\ \downarrow & & \uparrow \\ TF(I) & \rightarrow & TF(I) \times H \end{array} \qquad \text{où } H = TF(h)$$

- Dans le domaine spatial, le filtrage se fait par convolution.
- Dans le domaine spectral (ou fréquentiel), il se fait par multiplication
  - masquage de l'image

Lien avec la convolution

$$\begin{array}{ccc} I & \rightarrow & I * h \\ \downarrow & & \uparrow \\ TF(I) & \rightarrow & TF(I) \times H \end{array} \qquad \text{où } H = TF(h)$$

- Dans le domaine spatial, le filtrage se fait par convolution.
- Dans le domaine spectral (ou fréquentiel), il se fait par multiplication
  - masquage de l'image
- H est la **fonction de transfert du filtre**

Filtrage fréquentiel

$$\begin{array}{ccc} \text{Domaine spatial} & & \text{Domaine fréquentiel} \\ \boxed{\text{Image originale}} & \Rightarrow & \boxed{\text{Image transformée}} \\ \Downarrow & & \Uparrow \\ \boxed{\text{Image spectre}} & \Rightarrow & \boxed{\text{Image spectre transformée}} \end{array}$$

Filtrage fréquentiel

- **Passe-bas** :

Filtrage fréquentiel

- **Passe-bas** :
  - Mise en évidence les zones homogènes dans l'image
  - peu de variation des niveaux
  - élimination des hautes fréquences

Filtrage fréquentiel

- **Passe-bas** :
  - Mise en évidence les zones homogènes dans l'image
  - peu de variation des niveaux
  - élimination des hautes fréquences
- **Passe-haut** :

Filtrage fréquentiel

- **Passe-bas** :
  - Mise en évidence les zones homogènes dans l'image
  - peu de variation des niveaux
  - élimination des hautes fréquences
- **Passe-haut** :
  - Mise en évidence les zones hétérogènes
  - variations locales importantes des nivaux
  - élimination des basses fréquences

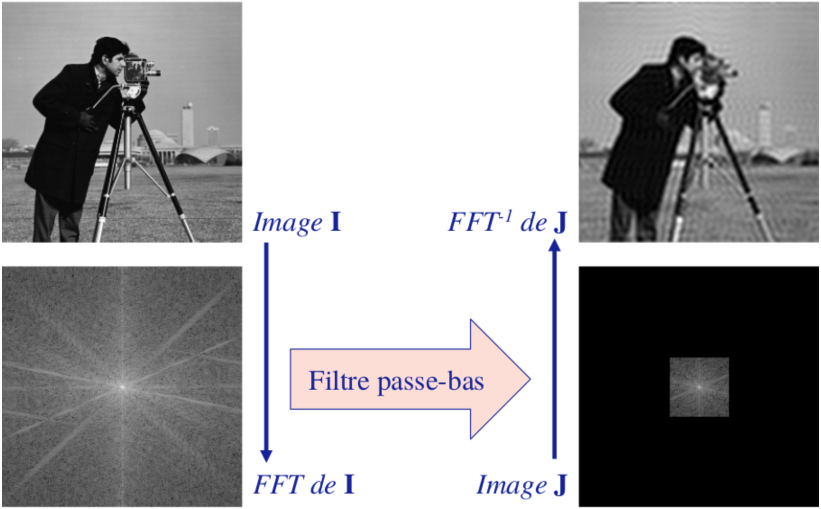
Filtrage fréquentiel

- **Passe-bas** :
  - Mise en évidence les zones homogènes dans l'image
  - peu de variation des niveaux
  - élimination des hautes fréquences
- **Passe-haut** :
  - Mise en évidence les zones hétérogènes
  - variations locales importantes des nivaux
  - élimination des basses fréquences
- **Passe-bande et coupe-bande** :

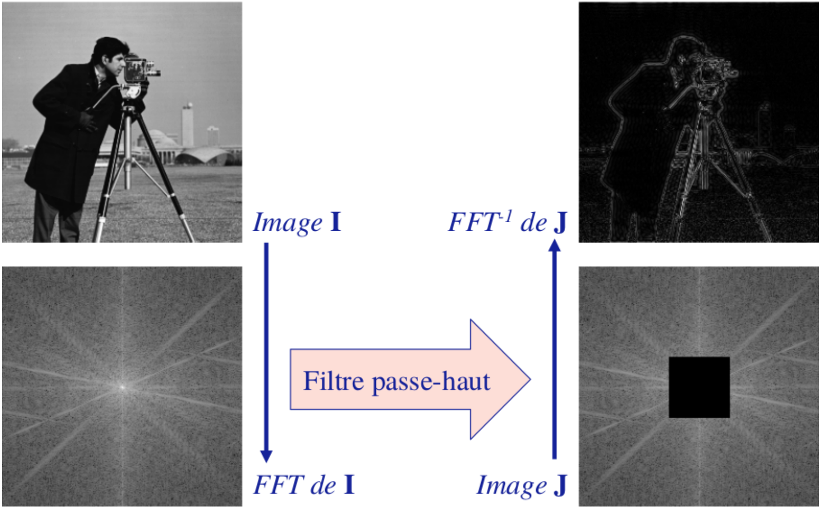
Filtrage fréquentiel

- **Passe-bas** :
  - Mise en évidence les zones homogènes dans l'image
  - peu de variation des niveaux
  - élimination des hautes fréquences
- **Passe-haut** :
  - Mise en évidence les zones hétérogènes
  - variations locales importantes des nivaux
  - élimination des basses fréquences
- **Passe-bande et coupe-bande** :
  - sélection d'une fréquence particulière.

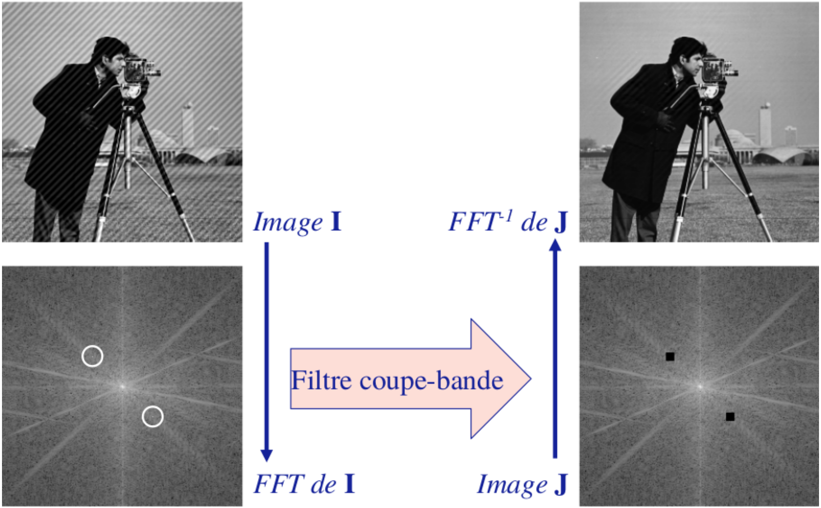
Filtre passe-bas



Filtre passe-haut



Filtre coupe-bande



Filtre passe-bande