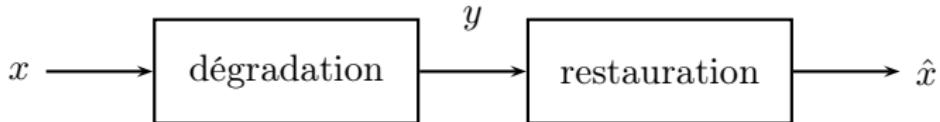


# **RESTAURATION**

# Restauration d'images

(image restoration)

Restauration = suppression des dégradations subies par l'image.

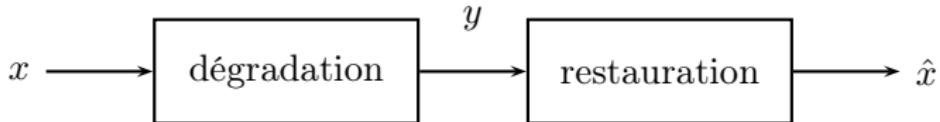


- $x$  est l'image inconnue à restaurer
- $y$  est l'image observée et dégradée
- $\hat{x}$  est l'image restaurée

# Restauration d'images

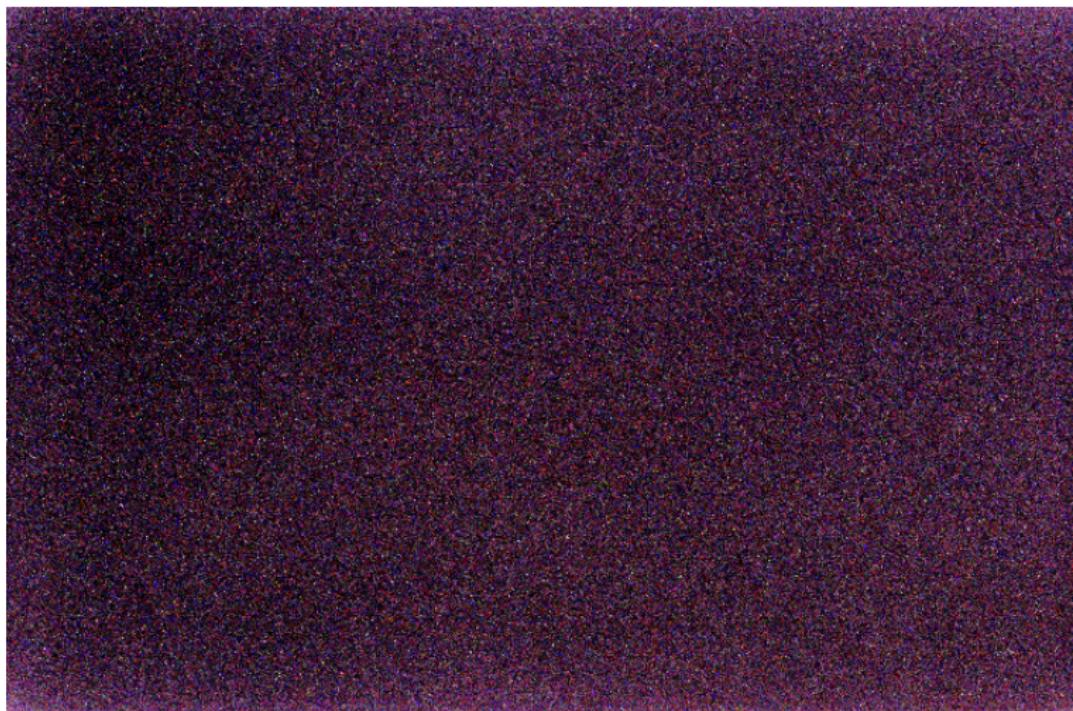
(image restoration)

Restauration = suppression des dégradations subies par l'image.

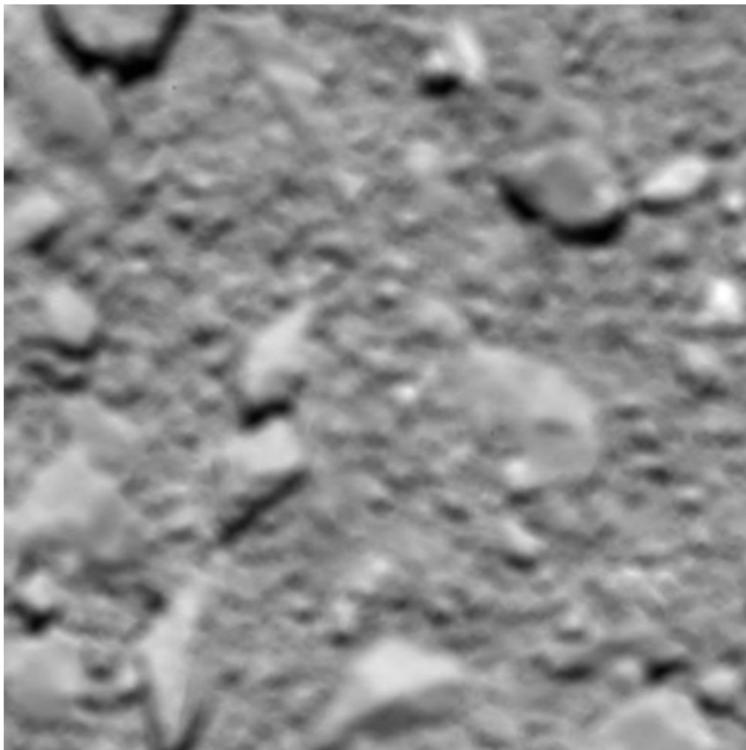


- $x$  est l'image inconnue à restaurer
- $y$  est l'image observée et dégradée
- $\hat{x}$  est l'image restaurée : **on souhaite que  $\hat{x} \approx x$ .**

# Débruitage



# Déconvolution



[ESA/Rosetta/MPS]

## *dehazing*



Image avec brume



Image traitée [Kopf et al. 2008]

# Correction des distortions



# Restauration d'images

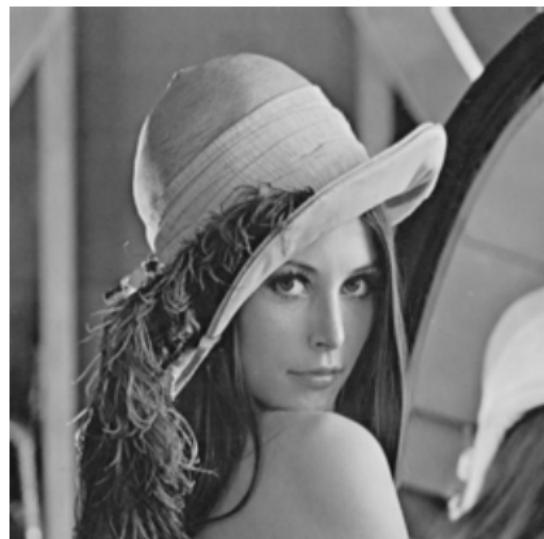
- Besoin d'un modèle de la dégradation
- « Inversion » de la dégradation  
⇒ la restauration est un *problème inverse*
- Formulation et optimisation d'un critère

# Sommaire

- Débruitage
- Déconvolution
- Mesure de la qualité

# Débruitage *(denoising)*

# Débruitage

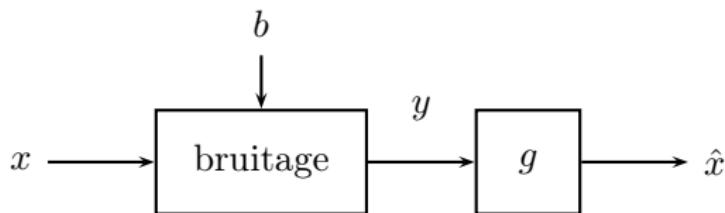


## Sources du bruit

Les principales sources de bruit dans les images numériques interviennent :

- durant l'acquisition (quantité de photons collectés trop faible, température du capteur...)
- durant la transmission (échos et distorsions atmosphériques en communication sans-fil)
- dans le modèle pour représenter les imprécisions du modèle de dégradation, forcément imparfait par rapport à la réalité!

# Modélisation



Les intensités des pixels du bruit  $b$  peuvent être considérées comme des variables aléatoires distribuées selon une densité de probabilité.

## Bruit additif gaussien indépendant (AWGN : additive white Gaussian noise)

L'intensité de chaque pixel  $(m,n)$  de l'observation  $y$  est en moyenne égale à l'intensité du pixel  $(m,n)$  de l'image non bruitée  $x$ :

$$\forall m,n \quad y(m,n) = x(m,n) + b(m,n) \quad \text{où} \quad b(m,n) \sim \mathcal{N}(0, \sigma^2)$$

Ce modèle est simple et facilite les calculs. Il est utilisé dans la plupart des applications, dont la photographie.

## Bruit de Poisson

(Poisson noise, shot noise)

Le nombre de photons mesurés par un photosite est aléatoire et dépend de l'illumination. Il est modélisé par un processus de Poisson dont la moyenne est égale à l'illumination.

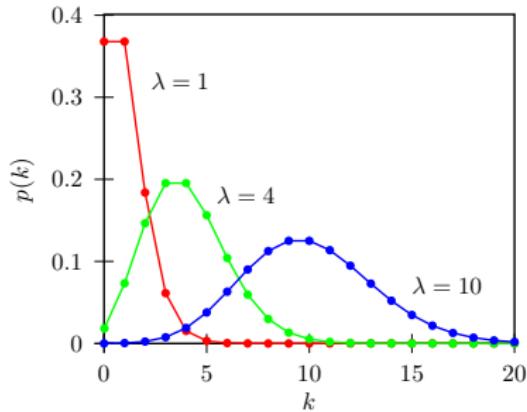
L'intensité de chaque pixel  $(m,n)$  de l'observation  $y$  est :

$$\forall m,n \quad y(m,n) \sim \mathcal{P}(x(m,n))$$

Ce modèle est utilisé dans le cas d'acquisitions à faible nombre de photons, par exemple en astronomie.

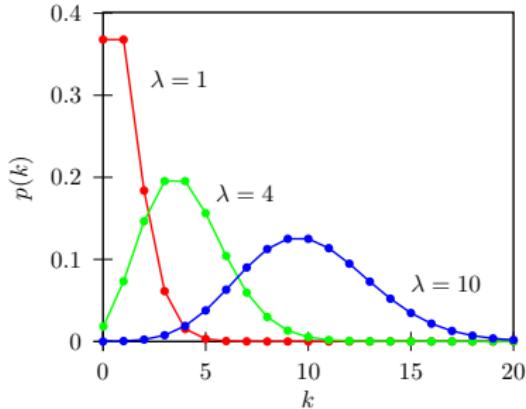
# Loi de Poisson $\mathcal{P}(\lambda)$

$$p(k) = \frac{\lambda^k}{k!} e^{-\lambda}$$



## Loi de Poisson $\mathcal{P}(\lambda)$

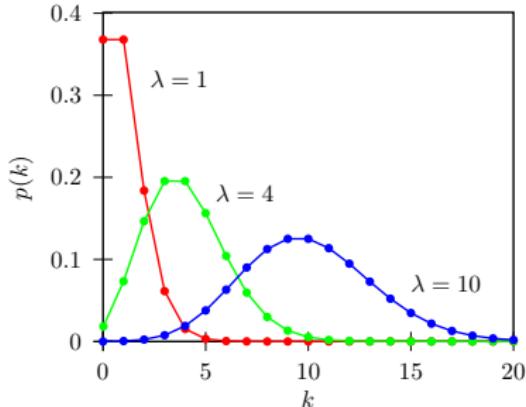
$$p(k) = \frac{\lambda^k}{k!} e^{-\lambda}$$



- $\mathbb{E}[X] = \lambda, \mathbb{V}[X] = \lambda$  : moyenne et variance dépendent du nombre de photons incidents  $\Rightarrow$  le bruit dépend du signal.

## Loi de Poisson $\mathcal{P}(\lambda)$

$$p(k) = \frac{\lambda^k}{k!} e^{-\lambda}$$



- $\mathbb{E}[X] = \lambda, \mathbb{V}[X] = \lambda$  : moyenne et variance dépendent du nombre de photons incidents  $\Rightarrow$  le bruit dépend du signal.
- Lorsque  $\lambda$  augmente, la loi de Poisson tend vers la loi normale  $\Rightarrow$  le bruit de Poisson peut être considéré comme un bruit gaussien additif s'il y a suffisamment de photons collectés.

## Bruit impulsionnel (sel & poivre)

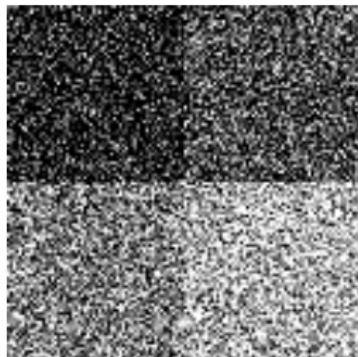
(salt-and-pepper noise)

Le bruit impulsionnel permet de modéliser le cas où certains pixels sont saturés ou morts (dysfonctionnement des photosites, saturation, etc.).

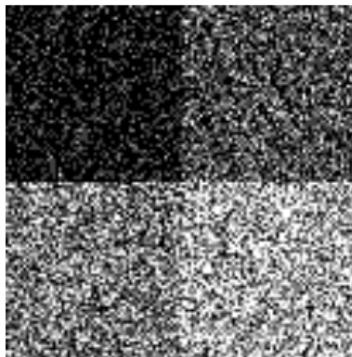
$$\forall m,n \quad y(m,n) = \begin{cases} x_{\min} & \text{avec une probabilité } p_{\min}, \\ x_{\max} & \text{avec une probabilité } p_{\max}, \\ x(m,n) & \text{avec une probabilité } 1 - p_{\min} - p_{\max}. \end{cases}$$

où  $x_{\min}$  et  $x_{\max}$  sont les valeurs minimales et maximales des intensités.

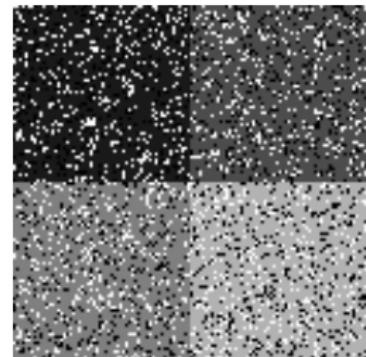
## Exemples de bruits



Bruit gaussien  
additif



Bruit de Poisson



Bruit impulsionnel

(Bruit de puissance sensiblement identique dans les trois images)

# Rapport signal-à-bruit (RSB)

(*SNR : signal-to-noise ratio*)

$$\text{RSB} = \frac{\text{puissance de l'image non bruitée}}{\text{puissance du bruit}}$$

# Rapport signal-à-bruit (RSB)

(SNR : signal-to-noise ratio)

$$\text{RSB} = \frac{\text{puissance de l'image non bruitée}}{\text{puissance du bruit}}$$

- Rappel : puissance de  $x = \frac{1}{M \times N} \sum_{m,n} x(m,n)^2$

# Rapport signal-à-bruit (RSB)

(SNR : signal-to-noise ratio)

$$\text{RSB} = \frac{\text{puissance de l'image non bruitée}}{\text{puissance du bruit}}$$

- Rappel : puissance de  $x = \frac{1}{M \times N} \sum_{m,n} x(m,n)^2$
- On travaille souvent en échelle logarithmique (unité : dB) :

$$\Rightarrow \text{RSB} = 10 \log_{10} \left( \frac{\sum_{m,n} x(m,n)^2}{\sum_{m,n} b(m,n)^2} \right)$$

## *Peak signal-to-noise ratio (PSNR)*

$$\text{PSNR} = \frac{\text{dynamique de l'image non bruitée}^2}{\text{puissance du bruit}}$$

où la dynamique est la différence entre l'intensité maximale et l'intensité minimale.

## *Peak signal-to-noise ratio (PSNR)*

$$\text{PSNR} = \frac{\text{dynamique de l'image non bruitée}^2}{\text{puissance du bruit}}$$

où la dynamique est la différence entre l'intensité maximale et l'intensité minimale.

- Formule définie dans le cas d'un bruit additif

## *Peak signal-to-noise ratio (PSNR)*

$$\text{PSNR} = \frac{\text{dynamique de l'image non bruitée}^2}{\text{puissance du bruit}}$$

où la dynamique est la différence entre l'intensité maximale et l'intensité minimale.

- Formule définie dans le cas d'un bruit additif

$$\text{PSNR} = 10 \log_{10} \left( \frac{\Delta f^2}{\frac{1}{M \times N} \sum_{m,n} b(m,n)^2} \right)$$

## RSB et PSNR



RSB = 0 dB



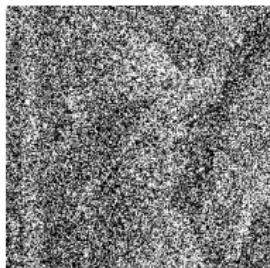
RSB = 10 dB



RSB = 20 dB



RSB = 30 dB



PSNR = 0 dB



PSNR = 10 dB



PSNR = 20 dB



PSNR = 30 dB

## RSB et PSNR



RSB = 0 dB



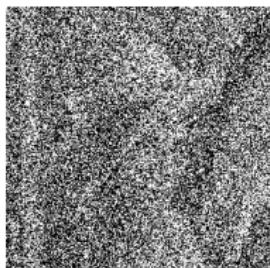
RSB = 10 dB



RSB = 20 dB



RSB = 30 dB



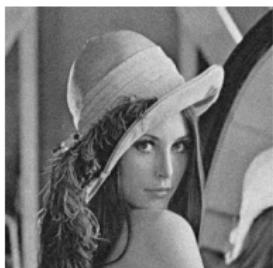
PSNR = 0 dB



PSNR = 10 dB



PSNR = 20 dB



PSNR = 30 dB

RSB/PSNR ↗ ⇒ bruit ↘

# État de l'art des méthodes de débruitage

- Filtres moyenneurs, médians...
- Filtrage fréquentiel de bruits périodiques
- Régularisation TV [Rudin et coll. 1992, Chambolle 2004]
- Ondelettes
- Approches statistiques
- Moyennes non locales [Buades et coll. 2005]
- ...

# État de l'art des méthodes de débruitage

- Filtres moyenneurs, médians...
- Filtrage fréquentiel de bruits périodiques
- Régularisation TV [Rudin et coll. 1992, Chambolle 2004]
- Ondelettes
- Approches statistiques
- Moyennes non locales [Buades et coll. 2005]
- ...

## Filtre moyenneur

- Méthode très simple!
- Chaque pixel  $(m, n)$  de l'image débruitée  $\hat{x}$  est la moyenne des pixels de l'image bruitée  $y$  autour de  $(m, n)$  :

$$\forall m, n \quad \hat{x}(m, n) = \frac{1}{|V_{m,n}|} \sum_{(u,v) \in V_{m,n}} y(u, v)$$

où

- $V_{m,n}$  est le *voisinage* (*neighborhood*) et définit quels sont les pixels autour de  $(m, n)$ ;
- $|V_{m,n}|$  est le cardinal de  $V_{m,n}$  : c'est le nombre de pixels dans le voisinage.

## Filtre moyenneur

Originale



Observation  
(RSB=20 dB)

Voisinage  
 $3 \times 3$



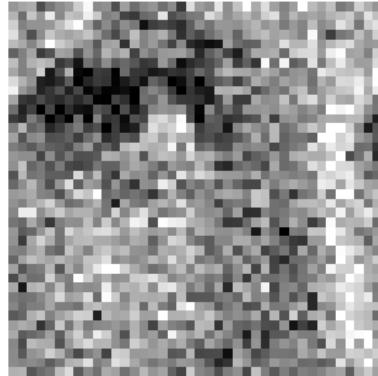
Voisinage  
 $7 \times 7$

## Filtre moyenneur

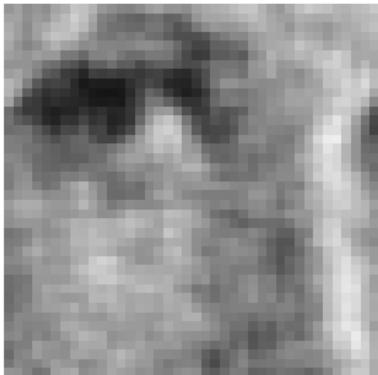
Originale



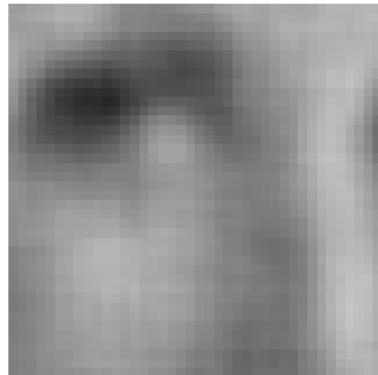
Observation  
(RSB=20 dB)



Voisinage  
 $3 \times 3$



Voisinage  
 $7 \times 7$



## Filtre moyenneur

Comment exprimer le filtre moyenneur avec une convolution ?



## Filtre moyenneur

Comment exprimer le filtre moyenneur avec une convolution ?

Dans le cas où le voisinage est un carré de taille  $N \times N$  pixels :

$$\hat{x}(m,n) = \frac{1}{N^2} \sum_{(u,v) \in V_{m,n}} y(u,v) = \sum_{u,v} g(m-u, n-v) y(u,v)$$

où 
$$g(u,v) = \begin{cases} \frac{1}{N^2} & \text{si } u \in \left\{-\frac{N}{2}, \dots, \frac{N}{2}\right\} \text{ et } v \in \left\{-\frac{N}{2}, \dots, \frac{N}{2}\right\} \\ 0 & \text{sinon} \end{cases}$$

## Filtre moyenneur

Comment exprimer le filtre moyenneur avec une convolution ?

Dans le cas où le voisinage est un carré de taille  $N \times N$  pixels :

$$\hat{x}(m,n) = \frac{1}{N^2} \sum_{(u,v) \in V_{m,n}} y(u,v) = \sum_{u,v} g(m-u, n-v) y(u,v)$$

où 
$$g(u,v) = \begin{cases} \frac{1}{N^2} & \text{si } u \in \left\{-\frac{N}{2}, \dots, \frac{N}{2}\right\} \text{ et } v \in \left\{-\frac{N}{2}, \dots, \frac{N}{2}\right\} \\ 0 & \text{sinon} \end{cases}$$

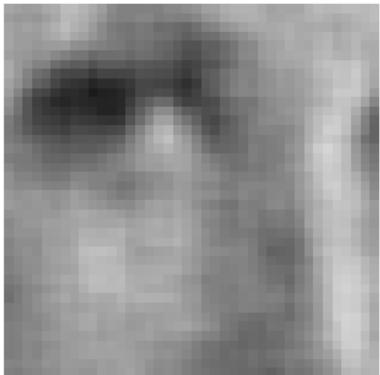
On peut étendre cette définition à tout type de noyau  $g$ !

## Filtre moyenneur

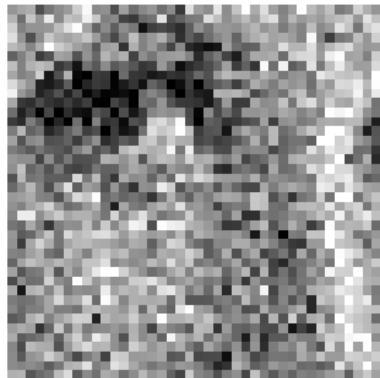
Originale



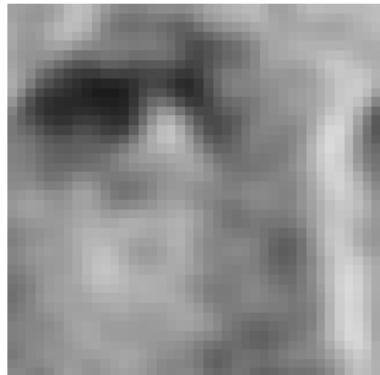
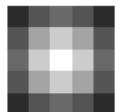
carré  
 $5 \times 5$



Observation  
(RSB=20 dB)



gaussienne  
 $5 \times 5$



## Filtre moyenneur

En résumé...

- le filtre moyenneur calcule la moyenne des pixels dans un voisinage
- le filtre moyenneur peut s'écrire comme une convolution
- lissage des variations ⇒ réduction du bruit par effet de flou

## Filtre médian

La médiane d'un ensemble de nombres est l'élément  $m$  de cet ensemble tel qu'il y a autant de nombres plus petit que  $m$  que de nombres plus grands que  $m$ .

Par exemple : Médiane( $\{1, 2, 4, 8, 16\}$ ) = 4.

## Filtre médian

La médiane d'un ensemble de nombres est l'élément  $m$  de cet ensemble tel qu'il y a autant de nombres plus petit que  $m$  que de nombres plus grands que  $m$ .

Par exemple :  $\text{Médiane}(\{1, 2, 4, 8, 16\}) = 4$ .

Filtre médian :

$$\forall m, n \quad \hat{x}(m, n) = \text{Médiane} \left( \{y(u, v) \mid (u, v) \in V_{m, n}\} \right)$$

## Filtre médian

La médiane d'un ensemble de nombres est l'élément  $m$  de cet ensemble tel qu'il y a autant de nombres plus petit que  $m$  que de nombres plus grands que  $m$ .

Par exemple :  $\text{Médiane}(\{1, 2, 4, 8, 16\}) = 4$ .

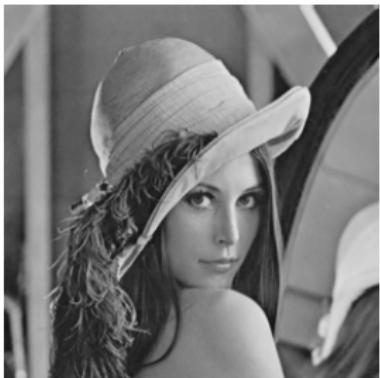
Filtre médian :

$$\forall m, n \quad \hat{x}(m, n) = \text{Médiane} \left( \{y(u, v) \mid (u, v) \in V_{m, n}\} \right)$$

- Malgré son nom, le filtre médian n'est pas un filtre car il ne respecte pas la propriété de linéarité : il ne peut donc pas s'écrire comme une convolution.
- Excellent dans le cas d'un bruit impulsif (car le résultat est moins flou qu'avec un filtre moyenneur).

## Filtre médian

Originale



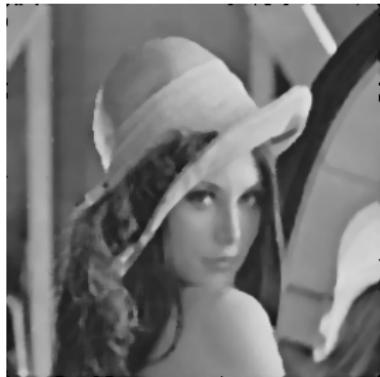
Observation



Moyenneur  
 $5 \times 5$



Médian  
 $5 \times 5$

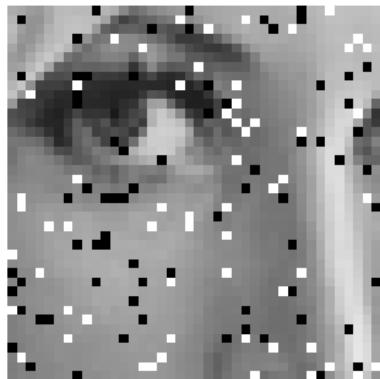


## Filtre médian

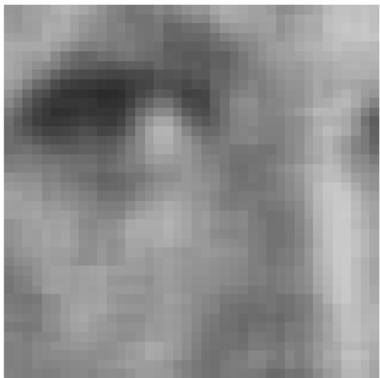
Originale



Observation



Moyenneur  
 $5 \times 5$



Médian  
 $5 \times 5$



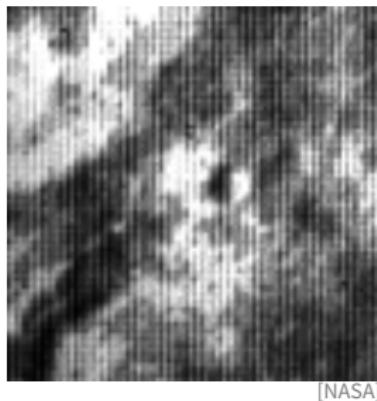
## Filtrage de bruits périodiques

Les bruits périodiques sont caractérisés par des structures dans la transformée de Fourier.

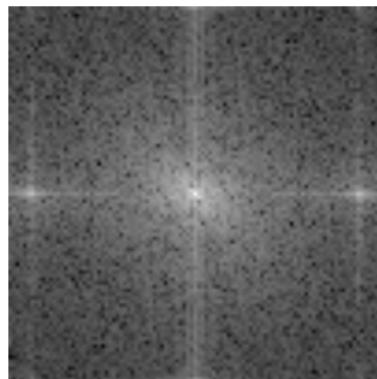
En supprimant ces structures dans le domaine de Fourier, on obtient par transformée de Fourier inverse une image débruitée.

# Filtrage de bruits périodiques

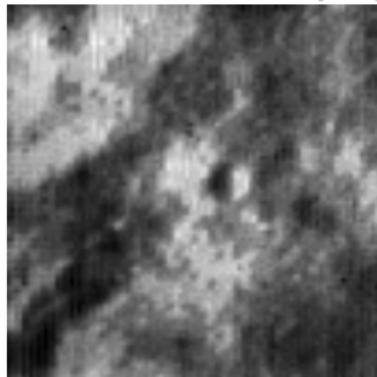
Originale



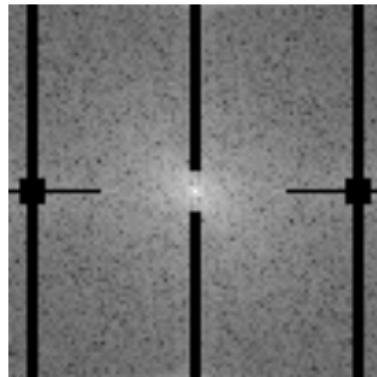
Spectre  
(module)



Filtrée



Spectre filtré  
(module)



## Débruitage par régularisation TV

L'objectif du débruitage est d'obtenir une image  $\hat{x}$ :

- proche de l'observation  $y$
- avec peu de variations d'intensité

Ces deux objectifs doivent être décrits sous forme de fonctions mathématiques (appelées *critères*) que l'on chercher à minimiser ensemble.

## Débruitage par régularisation TV

- Image proche de l'observation  $y$ : le critère d'*adéquation aux données (data-fit)* mesure la différence entre  $x$  et  $y$ .

## Débruitage par régularisation TV

- Image proche de l'observation  $y$ : le critère d'*adéquation aux données (data-fit)* mesure la différence entre  $x$  et  $y$ .

Choix classique : critère des moindres carrés

$$E(x,y) = \sum_{m,n} (y(m,n) - x(m,n))^2$$

## Débruitage par régularisation TV

- Image proche de l'observation  $y$  : le critère d'*adéquation aux données (data-fit)* mesure la différence entre  $x$  et  $y$ .

Choix classique : critère des moindres carrés

$$E(x,y) = \sum_{m,n} (y(m,n) - x(m,n))^2$$

- Image avec peu de variations d'intensité : le critère de *régularisation (regularization)* mesure la différence entre les pixels voisins de  $x$ .

## Débruitage par régularisation TV

- Image proche de l'observation  $y$  : le critère d'*adéquation aux données (data-fit)* mesure la différence entre  $x$  et  $y$ .

Choix classique : critère des moindres carrés

$$E(x,y) = \sum_{m,n} (y(m,n) - x(m,n))^2$$

- Image avec peu de variations d'intensité : le critère de *régularisation (regularization)* mesure la différence entre les pixels voisins de  $x$ .

Choix possible : *variation totale (total variation)*

$$R(x) = \sum_{m,n} |x(m+1,n) - x(m,n)| + \sum_{m,n} |x(m,n+1) - x(m,n)|$$

## Débruitage par régularisation TV

L'objectif du débruitage est donc de trouver  $\hat{x}$  qui minimise à la fois l'adéquation aux données et la régularisation :

$$\hat{x} = \arg \min_x E(x, y) + \lambda R(x)$$

où  $\lambda$  est le paramètre de régularisation et permet de régler le compromis entre adéquation aux données et régularisation.

## Débruitage par régularisation TV

L'objectif du débruitage est donc de trouver  $\hat{x}$  qui minimise à la fois l'adéquation aux données et la régularisation :

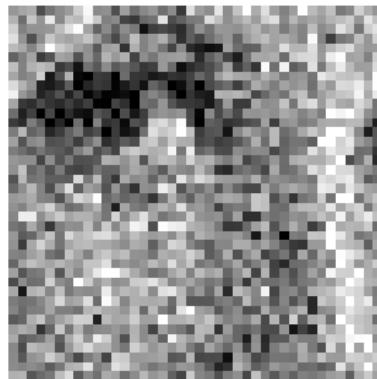
$$\hat{x} = \arg \min_x E(x, y) + \lambda R(x)$$

où  $\lambda$  est le paramètre de régularisation et permet de régler le compromis entre adéquation aux données et régularisation.

C'est un problème d'optimisation : un grand nombre d'algorithmes existent pour minimiser la fonction  $E(x, y) + \lambda R(x)$ .

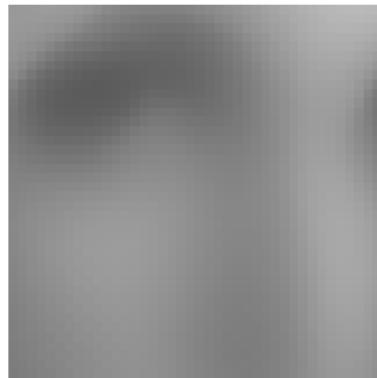
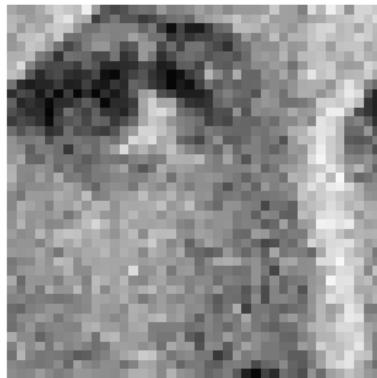
## Débruitage par régularisation TV

Originale



Observation  
(RSB=20 dB)

Régul. TV  
 $\lambda = 0,01$



Régul. TV  
 $\lambda = 1$

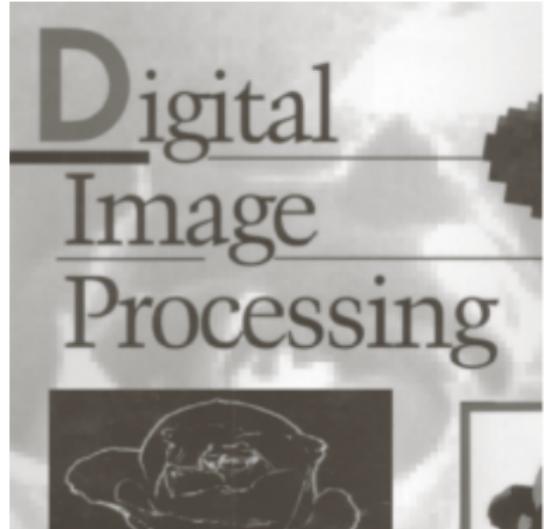
# Conclusion

- Il existe différents types de bruits, le modèle le plus courant (car le plus simple) étant le bruit blanc gaussien additif.
- Méthodes simples :
  - Bruit additif de moyenne nulle → filtre moyenneur
  - Bruit impulsif → filtre médian
  - Bruit périodique → filtrage défini à partir du spectre
- Méthodes par régularisation (TV ou autre)
- Autres méthodes...

# Déconvolution

*(deconvolution)*

# Déconvolution



[Gonzales & Woods]

Flou de bougé

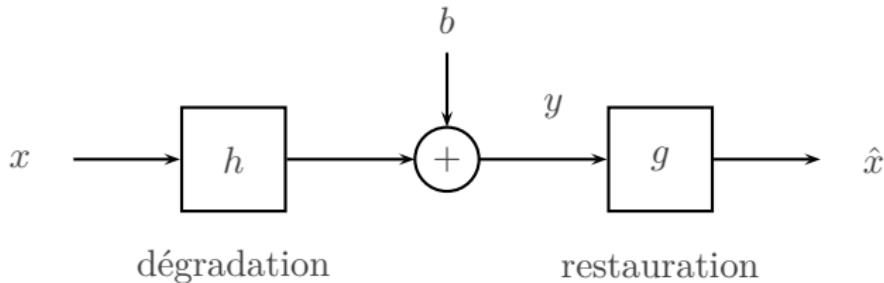
# Déconvolution



[Gonzales & Woods]

Flou de turbulence, flou de mise au point

# Modélisation



$$y = h * x + b$$

$$\hat{x} = g * y$$

## Estimation de la fonction de dégradation

Les méthodes que nous allons étudier nécessitent de connaître  $h$ .

## Estimation de la fonction de dégradation

Les méthodes que nous allons étudier nécessitent de connaître  $h$ .

- **Estimation par observation** en trouvant dans l'image des indices permettant d'estimer  $h$  (par exemple : objet ponctuel isolé, déconvolution naïve en l'absence de bruit dans une petite zone, essai–erreur, ...).

## Estimation de la fonction de dégradation

Les méthodes que nous allons étudier nécessitent de connaître  $h$ .

- **Estimation par observation** en trouvant dans l'image des indices permettant d'estimer  $h$  (par exemple : objet ponctuel isolé, déconvolution naïve en l'absence de bruit dans une petite zone, essai–erreur, ...).
- **Estimation par expérimentation** en reproduisant en laboratoire les conditions d'observation. L'image d'une impulsion observée ainsi donne une estimation de  $h$ .

## Estimation de la fonction de dégradation

Les méthodes que nous allons étudier nécessitent de connaître  $h$ .

- **Estimation par observation** en trouvant dans l'image des indices permettant d'estimer  $h$  (par exemple : objet ponctuel isolé, déconvolution naïve en l'absence de bruit dans une petite zone, essai–erreur, ...).
- **Estimation par expérimentation** en reproduisant en laboratoire les conditions d'observation. L'image d'une impulsion observée ainsi donne une estimation de  $h$ .
- **Estimation par modélisation mathématique** de la physique de l'observation.

## Estimation de la fonction de dégradation

Les méthodes que nous allons étudier nécessitent de connaître  $h$ .

- **Estimation par observation** en trouvant dans l'image des indices permettant d'estimer  $h$  (par exemple : objet ponctuel isolé, déconvolution naïve en l'absence de bruit dans une petite zone, essai–erreur, ...).
- **Estimation par expérimentation** en reproduisant en laboratoire les conditions d'observation. L'image d'une impulsion observée ainsi donne une estimation de  $h$ .
- **Estimation par modélisation mathématique** de la physique de l'observation.

Remarque : les méthodes de déconvolution myope (*blind deconvolution*) estiment non seulement  $x$  mais également  $h$ .

## Filtre inverse

$$y = h * x + b$$

## Filtre inverse

$$y = h * x + b$$

T. de Fourier  
→

$$Y = HX + B$$

## Filtre inverse

$$y = h * x + b$$

T. de Fourier  
→

$$Y = HX + B$$



$$X = \frac{Y - B}{H}$$

## Filtre inverse

$$y = h * x + b$$

T. de Fourier  
→

$$Y = HX + B$$



$$x = \mathcal{F}^{-1} \left[ \frac{Y - B}{H} \right]$$

T. de Fourier inverse  
←

$$X = \frac{Y - B}{H}$$

## Filtre inverse

$$y = h * x + b$$

T. de Fourier  
→

$$Y = HX + B$$



$$x = \mathcal{F}^{-1} \left[ \frac{Y - B}{H} \right]$$

T. de Fourier inverse  
←

$$X = \frac{Y - B}{H}$$

Bruit inconnu

⇒

On calcule  $\hat{x} = \mathcal{F}^{-1} \left[ \frac{Y}{H} \right]$

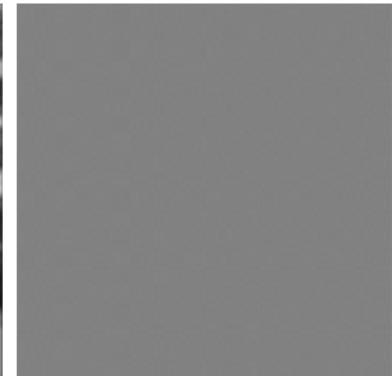
## Filtre inverse



Image originale  $x$



Image observée  $y$



Estimation

$$\hat{x} = \mathcal{F}^{-1} \begin{bmatrix} Y \\ H \end{bmatrix}$$

## Filtre inverse

L'estimation de  $f$  est obtenue par l'équation :

$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{Y}{H} \right]$$

## Filtre inverse

L'estimation de  $f$  est obtenue par l'équation :

$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{Y}{H} \right] = \mathcal{F}^{-1} \left[ X + \frac{B}{H} \right]$$

## Filtre inverse

L'estimation de  $f$  est obtenue par l'équation :

$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{Y}{H} \right] = \mathcal{F}^{-1} \left[ X + \frac{B}{H} \right] = x + \mathcal{F}^{-1} \left[ \frac{B}{H} \right]$$

## Filtre inverse

L'estimation de  $f$  est obtenue par l'équation :

$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{Y}{H} \right] = \mathcal{F}^{-1} \left[ X + \frac{B}{H} \right] = x + \mathcal{F}^{-1} \left[ \frac{B}{H} \right] \neq x$$

## Filtre inverse

L'estimation de  $f$  est obtenue par l'équation :

$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{Y}{H} \right] = \mathcal{F}^{-1} \left[ X + \frac{B}{H} \right] = x + \mathcal{F}^{-1} \left[ \frac{B}{H} \right] \neq x$$

Très souvent,  $H$  est un passe-bas, donc les valeurs de  $H(m,n)$  tendent vers 0 lorsque les fréquences  $(m,n)$  deviennent grandes  $\Rightarrow$  le terme  $B/H$  peut rapidement dominer  $X$ .

## Filtre inverse

L'estimation de  $f$  est obtenue par l'équation :

$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{Y}{H} \right] = \mathcal{F}^{-1} \left[ X + \frac{B}{H} \right] = x + \mathcal{F}^{-1} \left[ \frac{B}{H} \right] \neq x$$

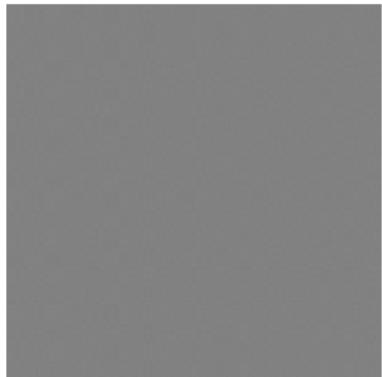
Très souvent,  $H$  est un passe-bas, donc les valeurs de  $H(m,n)$  tendent vers 0 lorsque les fréquences  $(m,n)$  deviennent grandes  $\Rightarrow$  le terme  $B/H$  peut rapidement dominer  $X$ .

Solution : limiter le filtrage autour des fréquences faibles (c'est-à-dire là où  $H$  a des valeurs loin de 0).

## Filtre inverse



Image originale



Filtre inverse



Filtre inverse tronqué  
(10 % de la taille de  
l'image)

# État de l'art des méthodes de déconvolution

- Filtre de Wiener
- Richardson–Lucy [Richardson 1972, Lucy 1974]
- Méthodes par régularisation
- Méthodes bayésiennes
- ...

# État de l'art des méthodes de déconvolution

- Filtre de Wiener
- Richardson–Lucy [Richardson 1972, Lucy 1974]
- Méthodes par régularisation
- Méthodes bayésiennes
- ...

## Filtre de Wiener

Le filtre de Wiener  $G$  prend en compte explicitement les caractéristiques du bruit.

→ Calcul

## Filtre de Wiener

Le filtre de Wiener  $G$  prend en compte explicitement les caractéristiques du bruit.

→ Calcul

$$\hat{x} = \mathcal{F}^{-1} \left[ \underbrace{\frac{H^* S_x}{|H|^2 S_x + S_b}}_G Y \right]$$

- $H$ : transformée de Fourier de la dégradation  $h$
- $H^*$  : conjugué de  $H$
- $|H|^2 = H^* H$ : carré du module de  $H$
- $S_x = |X|^2$  : densité spectrale de puissance de l'image
- $S_b = |B|^2$  : densité spectrale de puissance du bruit

## Filtre de Wiener

$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{H^* S_x}{|H|^2 S_x + S_b} Y \right] = \mathcal{F}^{-1} \left[ \frac{H^*}{|H|^2 + S_b/S_x} Y \right]$$

## Filtre de Wiener

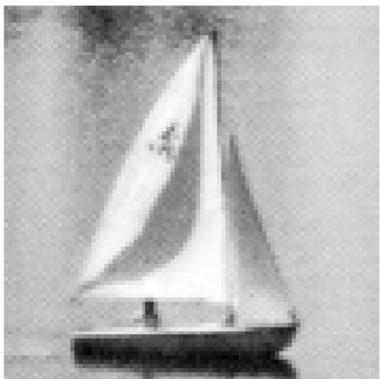
$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{H^* S_x}{|H|^2 S_x + S_b} Y \right] = \mathcal{F}^{-1} \left[ \frac{H^*}{|H|^2 + S_b/S_x} Y \right]$$

- Pour une image  $x$  particulière,  $S_x$  est une constante.
- Dans le cas d'un bruit blanc,  $S_b$  est une constante.
- $S_b$  rarement connu  $\Rightarrow S_b/S_x$  est remplacé par une constante  $K$  que l'on règle.
- Lorsque  $H$ s'annule, on n'observe pas le problème d'augmentation du bruit comme avec le filtre inverse.
- Si le bruit est nul, alors  $S_b = 0$  et on obtient le filtre inverse :

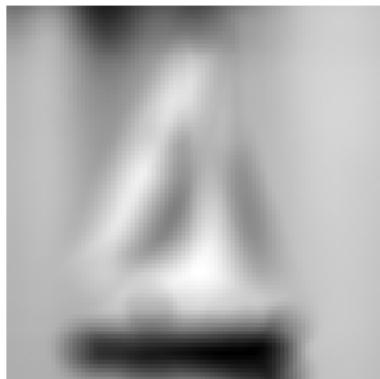
$$\hat{x} = \mathcal{F}^{-1} \left[ \frac{H^*}{|H|^2} Y \right] = \mathcal{F}^{-1} \left[ \frac{1}{H} Y \right]$$

## Filtre de Wiener

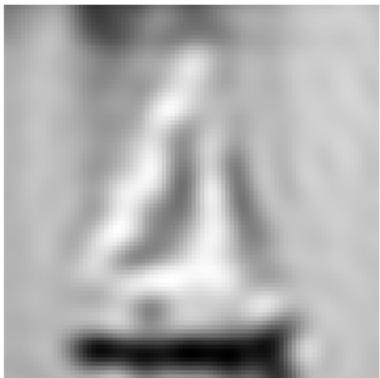
Originale



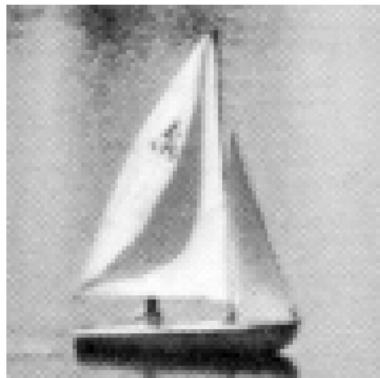
Observation



F. inverse  
tronqué  
(10 %)



F. Wiener  
( $K = 10^{-5}$ )

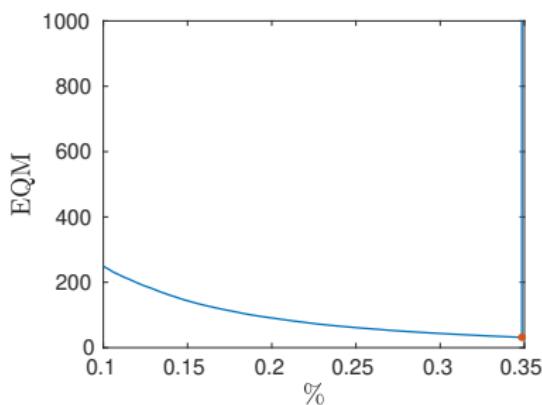


## Filtre de Wiener

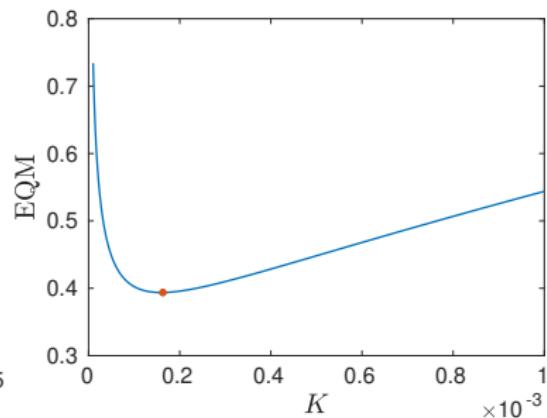
Comment régler le paramètre de chaque méthode ?

# Filtre de Wiener

Comment régler le paramètre de chaque méthode ?



Filtre inverse tronqué

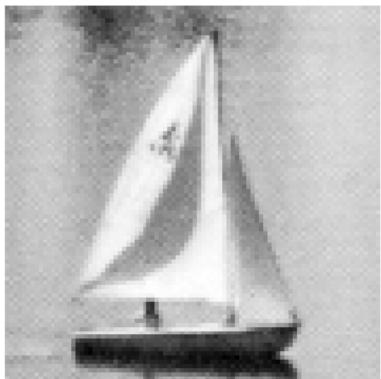


Filtre de Wiener

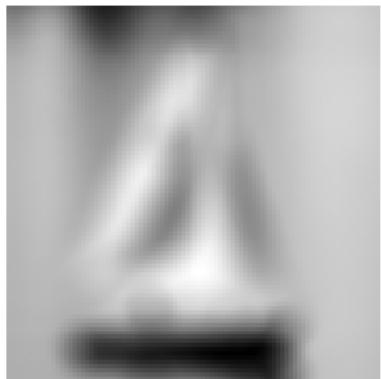
Définition de l'EQM planche 56.

## Filtre de Wiener

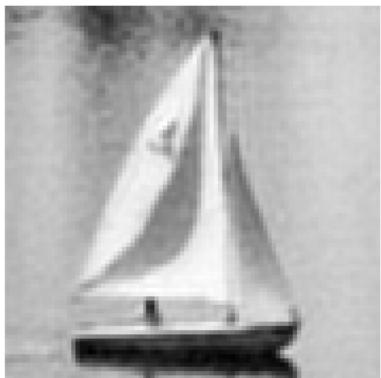
Originale



Observation



F. inverse  
tronqué  
(35 %)



F. Wiener  
( $K=1,6 \cdot 10^{-4}$ )



# Conclusion

- Déconvolution = inverser le filtrage (généralement passe-bas) qu'a subie l'image observée
- Effectuer le filtrage inverse ne fonctionne pas car il tend à augmenter le bruit présent dans l'image
- Filtre de Wiener
- Techniques de régularisation
- Autres méthodes...

## Mesure de la qualité de la restauration

## Mesure de la qualité de la restauration

La différence  $\hat{b}$  entre l'image réelle  $x$ , appelée vérité terrain (*ground truth*), et l'image estimée  $\hat{x}$  est considérée comme un bruit. On peut donc mesurer la qualité de la déconvolution en calculant l'un des indicateurs suivants :

- Le RSB entre  $x$  et  $\hat{b}$ .
- Le PSNR entre  $x$  et  $\hat{b}$ .
- L'erreur quadratique moyenne (*mean square error*)

$$\text{EQM} = \frac{1}{MN} \sum_{m,n} (x(m,n) - \hat{x}(m,n))^2 = \frac{1}{MN} \sum_{m,n} \hat{b}(m,n)^2$$

## Mesure de la qualité de la restauration

La différence  $\hat{b}$  entre l'image réelle  $x$ , appelée vérité terrain (*ground truth*), et l'image estimée  $\hat{x}$  est considérée comme un bruit. On peut donc mesurer la qualité de la déconvolution en calculant l'un des indicateurs suivants :

- Le RSB entre  $x$  et  $\hat{b}$ .
- Le PSNR entre  $x$  et  $\hat{b}$ .
- L'erreur quadratique moyenne (*mean square error*)

$$\text{EQM} = \frac{1}{MN} \sum_{m,n} (x(m,n) - \hat{x}(m,n))^2 = \frac{1}{MN} \sum_{m,n} \hat{b}(m,n)^2$$

Attention : ces mesures quantitatives ne représentent pas forcément la qualité perçue !

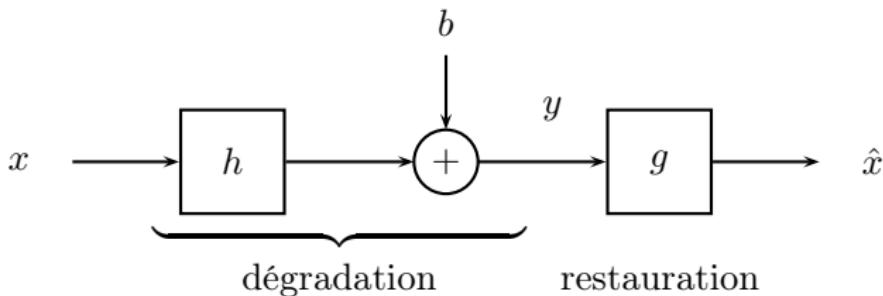
# Mesure de la qualité de la restauration

Procédure :

- 1 Synthétiser ou utiliser une image « originale »  $\rightarrow x$
- 2 Dégrader l'image originale  $\rightarrow y$
- 3 Appliquer la restauration à évaluer  $\rightarrow \hat{x}$
- 4 Quantifier le résultat en utilisant le RSB, le PSNR ou l'EQM

# Conclusion

# Conclusion



- Techniques de débruitage :  
Filtres moyenneur et médian, régularisation...
- Techniques de déconvolution :  
~~Filtre inverse~~, filtre de Wiener, régularisation...

# Bibliographie

- A. Buades, B. Coll, J.-M. Morel « A Non-Local Algorithm for Image Denoising », CVPR 2005.
- L.I. Rudin, S. Osher, E. Fatemi, « Nonlinear total variation based noise removal algorithms » *Physica D*, 60, p. 259–268, 1992.
- A. Chambolle « An algorithm for total variation minimization and applications », *Journal of Mathematical Imaging and Vision*, 20, p. 89–97, 2004.
- W.H. Richardson, « Bayesian-Based Iterative Method of Image Restoration », *Journal of the Optical Society of America*, 62(1), p. 55–59, 1972.
- L.B. Lucy, « An iterative technique for the rectification of observed distributions », *Astronomical Journal*, 79(6), p. 745–754, 1974.