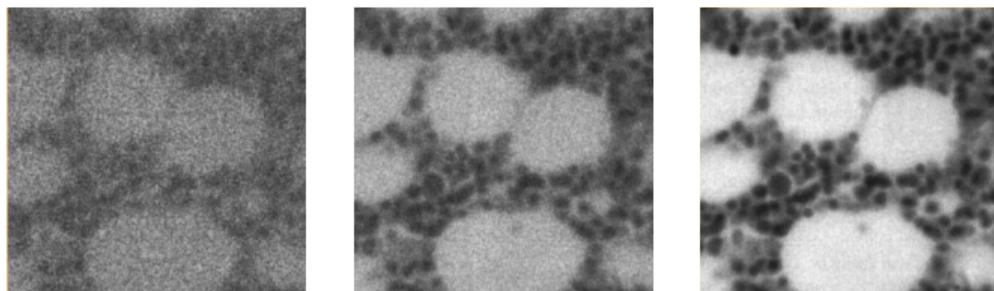


## Partie C — thème 8

### Débruitage (2) : bruit aléatoire



- ▶ *famille des méthodes de restauration d'image*
- ▶ *causes : acquisition, reconstruction, transmission, compression, etc.*
- ▶ *caractérisation : périodique ou aléatoire*
- ▶ *filtrage spatial : filtres moyenneurs, filtres d'ordre*

# 1. Bruit aléatoire en imagerie optique

## 1.1. Altération aléatoire de tous les pixels de l'image

### o Modèles de bruits

▶  $\nu(x, y)$  : fonction bruit définissant la dégradation en chaque pixel

▶ bruit **additif** : le plus courant en imagerie optique

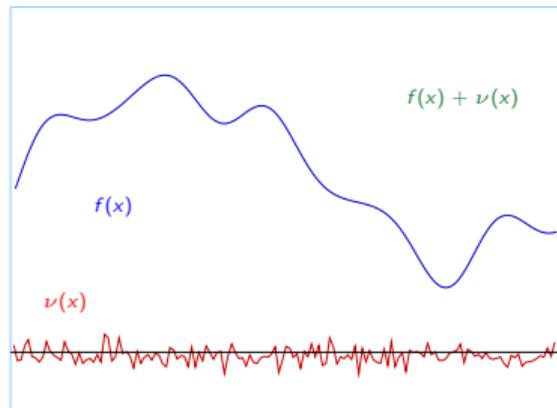
$$I'(x, y) = I(x, y) + \nu(x, y)$$

▶ filtre moyenneur : suppression des hautes fréquences  
→ réduction du bruit mais également des hautes fréquences de  $I$  : **compromis**

▶ la capacité à reconstruire le signal dépend du **rapport signal sur bruit** (SNR : *signal-to-noise ratio*)

▶ bruit **multiplicatif** : grain (par exemple radar)

$$I'(x, y) = I(x, y) \times \nu(x, y)$$



$$SNR = \frac{\bar{f}^2}{\nu^2} \text{ ou en dB}$$
$$SNR_{DB} = 10 \log_{10} \left( \frac{\bar{f}^2}{\nu^2} \right)$$

# 1. Bruit aléatoire en imagerie optique

## 1.1. Altération aléatoire de tous les pixels de l'image

### o Modèles de bruits

▶  $\nu(x, y)$  : fonction bruit définissant la dégradation en chaque pixel

▶ **bruit additif** : le plus courant en imagerie optique

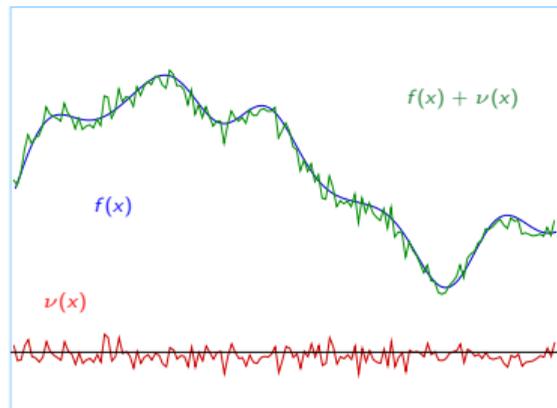
$$I'(x, y) = I(x, y) + \nu(x, y)$$

▶ filtre moyenneur : suppression des hautes fréquences  
→ réduction du bruit mais également des hautes fréquences de  $I$  : **compromis**

▶ la capacité à reconstruire le signal dépend du **rapport signal sur bruit** (SNR : *signal-to-noise ratio*)

▶ **bruit multiplicatif** : grain (par exemple radar)

$$I'(x, y) = I(x, y) \times \nu(x, y)$$



$$SNR = \frac{\bar{f}^2}{\nu^2} \text{ ou en dB}$$
$$SNR_{DB} = 10 \log_{10} \left( \frac{\bar{f}^2}{\nu^2} \right)$$

# 1. Bruit aléatoire en imagerie optique

## 1.1. Altération aléatoire de tous les pixels de l'image

### o Modèles de bruits

▶  $\nu(x, y)$  : fonction bruit définissant la dégradation en chaque pixel

▶ bruit **additif** : le plus courant en imagerie optique

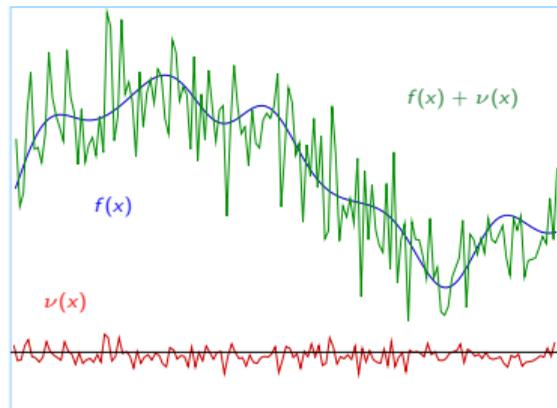
$$I'(x, y) = I(x, y) + \nu(x, y)$$

▶ filtre moyenneur : suppression des hautes fréquences  
→ réduction du bruit mais également des hautes fréquences de  $I$  : **compromis**

▶ la capacité à reconstruire le signal dépend du **rapport signal sur bruit** (SNR : *signal-to-noise ratio*)

▶ bruit **multiplicatif** : grain (par exemple radar)

$$I'(x, y) = I(x, y) \times \nu(x, y)$$



$$SNR = \frac{\bar{f}^2}{\nu^2} \text{ ou en dB}$$
$$SNR_{DB} = 10 \log_{10} \left( \frac{\bar{f}^2}{\nu^2} \right)$$

# 1. Bruit aléatoire en imagerie optique

## 1.1. Altération aléatoire de tous les pixels de l'image

### o Modèles de bruits

▶  $\nu(x, y)$  : fonction bruit définissant la dégradation en chaque pixel

▶ bruit **additif** : le plus courant en imagerie optique

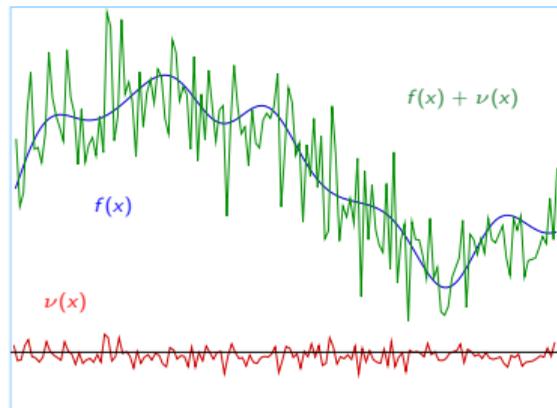
$$I'(x, y) = I(x, y) + \nu(x, y)$$

▶ filtre moyenneur : suppression des hautes fréquences  
→ réduction du bruit mais également des hautes fréquences de  $I$  : **compromis**

▶ la capacité à reconstruire le signal dépend du **rapport signal sur bruit** (SNR : *signal-to-noise ratio*)

▶ bruit **multiplicatif** : grain (par exemple radar)

$$I'(x, y) = I(x, y) \times \nu(x, y)$$

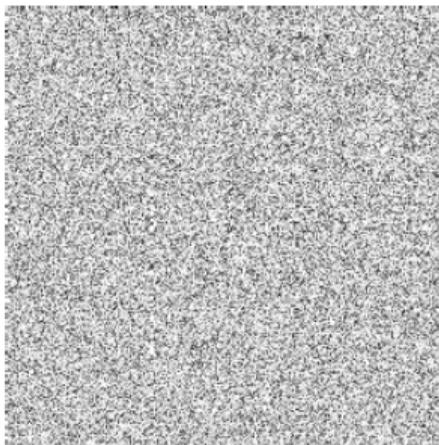


$$SNR = \frac{\bar{f}^2}{\nu^2} \text{ ou en dB}$$
$$SNR_{DB} = 10 \log_{10} \left( \frac{\bar{f}^2}{\nu^2} \right)$$

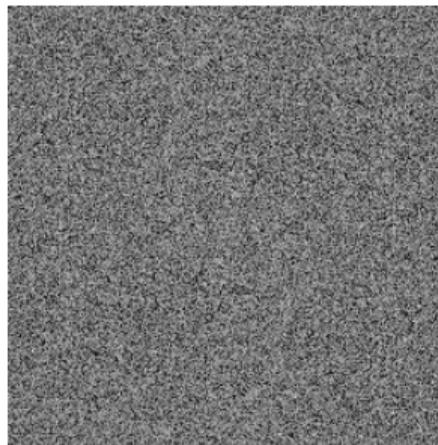
○ **Caractérisation fréquentielle du bruit additif en imagerie optique**

▶ généralement : bruit blanc (« neige »)

▶ le spectre du bruit contient toutes les fréquences avec la même énergie (analogie avec lumière blanche)



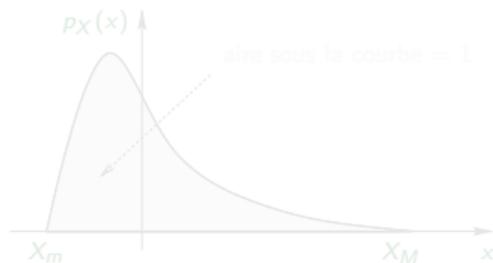
$I$



$\mathcal{F}(I)$

## o Caractérisation statistique du bruit par une densité de probabilité

- ▶ modélisation statistique : description  $\nu$  (fonction bruit) par sa densité de probabilité  
bruit : écarts de niveaux de gris par rapport au signal image ( $I(P) + \nu(P)$ )  
l'amplitude des écarts est caractérisée par une densité de probabilité
- ▶ rappel sur les lois de probabilité continues
  - ▶ variable aléatoire continue  $X$  sur  $[X_m, X_M]$  caractérisée par sa fonction de densité de probabilité  $p_X(x)$

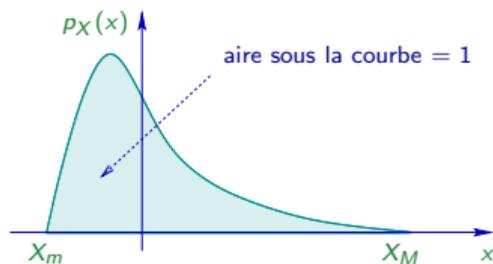


$$p\{X \in [a, b]\} = \int_a^b p_X(x) dx$$

$$\left[ \text{probabilité que le bruit en un point } P \text{ soit égal à } n \right] = p(\nu(P) = n) = \int_{n-0.5}^{n+0.5} p(x) dx \sim p(n)$$

## o Caractérisation statistique du bruit par une densité de probabilité

- ▶ modélisation statistique : description  $\nu$  (fonction bruit) par sa densité de probabilité  
bruit : écarts de niveaux de gris par rapport au signal image ( $I(P) + \nu(P)$ )  
l'amplitude des écarts est caractérisée par une densité de probabilité
- ▶ rappel sur les lois de probabilité continues
  - ▶ variable aléatoire continue  $X$  sur  $[X_m, X_M]$  caractérisée par sa fonction de densité de probabilité  $p_X(x)$

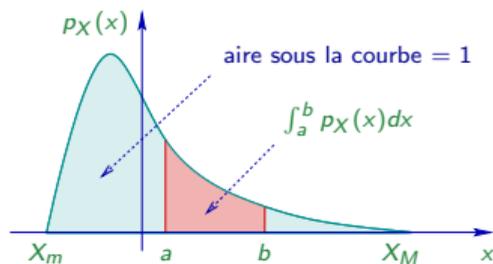


$$p\{X \in [a, b]\} = \int_a^b p_X(x) dx$$

$$\left[ \text{probabilité que le bruit en un point } P \text{ soit égal à } n \right] = p(\nu(P) = n) = \int_{n-0.5}^{n+0.5} p(x) dx \sim p(n)$$

○ **Caractérisation statistique du bruit par une densité de probabilité**

- ▶ modélisation statistique : description  $\nu$  (fonction bruit) par sa densité de probabilité  
bruit : écarts de niveaux de gris par rapport au signal image ( $I(P) + \nu(P)$ )  
l'amplitude des écarts est caractérisée par une densité de probabilité
- ▶ rappel sur les lois de probabilité continues
  - ▶ variable aléatoire continue  $X$  sur  $[X_m, X_M]$  caractérisée par sa fonction de densité de probabilité  $p_X(x)$

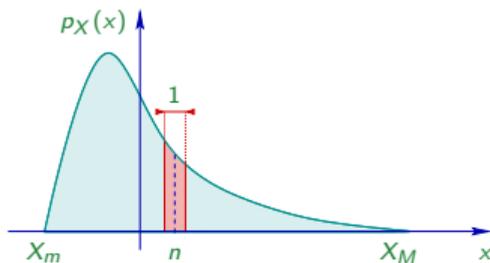


$$p\{X \in [a, b]\} = \int_a^b p_X(x) dx$$

$$\left[ \text{probabilité que le bruit en un point } P \text{ soit égal à } n \right] = p(\nu(P) = n) = \int_{n-0.5}^{n+0.5} p(x) dx \sim p(n)$$

○ **Caractérisation statistique du bruit par une densité de probabilité**

- ▶ modélisation statistique : description  $\nu$  (fonction bruit) par sa densité de probabilité  
bruit : écarts de niveaux de gris par rapport au signal image ( $I(P) + \nu(P)$ )  
l'amplitude des écarts est caractérisée par une densité de probabilité
- ▶ rappel sur les lois de probabilité continues
  - ▶ variable aléatoire continue  $X$  sur  $[X_m, X_M]$  caractérisée par sa fonction de densité de probabilité  $p_X(x)$

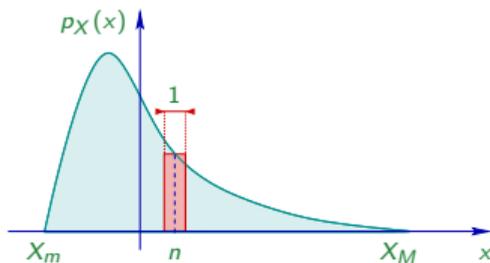


$$p\{X \in [a, b]\} = \int_a^b p_X(x) dx$$

$$\left[ \begin{array}{l} \text{probabilité que le bruit en} \\ \text{un point } P \text{ soit égal à } n \end{array} \right] = p(\nu(P) = n) = \int_{n-0.5}^{n+0.5} p(x) dx \sim p(n)$$

○ **Caractérisation statistique du bruit par une densité de probabilité**

- ▶ modélisation statistique : description  $\nu$  (fonction bruit) par sa densité de probabilité  
bruit : écarts de niveaux de gris par rapport au signal image ( $I(P) + \nu(P)$ )  
l'amplitude des écarts est caractérisée par une densité de probabilité
- ▶ rappel sur les lois de probabilité continues
  - ▶ variable aléatoire continue  $X$  sur  $[X_m, X_M]$  caractérisée par sa fonction de densité de probabilité  $p_X(x)$



$$p\{X \in [a, b]\} = \int_a^b p_X(x) dx$$

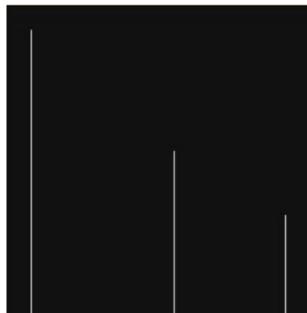
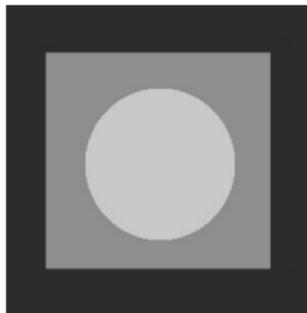
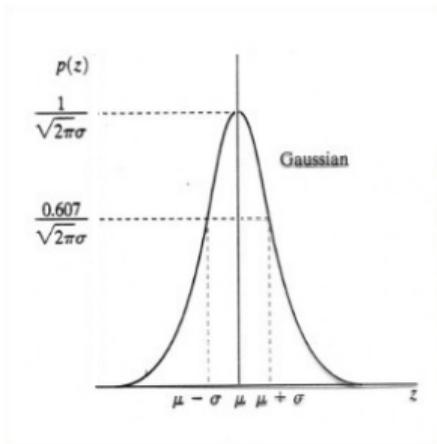
$$\left[ \begin{array}{l} \text{probabilité que le bruit en} \\ \text{un point } P \text{ soit égal à } n \end{array} \right] = p(\nu(P)=n) = \int_{n-0.5}^{n+0.5} p(x) dx \sim p(n)$$

## ○ Bruit gaussien

- ▶ bruit le plus courant en imagerie optique : circuits électroniques, des capteurs en faible luminosité, etc.
- ▶ la densité de probabilité de  $\nu$  est une gaussienne :

$$p_{\nu}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x - \mu)^2}{2\sigma^2}}$$

- ▶ la probabilité du bruit décroît exponentiellement autour de  $\mu$



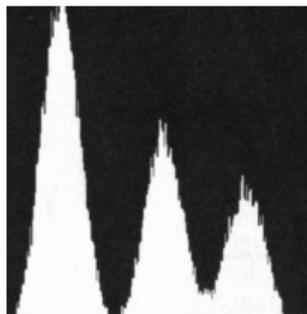
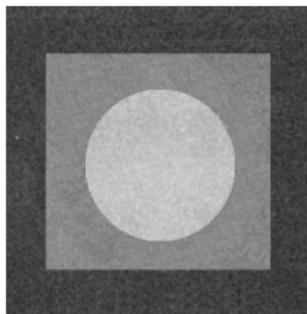
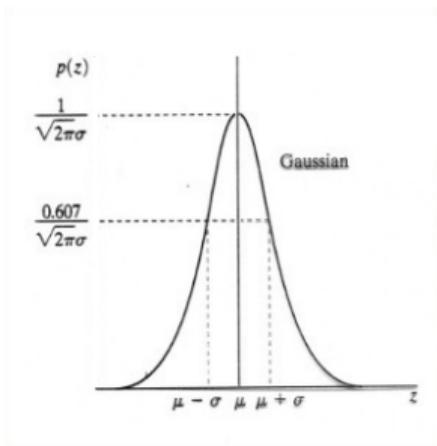
$$\begin{aligned}\mu &= 0 \\ \sigma &= 16\end{aligned}$$

## ○ Bruit gaussien

- ▶ bruit le plus courant en imagerie optique : circuits électroniques, des capteurs en faible luminosité, etc.
- ▶ la densité de probabilité de  $\nu$  est une gaussienne :

$$p_{\nu}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x - \mu)^2}{2\sigma^2}}$$

- ▶ la probabilité du bruit décroît exponentiellement autour de  $\mu$



$$\begin{aligned}\mu &= 0 \\ \sigma &= 16\end{aligned}$$

## ○ Simulation d'un bruit gaussien

- ▶ **théorème central limite** : une somme de processus aléatoires indépendants tend vers une distribution gaussienne
- ▶ application : approximation de  $X_g$  de densité de probabilité  $G_\sigma$  ( $\mu = 0$ ) par une somme de tirages aléatoires uniformes sur  $[0, 1]$  :

$$X_g = \sigma \sqrt{\frac{12}{n}} \left[ \left( \sum_1^n \frac{\text{rand}()}{\text{RAND\_MAX}} \right) - \frac{n}{2} \right]$$

on peut prendre  $n = 12$

$$X_g = \sigma \times \left[ \left( \sum_1^{12} \frac{\text{rand}()}{\text{RAND\_MAX}} \right) - 6 \right]$$

- ▶ mise en œuvre du bruitage gaussien d'une image  $I$  : pour chaque point  $(i, j)$ 
  1. tirage de la variable aléatoire  $X_g$
  2.  $I(i, j) = I(i, j) + X_g$

## ○ Simulation d'un bruit gaussien

- ▶ **théorème central limite** : une somme de processus aléatoires indépendants tend vers une distribution gaussienne
- ▶ application : approximation de  $X_g$  de densité de probabilité  $G_\sigma$  ( $\mu = 0$ ) par une **somme de tirages aléatoires uniformes** sur  $[0, 1]$  :

$$X_g = \sigma \sqrt{\frac{12}{n}} \left[ \left( \sum_1^n \frac{\text{rand}()}{\text{RAND\_MAX}} \right) - \frac{n}{2} \right]$$

on peut prendre  $n = 12$

$$X_g = \sigma \times \left[ \left( \sum_1^{12} \frac{\text{rand}()}{\text{RAND\_MAX}} \right) - 6 \right]$$

- ▶ mise en œuvre du bruitage gaussien d'une image  $I$  : pour chaque point  $(i, j)$ 
  1. tirage de la variable aléatoire  $X_g$
  2.  $I(i, j) = I(i, j) + X_g$

## ○ Simulation d'un bruit gaussien

- ▶ **théorème central limite** : une somme de processus aléatoires indépendants tend vers une distribution gaussienne
- ▶ application : approximation de  $X_g$  de densité de probabilité  $G_\sigma$  ( $\mu = 0$ ) par une **somme de tirages aléatoires uniformes** sur  $[0, 1]$  :

$$X_g = \sigma \sqrt{\frac{12}{n}} \left[ \left( \sum_1^n \frac{rand()}{RAND\_MAX} \right) - \frac{n}{2} \right]$$

on peut prendre  $n = 12$

$$X_g = \sigma \times \left[ \left( \sum_1^{12} \frac{rand()}{RAND\_MAX} \right) - 6 \right]$$

- ▶ mise en œuvre du bruitage gaussien d'une image  $I$  : pour chaque point  $(i, j)$ 
  1. tirage de la variable aléatoire  $X_g$
  2.  $I(i, j) = I(i, j) + X_g$

## 1.2. Bruit impulsionnel

### ○ Principe

- ▶ seuls certains pixels de l'image sont altérés
- ▶ capteurs défectueux, erreurs de transmission, poussières, taches

### ○ Bruit impulsionnel statique

- ▶ pour chaque pixel  $p$ , trois possibilités :
  - ▶  $p \leftarrow a$  (valeur sombre) avec une probabilité  $p_a$
  - ▶  $p \leftarrow b$  (valeur claire) avec une probabilité  $p_b$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_a - p_b$
- ▶ cas particulier (poivre et sel) :  $a = 0$  et  $b = I_{max}$

### ○ Bruit impulsionnel aléatoire

- ▶  $\nu$  une fonction bruit uniforme sur  $[0..i_{max}]$
- ▶ pour chaque pixel  $p$ , deux possibilités :
  - ▶  $p \leftarrow \nu$  avec une probabilité  $p_\nu$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_\nu$



## 1.2. Bruit impulsionnel

### ○ Principe

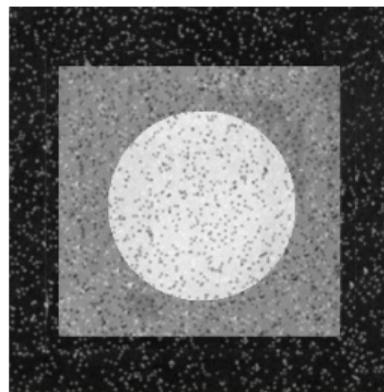
- ▶ seuls certains pixels de l'image sont altérés
- ▶ capteurs défectueux, erreurs de transmission, poussières, taches

### ○ Bruit impulsionnel statique

- ▶ pour chaque pixel  $p$ , trois possibilités :
  - ▶  $p \leftarrow a$  (valeur sombre) avec une probabilité  $p_a$
  - ▶  $p \leftarrow b$  (valeur claire) avec une probabilité  $p_b$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_a - p_b$
- ▶ cas particulier (poivre et sel) :  $a = 0$  et  $b = I_{max}$

### ○ Bruit impulsionnel aléatoire

- ▶  $\nu$  une fonction bruit uniforme sur  $[0..i_{max}]$
- ▶ pour chaque pixel  $p$ , deux possibilités :
  - ▶  $p \leftarrow \nu$  avec une probabilité  $p_\nu$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_\nu$



## 1.2. Bruit impulsionnel

### ○ Principe

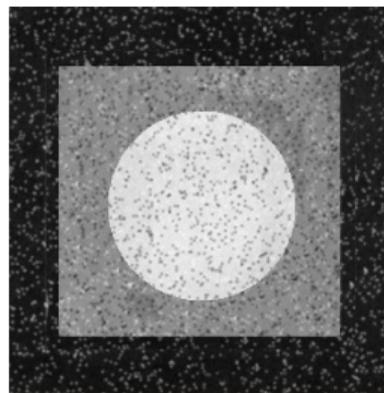
- ▶ seuls certains pixels de l'image sont altérés
- ▶ capteurs défectueux, erreurs de transmission, poussières, taches

### ○ Bruit impulsionnel statique

- ▶ pour chaque pixel  $p$ , trois possibilités :
  - ▶  $p \leftarrow a$  (valeur sombre) avec une probabilité  $p_a$
  - ▶  $p \leftarrow b$  (valeur claire) avec une probabilité  $p_b$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_a - p_b$
- ▶ cas particulier (poivre et sel) :  $a = 0$  et  $b = I_{max}$

### ○ Bruit impulsionnel aléatoire

- ▶  $\nu$  une fonction bruit uniforme sur  $[0..i_{max}]$
- ▶ pour chaque pixel  $p$ , deux possibilités :
  - ▶  $p \leftarrow \nu$  avec une probabilité  $p_\nu$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_\nu$



## 1.2. Bruit impulsionnel

### ○ Principe

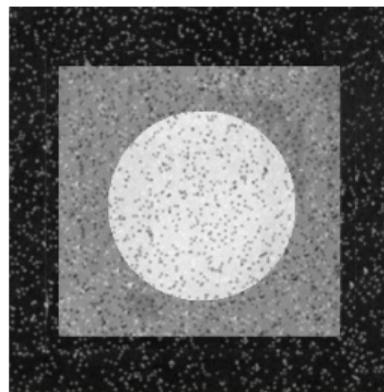
- ▶ seuls certains pixels de l'image sont altérés
- ▶ capteurs défectueux, erreurs de transmission, poussières, taches

### ○ Bruit impulsionnel statique

- ▶ pour chaque pixel  $p$ , trois possibilités :
  - ▶  $p \leftarrow a$  (valeur sombre) avec une probabilité  $p_a$
  - ▶  $p \leftarrow b$  (valeur claire) avec une probabilité  $p_b$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_a - p_b$
- ▶ cas particulier (poivre et sel) :  $a = 0$  et  $b = I_{max}$

### ○ Bruit impulsionnel aléatoire

- ▶  $\nu$  une fonction bruit uniforme sur  $[0..i_{max}]$
- ▶ pour chaque pixel  $p$ , deux possibilités :
  - ▶  $p \leftarrow \nu$  avec une probabilité  $p_\nu$
  - ▶  $p$  reste inchangé avec une probabilité  $1 - p_\nu$



## 1.3. Méthodes de traitement du bruit aléatoire

- ▶ bruit : hautes fréquences
  - ▶ bruit additif
    - ▶ bruit décorrélé, le plus souvent gaussien
    - ▶ filtres moyenneurs
  - ▶ bruit impulsionnel
    - ▶ remplacement de valeurs aberrantes
    - ▶ filtres d'ordre
- ▶ compromis entre
  - ▶ suppression du bruit et préservation des HF (détails, contours, textures)
  - ▶ adaptation locale : préservation des hautes fréquence pertinentes

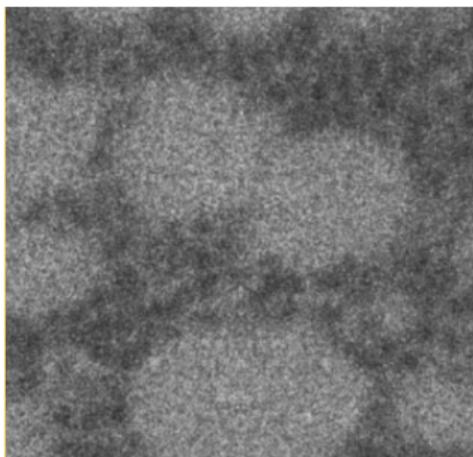
## 2. Traitement du bruit additif gaussien

### 2.1. Moyenne d'une suite d'images

- ▶ hypothèse : on peut acquérir un ensemble d'images  $g_i = f + \nu_i$

on construit 
$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^M g_i(x, y)$$

- ▶ si  $\nu(x, y)$  est **décorrélé** et de **moyenne nulle**, alors  $E\{\bar{g}(x, y)\} = f(x, y)$



Microscopie à fluorescence de la moelle osseuse – J.C.Russ,  
*Img. Process. Handbook 4<sup>ed</sup>*

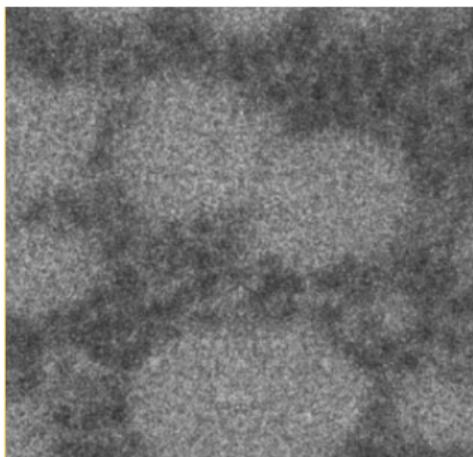
## 2. Traitement du bruit additif gaussien

### 2.1. Moyenne d'une suite d'images

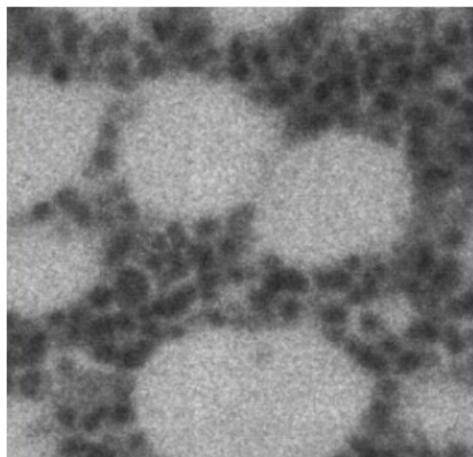
- ▶ hypothèse : on peut acquérir un ensemble d'images  $g_i = f + \nu_i$

on construit 
$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^M g_i(x, y)$$

- ▶ si  $\nu(x, y)$  est **décorrélé** et de **moyenne nulle**, alors  $E\{\bar{g}(x, y)\} = f(x, y)$



Microscopie à fluorescence de la moelle osseuse – J.C.Russ,  
*Img. Process. Handbook 4<sup>ed</sup>*



$M = 4$

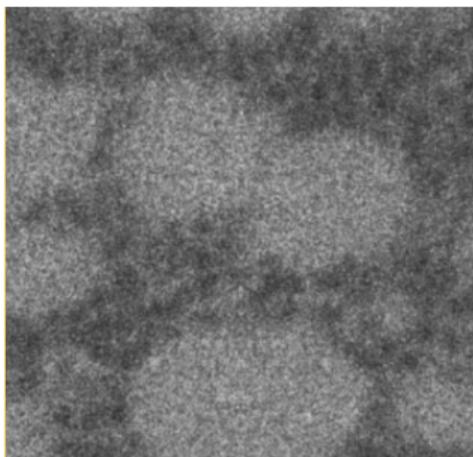
## 2. Traitement du bruit additif gaussien

### 2.1. Moyenne d'une suite d'images

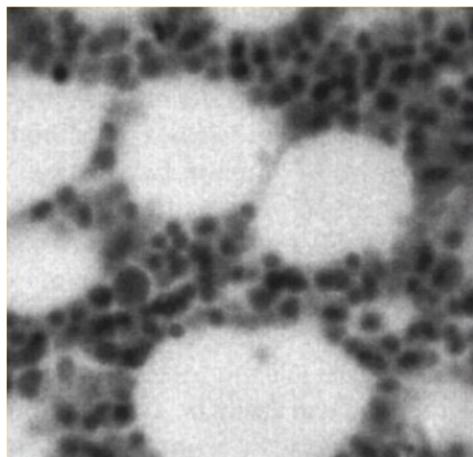
- ▶ hypothèse : on peut acquérir un ensemble d'images  $g_i = f + \nu_i$

on construit 
$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^M g_i(x, y)$$

- ▶ si  $\nu(x, y)$  est **décorrélé** et de **moyenne nulle**, alors  $E\{\bar{g}(x, y)\} = f(x, y)$



Microscopie à fluorescence de la moelle osseuse – J.C.Russ,  
*Img. Process. Handbook 4<sup>ed</sup>*



$M = 16$

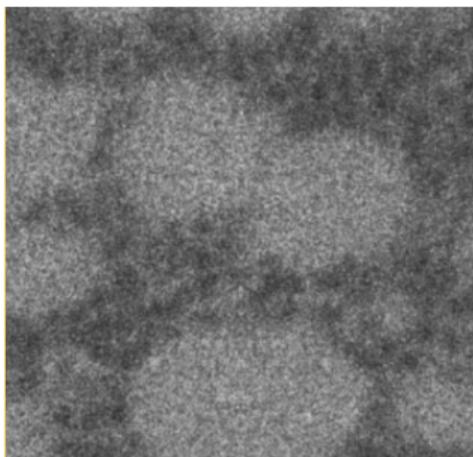
## 2. Traitement du bruit additif gaussien

### 2.1. Moyenne d'une suite d'images

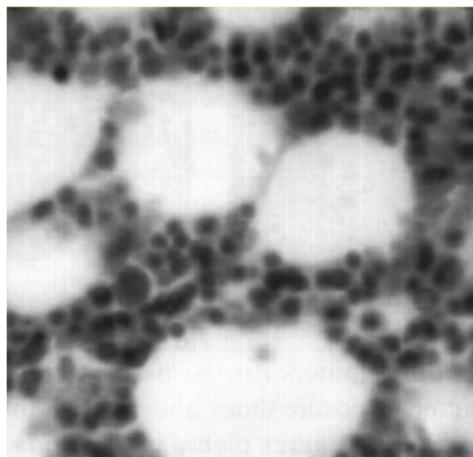
- ▶ hypothèse : on peut acquérir un ensemble d'images  $g_i = f + \nu_i$

on construit 
$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^M g_i(x, y)$$

- ▶ si  $\nu(x, y)$  est **décorrélé** et de **moyenne nulle**, alors  $E\{\bar{g}(x, y)\} = f(x, y)$



Microscopie à fluorescence de la moelle osseuse – J.C.Russ,  
*Img. Process. Handbook 4<sup>ed</sup>*



$M = 256$

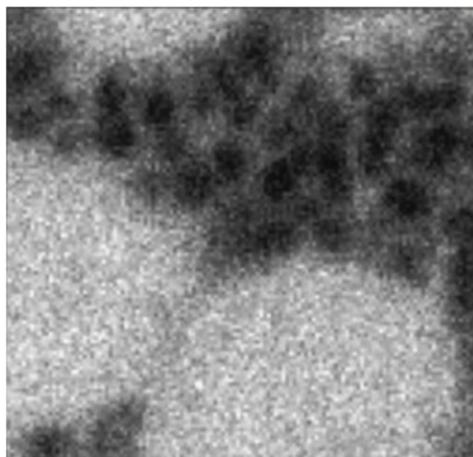
## 2. Traitement du bruit additif gaussien

### 2.1. Moyenne d'une suite d'images

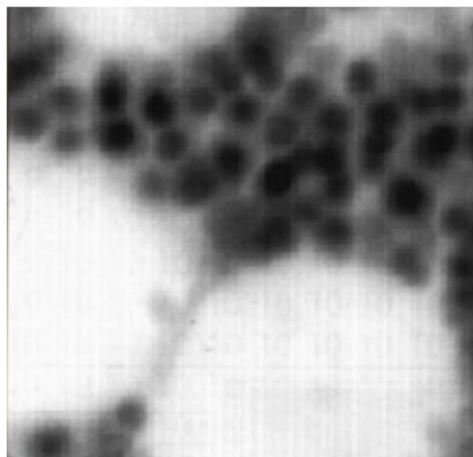
- ▶ hypothèse : on peut acquérir un ensemble d'images  $g_i = f + \nu_i$

on construit 
$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^M g_i(x, y)$$

- ▶ si  $\nu(x, y)$  est **décorrélé** et de **moyenne nulle**, alors  $E\{\bar{g}(x, y)\} = f(x, y)$



Microscopie à fluorescence de la moelle osseuse – J.C.Russ,  
*Img. Process. Handbook 4<sup>ed</sup>*



$M = 256$

## 2.2. Filtres moyenneurs

- **Principe** : moyenne des valeurs sur un voisinage  $V(p)$  de taille  $N$ 
  - ▶ moyenne **arithmétique** (convolution par un filtre boîte)

$$I'(p) = \frac{1}{N} \sum_{p \in V(p)} I(p)$$

- ▶ filtre de base, lissage important
- ▶ moyenne **géométrique**

$$I'(p) = \left( \prod_{p \in V(p)} I(p) \right)^{\frac{1}{N}}$$

- ▶ comparable au filtre arithmétique, préserve mieux les détails

## 2.2. Filtrés moyennés

- **Principe** : moyenne des valeurs sur un voisinage  $V(p)$  de taille  $N$ 
  - ▶ moyenne **arithmétique** (convolution par un filtre boîte)

$$I'(p) = \frac{1}{N} \sum_{p \in V(p)} I(p)$$

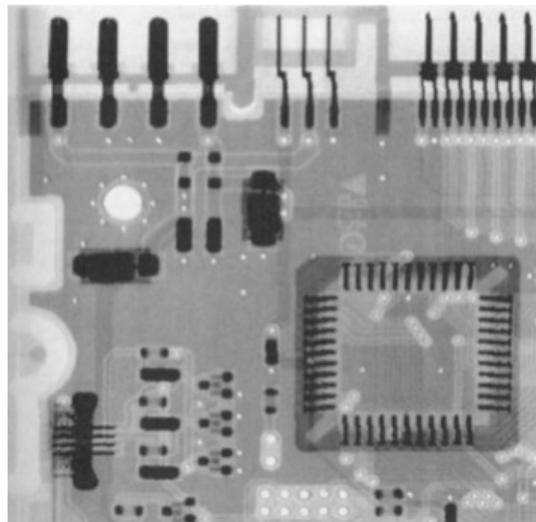
- ▶ filtre de base, lissage important

- ▶ moyenne **géométrique**

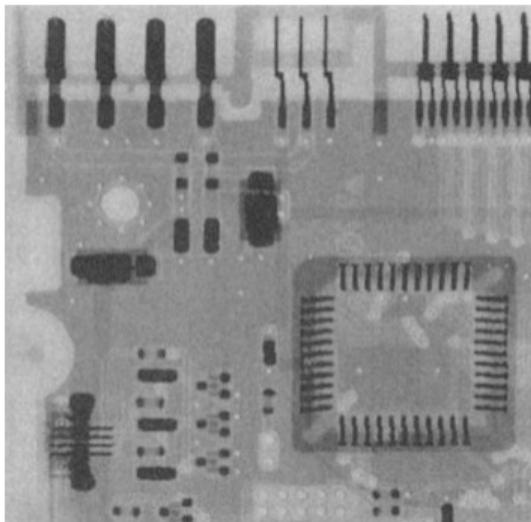
$$I'(p) = \left( \prod_{p \in V(p)} I(p) \right)^{\frac{1}{N}}$$

- ▶ comparable au filtre arithmétique, préserve mieux les détails

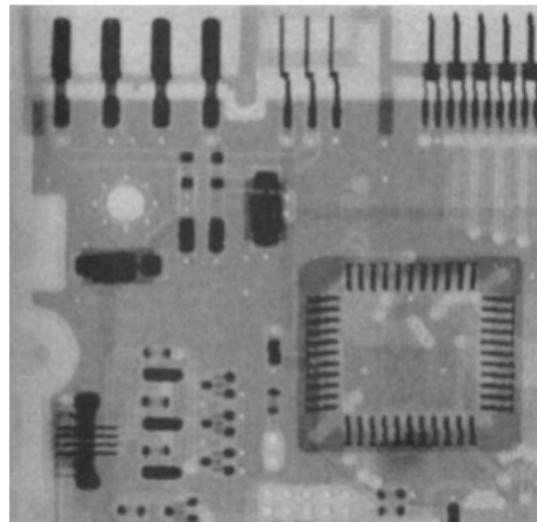
- ▶ exemple : comparaison des moyennes arithmétiques et géométriques



$I$



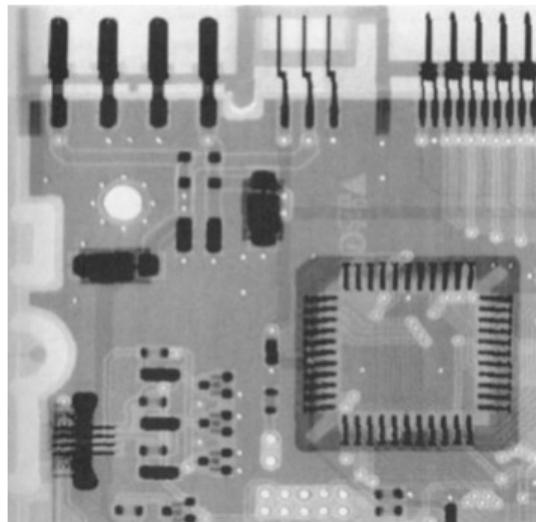
$I + \nu$



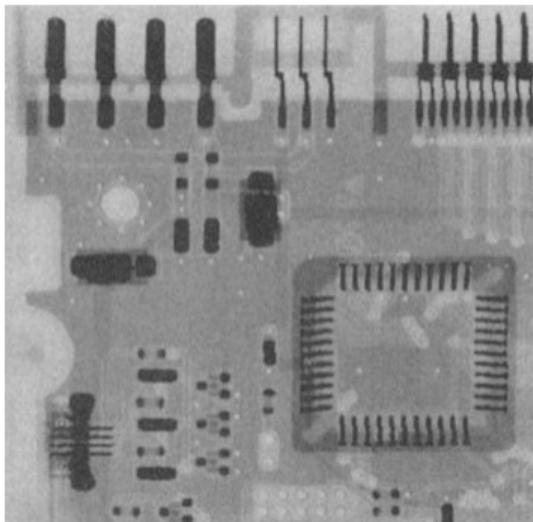
arithmétique

*Image rayon X d'une carte électronique – Gonzalez & Wood, DIP 4<sup>ed</sup> © J.E.Pascente, Lixi Inc.*

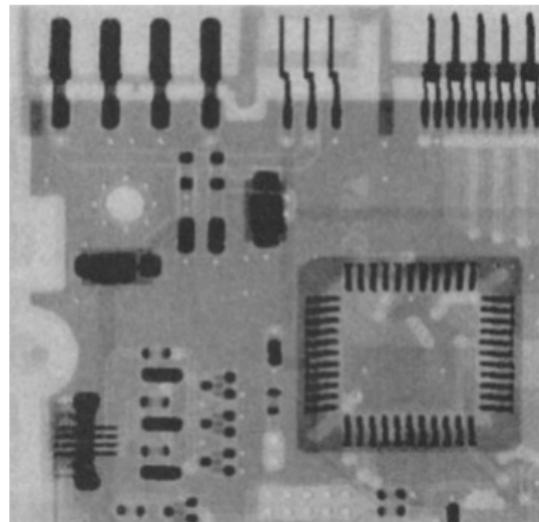
- exemple : comparaison des moyennes arithmétiques et géométriques



$I$



$I + \nu$



géométrique

*Image rayon X d'une carte électronique – Gonzalez & Wood, DIP 4<sup>ed</sup> © J.E.Pascente, Lixi Inc.*

## 2.3. Filtrés moyennés adaptatifs

- ▶ discrimination **bruit** / **détails**
- ▶ détermination statistique d'un **seuil** en deçà duquel un écart est un bruit

### ○ Filtre moyenné seuillé

- ▶ critère : écart à la moyenne locale

$$I'(p) = \begin{cases} \mu_V(p) & \text{si } |I(p) - \mu_V(p)| < T \\ I(p) & \text{sinon} \end{cases}$$

- ▶  $\mu_V(p)$  : moyenne sur un voisinage  $V$  du point  $p$
- ▶ seuil  $T$  (écart de niveau de gris) : peut être ajusté par rapport à l'écart type du bruit si on peut l'estimer (par exemple en calculant l'écart type sur une région suffisamment homogène)

## 2.3. Filtres moyennés adaptatifs

- ▶ discrimination **bruit** / **détails**
- ▶ détermination statistique d'un **seuil** en deçà duquel un écart est un bruit

### ○ Filtre moyenné seuillé

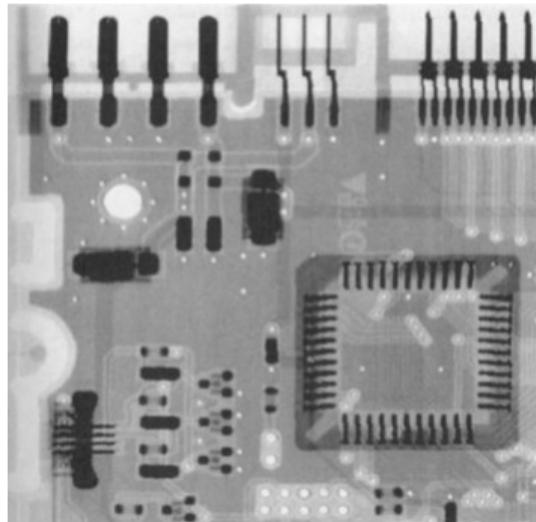
- ▶ critère : écart à la moyenne locale

$$I'(p) = \begin{cases} \mu_V(p) & \text{si } |I(p) - \mu_V(p)| < T \\ I(p) & \text{sinon} \end{cases}$$

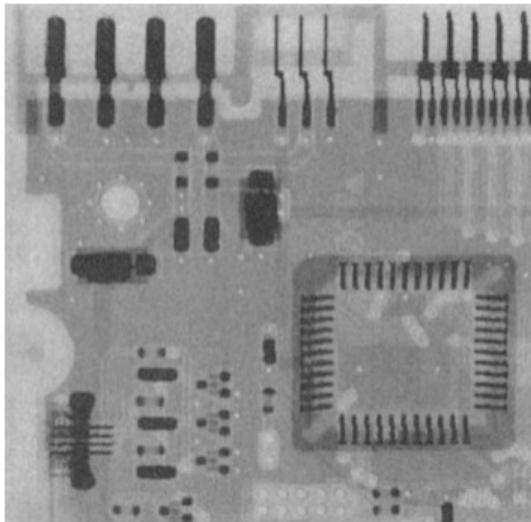
- ▶  $\mu_V(p)$  : moyenne sur un voisinage  $V$  du point  $p$
- ▶ seuil  $T$  (écart de niveau de gris) : peut être ajusté par rapport à l'écart type du bruit si on peut l'estimer (par exemple en calculant l'écart type sur une région suffisamment homogène)

► exemple : mise en œuvre de filtre moyeneur seuillé

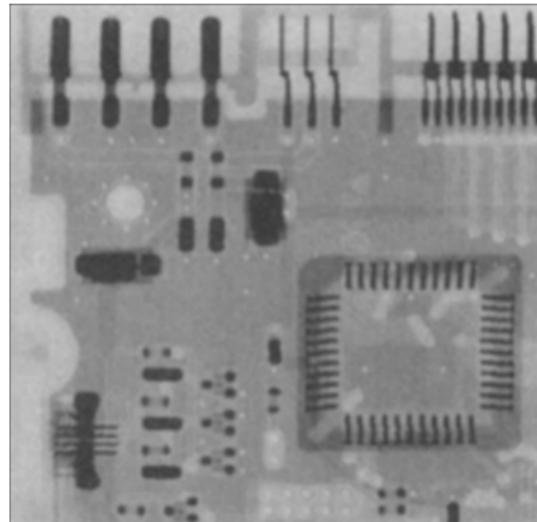
- $w$  : largeur du voisinage
- $T$  : seuil



$I$



$I + \nu$

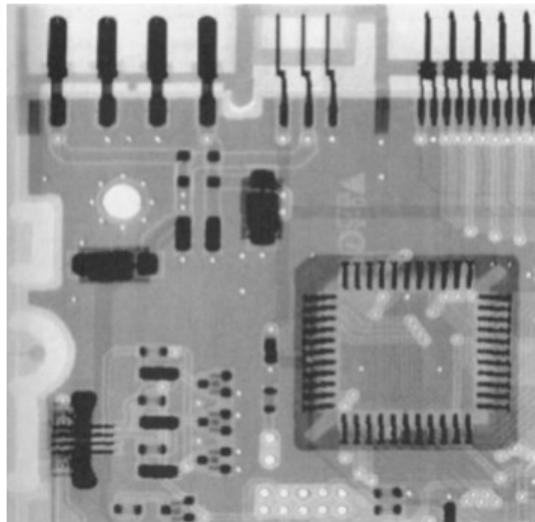


$w = 3$

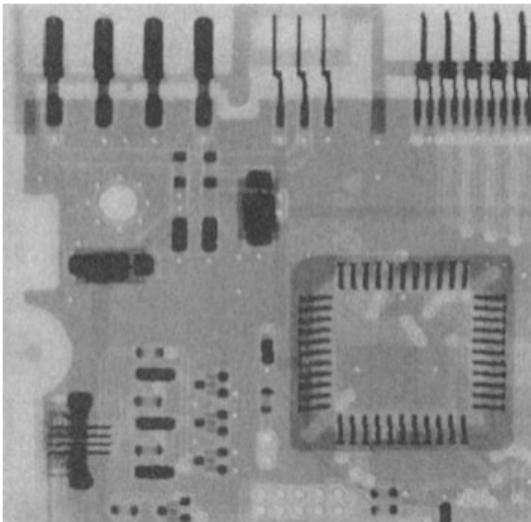
▶ exemple : mise en œuvre de filtre moyenneur seuillé

▶  $w$  : largeur du voisinage

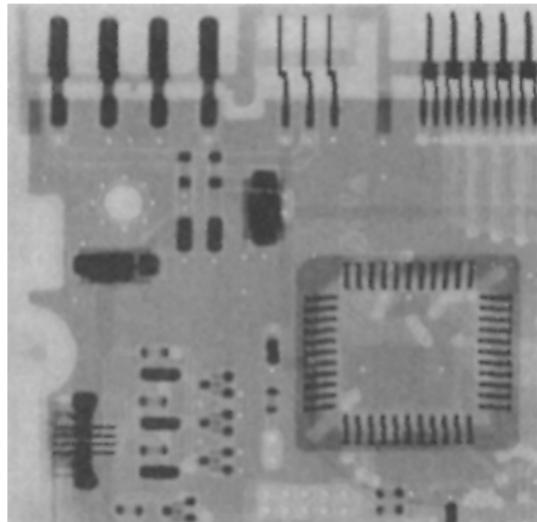
▶  $T$  : seuil



$I$



$I + \nu$

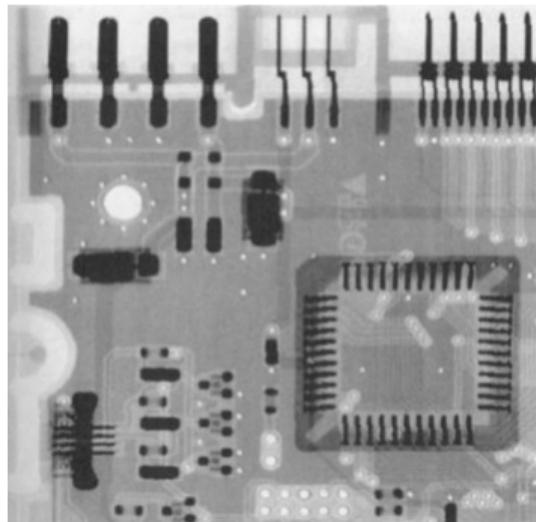


$w = 3$   
 $T = 16$

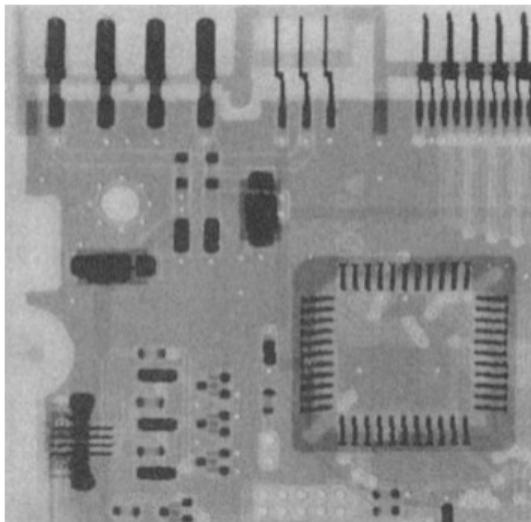
▶ exemple : mise en œuvre de filtre moyenneur seuillé

▶  $w$  : largeur du voisinage

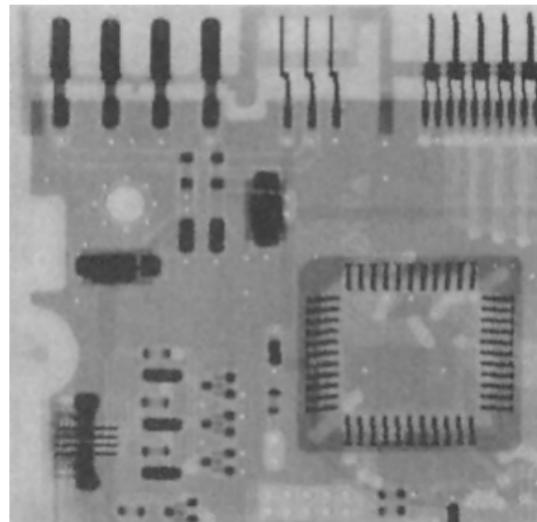
▶  $T$  : seuil



$I$



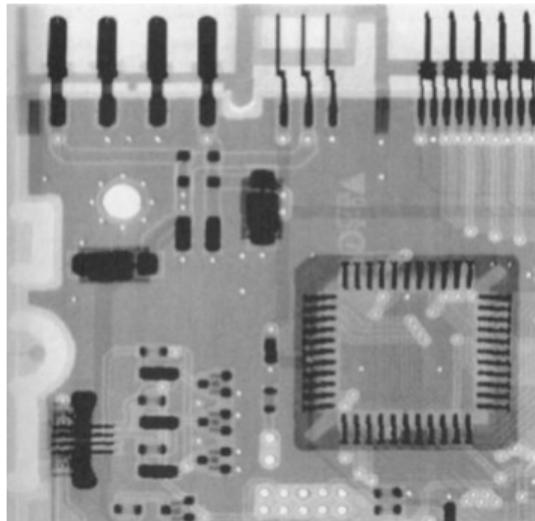
$I + \nu$



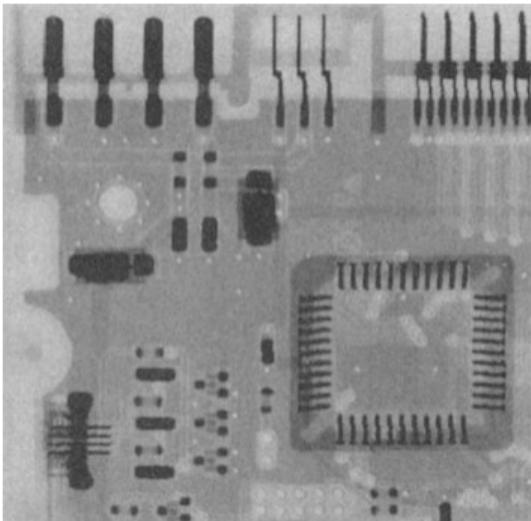
$w = 3$   
 $T = 8$

► exemple : mise en œuvre de filtre moyeneur seuillé

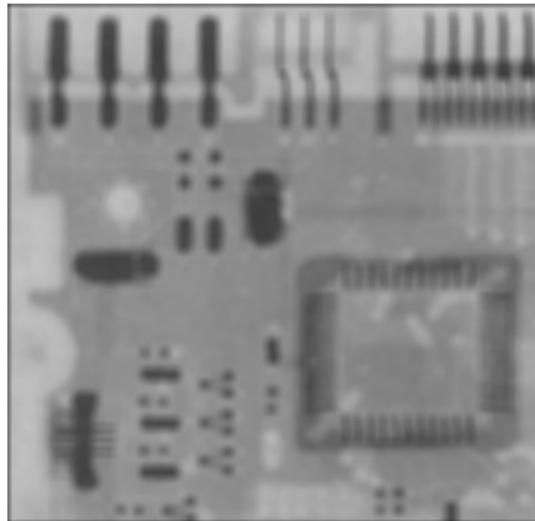
- $w$  : largeur du voisinage
- $T$  : seuil



$I$



$I + \nu$

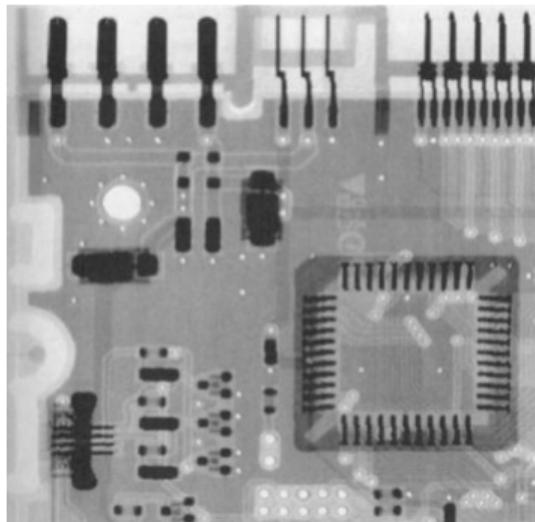


$w = 7$

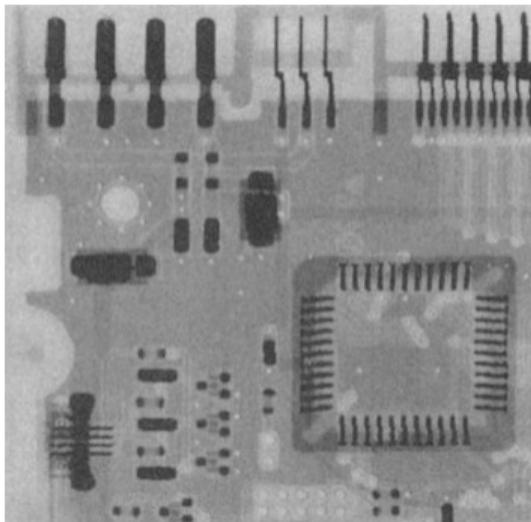
▶ exemple : mise en œuvre de filtre moyenneur seuillé

▶  $w$  : largeur du voisinage

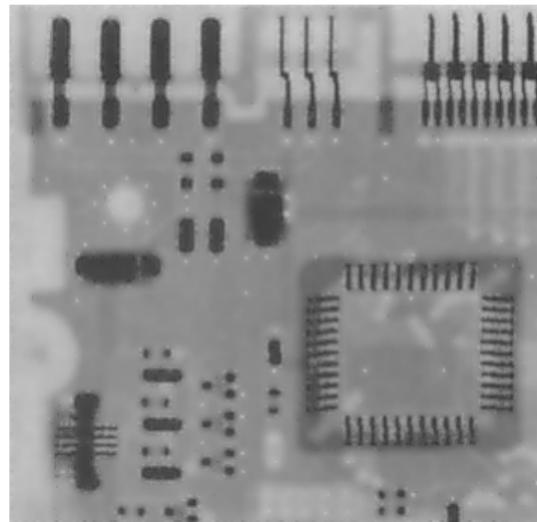
▶  $T$  : seuil



$I$



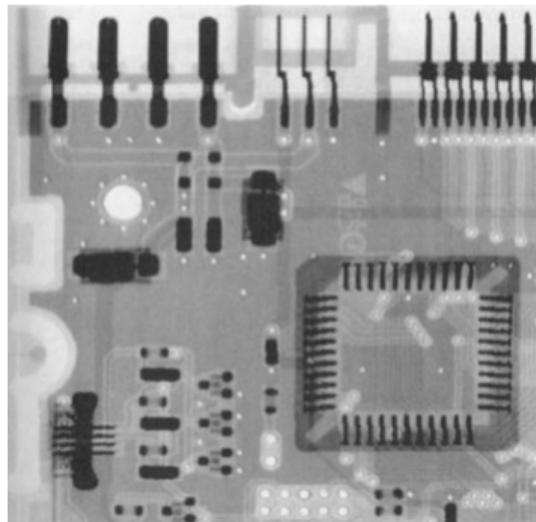
$I + \nu$



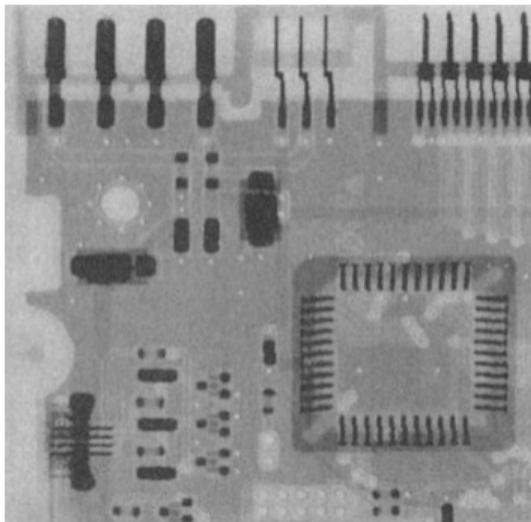
$w = 7$   
 $T = 16$

▶ exemple : mise en œuvre de filtre moyeneur seuillé

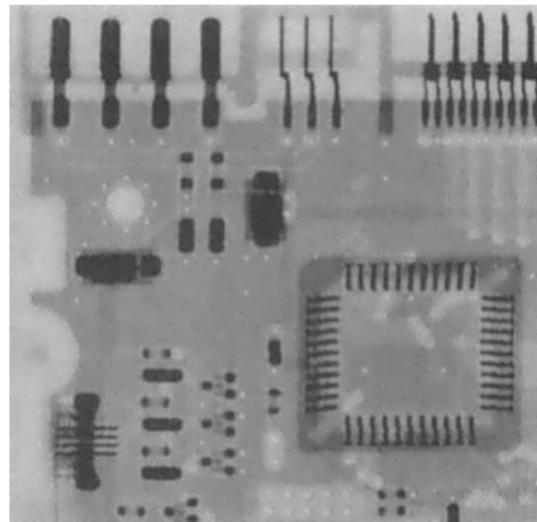
- ▶  $w$  : largeur du voisinage
- ▶  $T$  : seuil



$I$



$I + \nu$



$w = 7$   
 $T = 8$

## ★ Bruit ou détail ?

- ▶ on considère une image  $I$  bruitée par un bruit blanc additif  $\nu$
- ▶ on suppose que l'on connaît l'écart type  $\sigma_\nu$  du bruit  $\nu$
- ▶ on note  $\sigma_V(p)$  l'écart type calculé localement sur un voisinage  $V$  du point  $p$

	A	B	C	D
$\sigma_V(p) < \sigma_\nu$	CD	BR	AN	AN
$\sigma_V(p) \simeq \sigma_\nu$	BR	CD	CD	BR
$\sigma_V(p) \gg \sigma_\nu$	AN	AN	BR	CD

BR :  $p$  est un point de bruit

CD :  $p$  est un point de contour et/ou de détail

AN : condition anormale

## ★ Bruit ou détail ?

- ▶ on considère une image  $I$  bruitée par un bruit blanc additif  $\nu$
- ▶ on suppose que l'on connaît l'écart type  $\sigma_\nu$  du bruit  $\nu$
- ▶ on note  $\sigma_V(p)$  l'écart type calculé localement sur un voisinage  $V$  du point  $p$

	A	B	C	D
$\sigma_V(p) < \sigma_\nu$	CD	BR	AN	AN
$\sigma_V(p) \simeq \sigma_\nu$	BR	CD	CD	BR
$\sigma_V(p) \gg \sigma_\nu$	AN	AN	BR	CD

BR :  $p$  est un point de bruit

CD :  $p$  est un point de contour et/ou de détail

AN : condition anormale

## ○ Filtre statistique adaptatif

- ▶ calcul d'un coefficient local de lissage

$$I'(p) = I(p) - K_p \times (I(p) - \mu_{V(p)})$$

$$K_p \in [0, 1] : \begin{cases} K_p = 0 \longrightarrow I'(p) = I(p) & \text{(pas de lissage)} \\ K_p = 1 \longrightarrow I'(p) = \mu_{V(p)} & \text{(lissage maximal)} \end{cases}$$

- ▶ définition de  $K_p$  par estimation statistique du bruit au point  $p$  :  $\frac{\sigma_v}{\sigma_{V(p)}}$

$$K_p = \min\left(\frac{\sigma_v}{\sigma_{V(p)}}, 1\right) \longrightarrow \begin{cases} \text{hypothèse raisonnable :} & \sigma_{V(p)} \geq \sigma_v \\ \text{bruit maximal :} & \sigma_{V(p)} \simeq \sigma_v \\ \text{arêtes, texture, etc. :} & \sigma_{V(p)} \gg \sigma_v \end{cases}$$

- ▶ estimation de  $\sigma_v$  possible sur une zone uniforme ou par essais successifs
  - ▶ comptage du nombre de valeurs incorrectes :  $\sigma_{V(p)} < \sigma_v$
  - ▶ trop important :  $\sigma_v$  trop grand  $\Rightarrow$  filtrage excessif
  - ▶  $\sigma_v$  trop faible  $\Rightarrow$  filtrage insuffisant
- ▶ possibilité de fixer  $\sigma_v$  relativement à un pourcentage d'erreurs acceptées

## ○ Filtre statistique adaptatif

- ▶ calcul d'un coefficient local de lissage

$$I'(p) = I(p) - K_p \times (I(p) - \mu_{V(p)})$$

$$K_p \in [0, 1] : \begin{cases} K_p = 0 \longrightarrow I'(p) = I(p) & (\text{pas de lissage}) \\ K_p = 1 \longrightarrow I'(p) = \mu_{V(p)} & (\text{lissage maximal}) \end{cases}$$

- ▶ définition de  $K_p$  par estimation statistique du bruit au point  $p$  :  $\frac{\sigma_\nu}{\sigma_{V(p)}}$

$$K_p = \min\left(\frac{\sigma_\nu}{\sigma_{V(p)}}, 1\right) \longrightarrow \begin{cases} \text{hypothèse raisonnable :} & \sigma_{V(p)} \geq \sigma_\nu \\ \text{bruit maximal :} & \sigma_{V(p)} \simeq \sigma_\nu \\ \text{arêtes, texture, etc. :} & \sigma_{V(p)} \gg \sigma_\nu \end{cases} \quad K_p \in [0, 1]$$

- ▶ estimation de  $\sigma_\nu$  possible sur une zone uniforme ou par essais successifs
  - ▶ comptage du nombre de valeurs incorrectes :  $\sigma_{V(p)} < \sigma_\nu$
  - ▶ trop important :  $\sigma_\nu$  trop grand  $\Rightarrow$  filtrage excessif
  - ▶  $\sigma_\nu$  trop faible  $\Rightarrow$  filtrage insuffisant
- ▶ possibilité de fixer  $\sigma_\nu$  relativement à un pourcentage d'erreurs acceptées

## ○ Filtre statistique adaptatif

- ▶ calcul d'un coefficient local de lissage

$$I'(p) = I(p) - K_p \times (I(p) - \mu_{V(p)})$$

$$K_p \in [0, 1] : \begin{cases} K_p = 0 \longrightarrow I'(p) = I(p) & \text{(pas de lissage)} \\ K_p = 1 \longrightarrow I'(p) = \mu_{V(p)} & \text{(lissage maximal)} \end{cases}$$

- ▶ définition de  $K_p$  par estimation statistique du bruit au point  $p$  :  $\frac{\sigma_\nu}{\sigma_{V(p)}}$

$$K_p = \min\left(\frac{\sigma_\nu}{\sigma_{V(p)}}, 1\right) \longrightarrow \begin{cases} \text{hypothèse raisonnable : } \sigma_{V(p)} \geq \sigma_\nu & K_p \in [0, 1] \\ \text{bruit maximal : } \sigma_{V(p)} \simeq \sigma_\nu & K_p \rightarrow 1 \\ \text{arêtes, texture, etc. : } \sigma_{V(p)} \gg \sigma_\nu & \end{cases}$$

- ▶ estimation de  $\sigma_\nu$  possible sur une zone uniforme ou par essais successifs
  - ▶ comptage du nombre de valeurs incorrectes :  $\sigma_{V(p)} < \sigma_\nu$
  - ▶ trop important :  $\sigma_\nu$  trop grand  $\Rightarrow$  filtrage excessif
  - ▶  $\sigma_\nu$  trop faible  $\Rightarrow$  filtrage insuffisant
- ▶ possibilité de fixer  $\sigma_\nu$  relativement à un pourcentage d'erreurs acceptées

## ○ Filtre statistique adaptatif

- ▶ calcul d'un coefficient local de lissage

$$I'(p) = I(p) - K_p \times (I(p) - \mu_{V(p)})$$

$$K_p \in [0, 1] : \begin{cases} K_p = 0 \longrightarrow I'(p) = I(p) & \text{(pas de lissage)} \\ K_p = 1 \longrightarrow I'(p) = \mu_{V(p)} & \text{(lissage maximal)} \end{cases}$$

- ▶ définition de  $K_p$  par estimation statistique du bruit au point  $p$  :  $\frac{\sigma_\nu}{\sigma_{V(p)}}$

$$K_p = \min\left(\frac{\sigma_\nu}{\sigma_{V(p)}}, 1\right) \longrightarrow \begin{cases} \text{hypothèse raisonnable : } \sigma_{V(p)} \geq \sigma_\nu & K_p \in [0, 1] \\ \text{bruit maximal : } \sigma_{V(p)} \simeq \sigma_\nu & K_p \rightarrow 1 \\ \text{arêtes, texture, etc. : } \sigma_{V(p)} \gg \sigma_\nu & K_p \rightarrow 0 \end{cases}$$

- ▶ estimation de  $\sigma_\nu$  possible sur une zone uniforme ou par essais successifs
  - ▶ comptage du nombre de valeurs incorrectes :  $\sigma_{V(p)} < \sigma_\nu$
  - ▶ trop important :  $\sigma_\nu$  trop grand  $\Rightarrow$  filtrage excessif
  - ▶  $\sigma_\nu$  trop faible  $\Rightarrow$  filtrage insuffisant
- ▶ possibilité de fixer  $\sigma_\nu$  relativement à un pourcentage d'erreurs acceptées

## ○ Filtre statistique adaptatif

- ▶ calcul d'un coefficient local de lissage

$$I'(p) = I(p) - K_p \times (I(p) - \mu_{V(p)})$$

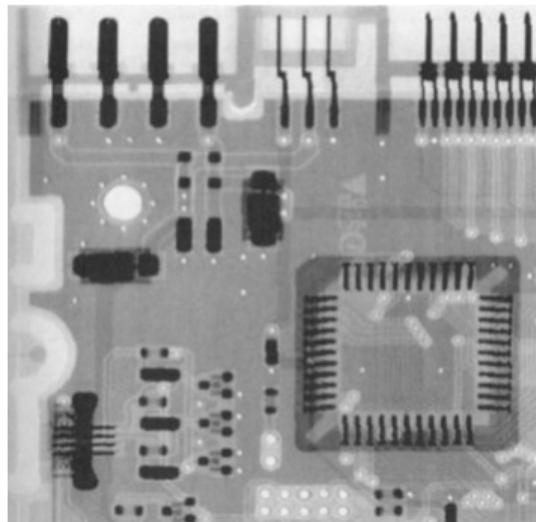
$$K_p \in [0, 1] : \begin{cases} K_p = 0 \longrightarrow I'(p) = I(p) & \text{(pas de lissage)} \\ K_p = 1 \longrightarrow I'(p) = \mu_{V(p)} & \text{(lissage maximal)} \end{cases}$$

- ▶ définition de  $K_p$  par **estimation statistique** du bruit au point  $p$  :  $\frac{\sigma_\nu}{\sigma_{V(p)}}$

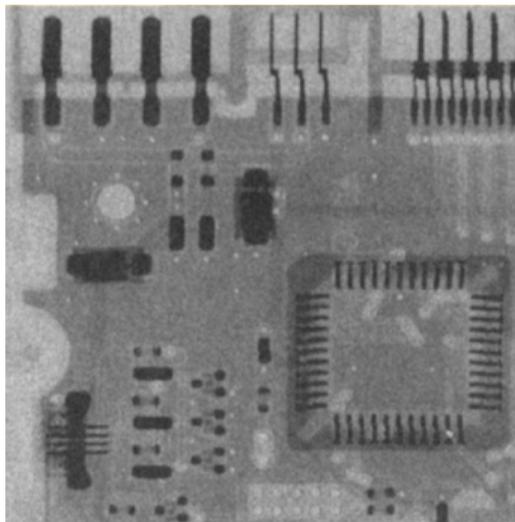
$$K_p = \min\left(\frac{\sigma_\nu}{\sigma_{V(p)}}, 1\right) \longrightarrow \begin{cases} \text{hypothèse raisonnable : } \sigma_{V(p)} \geq \sigma_\nu & K_p \in [0, 1] \\ \text{bruit maximal : } \sigma_{V(p)} \simeq \sigma_\nu & K_p \rightarrow 1 \\ \text{arêtes, texture, etc. : } \sigma_{V(p)} \gg \sigma_\nu & K_p \rightarrow 0 \end{cases}$$

- ▶ estimation de  $\sigma_\nu$  possible sur une **zone uniforme** ou par **essais successifs**
  - ▶ *comptage du nombre de valeurs incorrectes* :  $\sigma_{V(p)} < \sigma_\nu$
  - ▶ *trop important* :  $\sigma_\nu$  trop grand  $\Rightarrow$  filtrage excessif
  - ▶  $\sigma_\nu$  trop faible  $\Rightarrow$  filtrage insuffisant
- ▶ possibilité de fixer  $\sigma_\nu$  relativement à un **pourcentage** d'erreurs acceptées

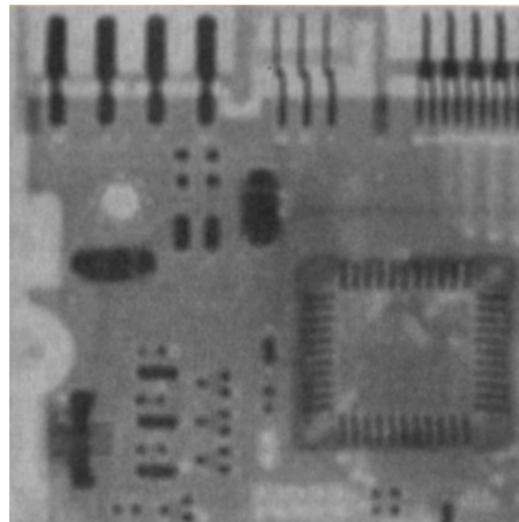
- ▶ exemple : comparaison des moyennes arithmétiques et statistique adaptatif



$I$



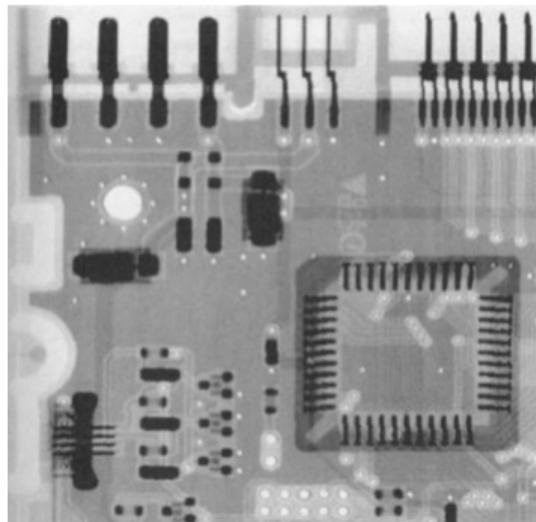
$I + \nu$



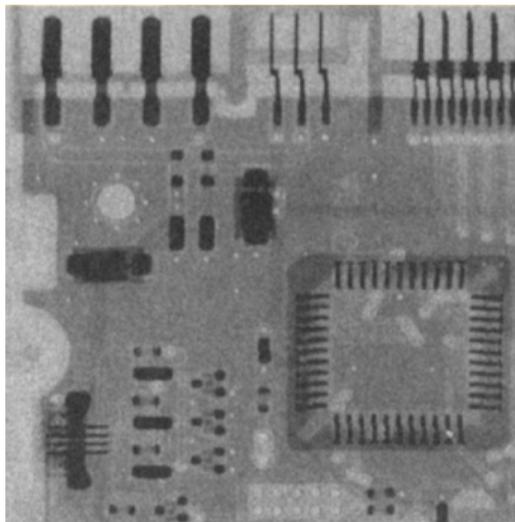
arithmétique

*Image rayon X d'une carte électronique – Gonzalez & Wood, DIP 4<sup>ed</sup> © J.E.Pascente, Lixi Inc.*

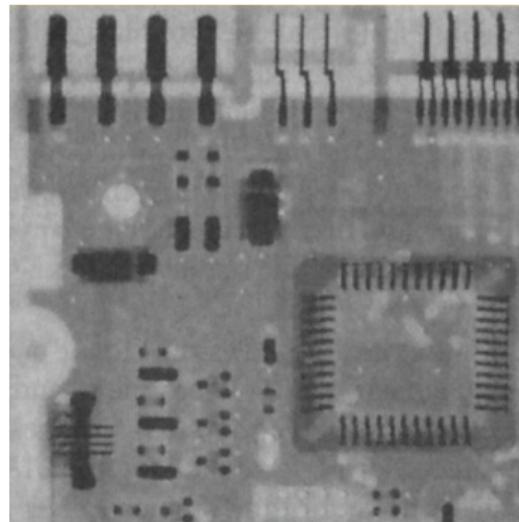
- ▶ exemple : comparaison des moyennes arithmétiques et statistique adaptatif



$I$



$I + \nu$



statistique adaptatif

*Image rayon X d'une carte électronique – Gonzalez & Wood, DIP 4<sup>ed</sup> © J.E.Pascente, Lixi Inc.*

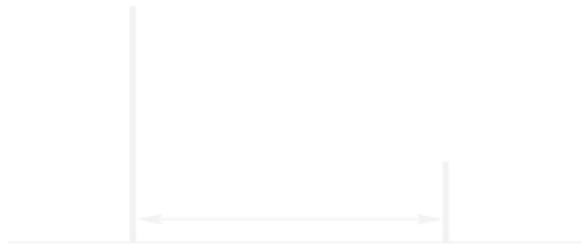
## 2.4. Filtre bilatéral

### ► lissage gaussien adaptatif de la ligne $i$ :

► moyenne des voisins sur  $[(i, j - L)..(i, j + L)]$  avec une double pondération

►  $\alpha_1$  et  $\alpha_2$  définis au moyen de gaussiennes sur  $[0, 1]$  :  $G_\sigma(x) = e^{-\frac{x^2}{2\sigma^2}}$

$$I'(i, j) = K_{i,j} \sum_{k=-L}^{+L} \alpha_1(k) \alpha_2(i, j, k) I(i, j - k)$$



►  $\alpha_1$  décroît en fonction de la distance :

$$\alpha_1(k) = G_{\sigma_1}(k)$$

►  $\alpha_2$  décroît en fonction de la différence d'amplitude :

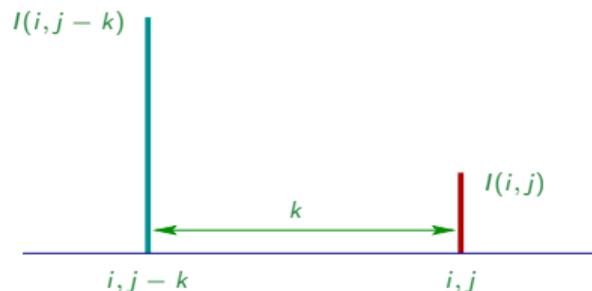
$$\alpha_2(i, j, k) = G_{\sigma_2}(|I(i, j - k) - I(i, j)|)$$

► normalisation des coefficients :  $K_{i,j}$  est l'inverse de la somme de tous les produits  $\alpha_1\alpha_2$

## 2.4. Filtre bilatéral

- ▶ lissage gaussien adaptatif de la ligne  $i$  :
  - ▶ moyenne des voisins sur  $[(i, j - L)..(i, j + L)]$  avec une double pondération
  - ▶  $\alpha_1$  et  $\alpha_2$  définis au moyen de gaussiennes sur  $[0, 1]$  :  $G_\sigma(x) = e^{-\frac{x^2}{2\sigma^2}}$

$$I'(i, j) = K_{i, j} \sum_{k=-L}^{+L} \alpha_1(k) \alpha_2(i, j, k) I(i, j - k)$$



- ▶  $\alpha_1$  décroît en fonction de la distance :

$$\alpha_1(k) = G_{\sigma_1}(k)$$

- ▶  $\alpha_2$  décroît en fonction de la différence d'amplitude :

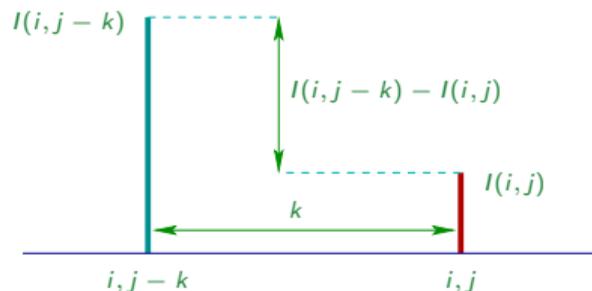
$$\alpha_2(i, j, k) = G_{\sigma_2}(|I(i, j - k) - I(i, j)|)$$

- ▶ normalisation des coefficients :  $K_{i, j}$  est l'inverse de la somme de tous les produits  $\alpha_1\alpha_2$

## 2.4. Filtre bilatéral

- ▶ lissage gaussien adaptatif de la ligne  $i$  :
  - ▶ moyenne des voisins sur  $[(i, j - L)..(i, j + L)]$  avec une double pondération
  - ▶  $\alpha_1$  et  $\alpha_2$  définis au moyen de gaussiennes sur  $[0, 1]$  :  $G_\sigma(x) = e^{-\frac{x^2}{2\sigma^2}}$

$$I'(i, j) = K_{i,j} \sum_{k=-L}^{+L} \alpha_1(k) \alpha_2(i, j, k) I(i, j - k)$$



- ▶  $\alpha_1$  décroît en fonction de la distance :

$$\alpha_1(k) = G_{\sigma_1}(k)$$

- ▶  $\alpha_2$  décroît en fonction de la différence d'amplitude :

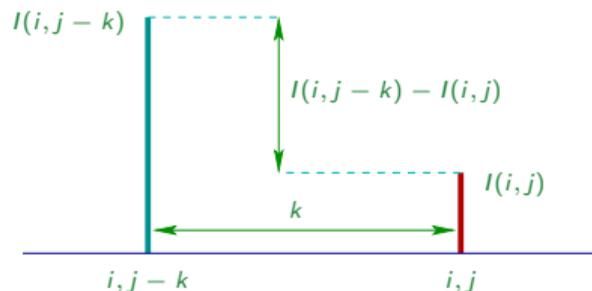
$$\alpha_2(i, j, k) = G_{\sigma_2}(|I(i, j - k) - I(i, j)|)$$

- ▶ normalisation des coefficients :  $K_{i,j}$  est l'inverse de la somme de tous les produits  $\alpha_1\alpha_2$

## 2.4. Filtre bilatéral

- ▶ lissage gaussien adaptatif de la ligne  $i$  :
  - ▶ moyenne des voisins sur  $[(i, j - L)..(i, j + L)]$  avec une double pondération
  - ▶  $\alpha_1$  et  $\alpha_2$  définis au moyen de gaussiennes sur  $[0, 1]$  :  $G_\sigma(x) = e^{-\frac{x^2}{2\sigma^2}}$

$$I'(i, j) = K_{i, j} \sum_{k=-L}^{+L} \alpha_1(k) \alpha_2(i, j, k) I(i, j - k)$$



- ▶  $\alpha_1$  décroît en fonction de la distance :

$$\alpha_1(k) = G_{\sigma_1}(k)$$

- ▶  $\alpha_2$  décroît en fonction de la différence d'amplitude :

$$\alpha_2(i, j, k) = G_{\sigma_2}(|I(i, j - k) - I(i, j)|)$$

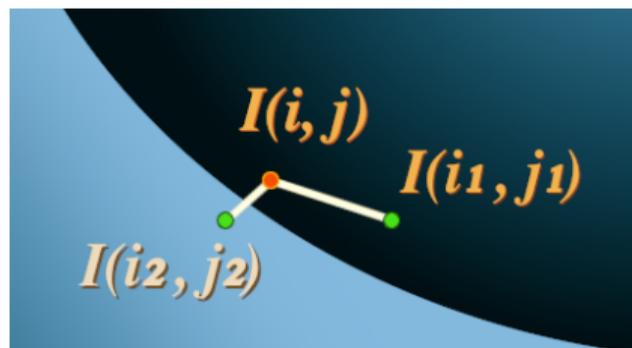
- ▶ normalisation des coefficients :  $K_{i, j}$  est l'inverse de la somme de tous les produits  $\alpha_1\alpha_2$

- ▶ le filtre dépend de deux paramètres :
  - ▶  $\sigma_1$  :  $\alpha_1$ , pondération selon la **distance**
  - ▶  $\sigma_2$  :  $\alpha_2$ , pondération selon l'**amplitude** : adaptation locale aux contours

- ▶ hypothèse :
  - ▶ **contour significatif** : variation forte d'amplitude
  - ▶ **bruit** : faible variation d'amplitude

- ▶ paramétrage du filtre bilatéral :
  - ▶ rôle de  $\sigma_1$  : réduction du bruit
  - ▶ rôle de  $\sigma_2$  : préservation des contours

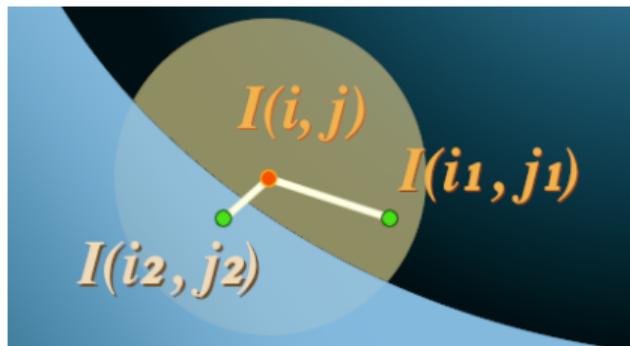
- ▶  $\sigma_2 \rightarrow 0$  : *augmente l'influence de la pondération selon l'amplitude : préservation des contours*
- ▶  $\sigma_2 \rightarrow +\infty$  : *réduit l'influence de la pondération selon l'amplitude et tend vers un filtrage gaussien d'écart type  $\sigma_1$*



*différence d'amplitude*

- faible pour  $(i_1, j_1)$  :  $\alpha_2 \nearrow$
- élevée pour  $(i_2, j_2)$  :  $\alpha_2 \searrow$

- ▶ le filtre dépend de deux paramètres :
  - ▶  $\sigma_1 : \alpha_1$ , pondération selon la **distance**
  - ▶  $\sigma_2 : \alpha_2$ , pondération selon l'**amplitude** : adaptation locale aux contours
- ▶ hypothèse :
  - ▶ **contour significatif** : variation forte d'amplitude
  - ▶ **bruit** : faible variation d'amplitude
- ▶ paramétrage du filtre bilatéral :
  - ▶ rôle de  $\sigma_1$  : réduction du bruit
  - ▶ rôle de  $\sigma_2$  : préservation des contours
    - ▶  $\sigma_2 \rightarrow 0$  : augmente l'influence de la pondération selon l'amplitude : préservation des contours
    - ▶  $\sigma_2 \rightarrow +\infty$  : réduit l'influence de la pondération selon l'amplitude et tend vers un filtrage gaussien d'écart type  $\sigma_1$



différence d'amplitude

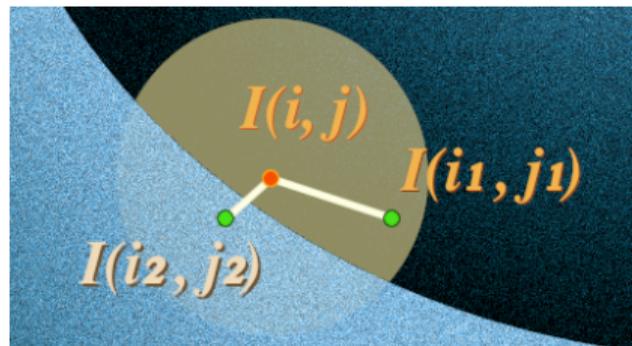
- faible pour  $(i_1, j_1)$  :  $\alpha_2 \nearrow$
- élevée pour  $(i_2, j_2)$  :  $\alpha_2 \searrow$

- ▶ le filtre dépend de deux paramètres :
  - ▶  $\sigma_1$  :  $\alpha_1$ , pondération selon la **distance**
  - ▶  $\sigma_2$  :  $\alpha_2$ , pondération selon l'**amplitude** : adaptation locale aux contours

- ▶ hypothèse :
  - ▶ **contour significatif** : variation forte d'amplitude
  - ▶ **bruit** : faible variation d'amplitude

- ▶ paramétrage du filtre bilatéral :
  - ▶ rôle de  $\sigma_1$  : réduction du bruit
  - ▶ rôle de  $\sigma_2$  : préservation des contours

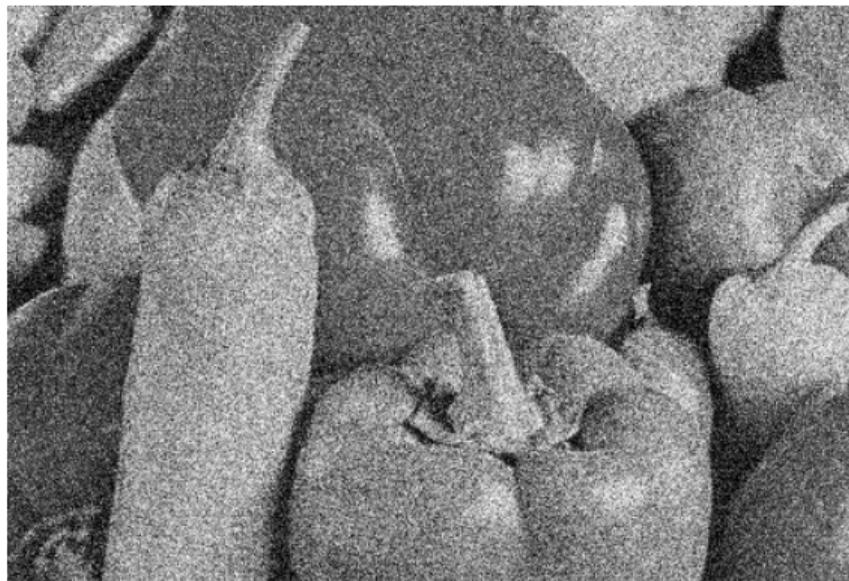
- ▶  $\sigma_2 \rightarrow 0$  : augmente l'influence de la pondération selon l'amplitude : préservation des contours
- ▶  $\sigma_2 \rightarrow +\infty$  : réduit l'influence de la pondération selon l'amplitude et **tend vers un filtrage gaussien** d'écart type  $\sigma_1$



*différence d'amplitude*

- faible pour  $(i_1, j_1)$  :  $\alpha_2 \nearrow$
- élevée pour  $(i_2, j_2)$  :  $\alpha_2 \searrow$

► exemples de débruitage avec le filtre bilatéral

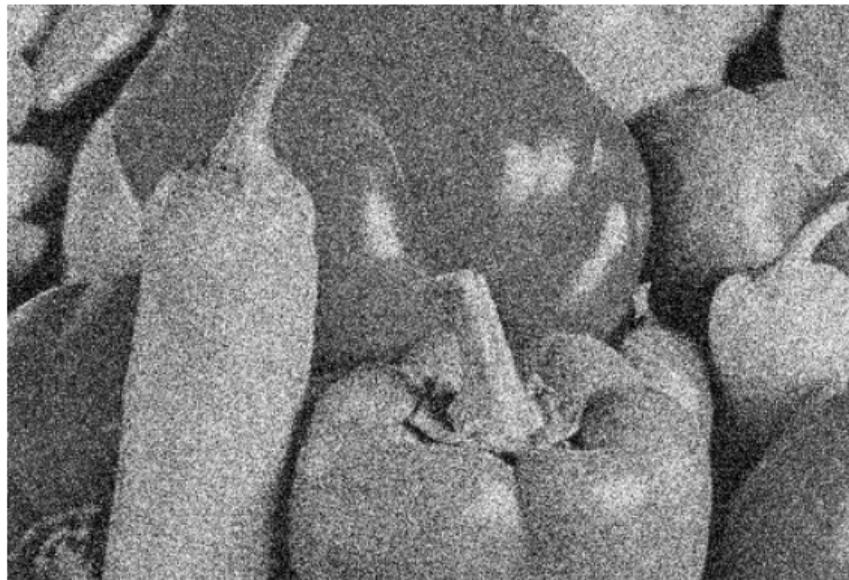


$I + \nu$



$\sigma_1 = 6$  et  $\sigma_2 = 90$

► exemples de débruitage avec le filtre bilatéral

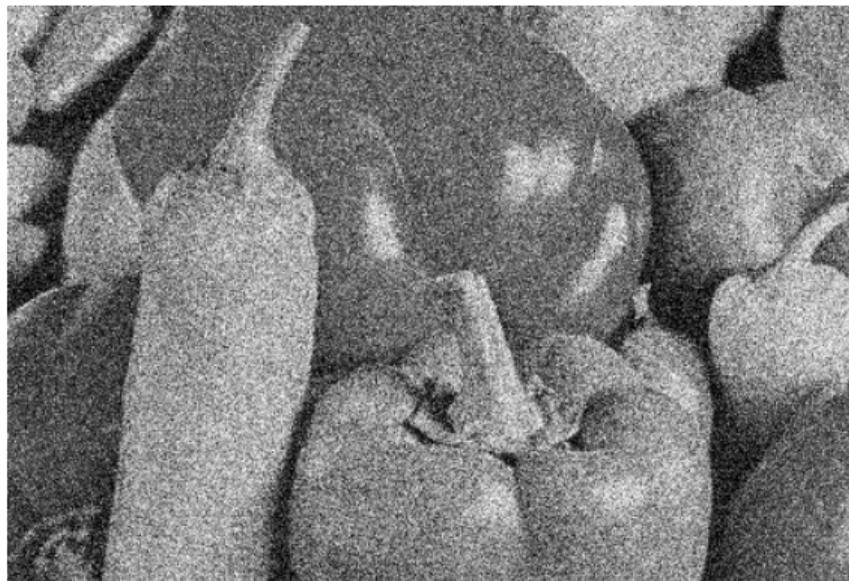


$I + \nu$

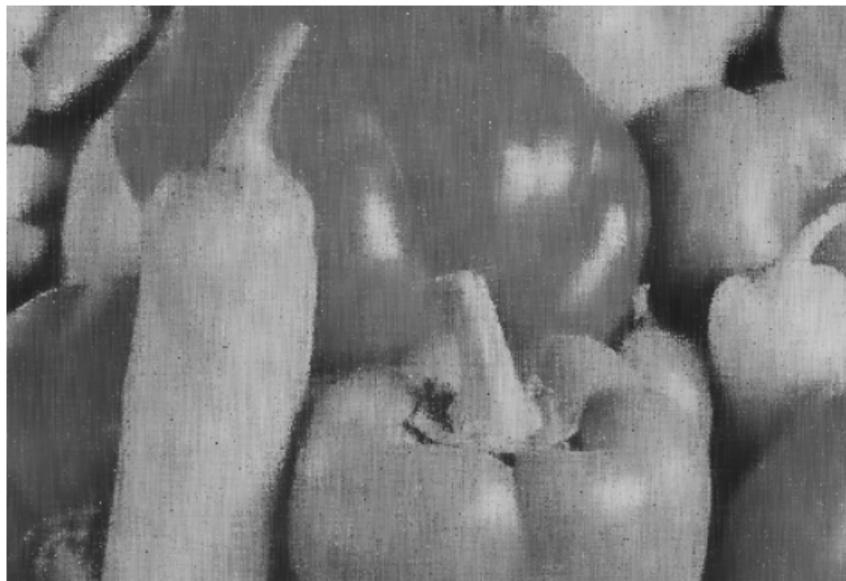


$\sigma_1 = 6$  et  $\sigma_2 = 50$

► exemples de débruitage avec le filtre bilatéral

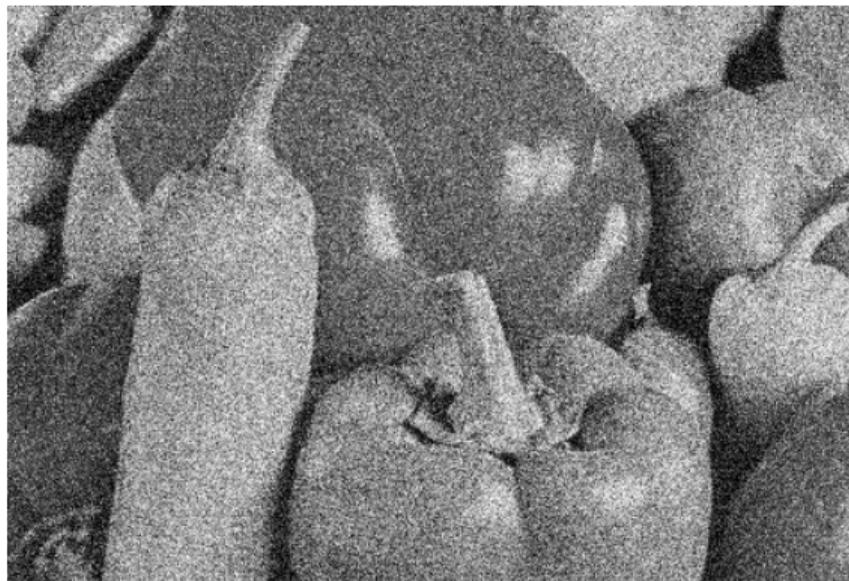


$I + \nu$

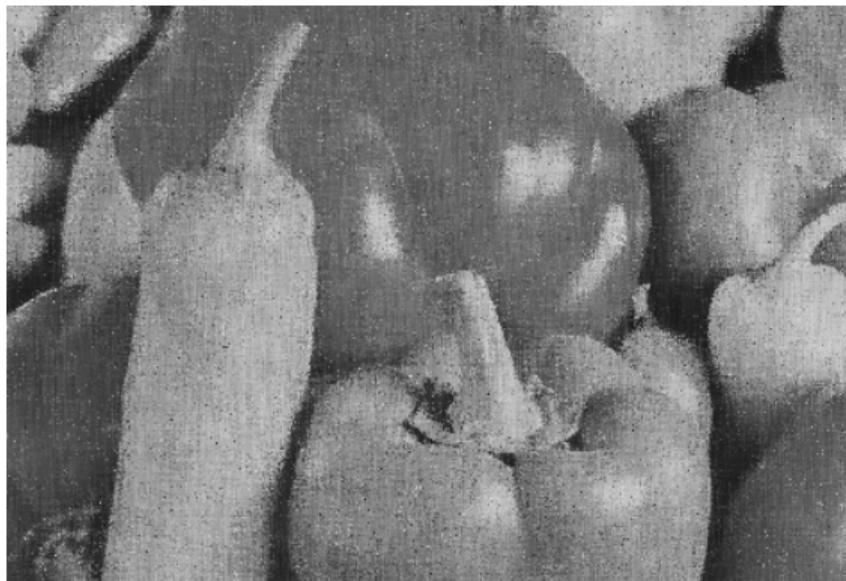


$\sigma_1 = 6$  et  $\sigma_2 = 40$

► exemples de débruitage avec le filtre bilatéral



$I + \nu$



$\sigma_1 = 6$  et  $\sigma_2 = 30$

## ★ Traitement d'une image bruitée



1

## ★ Traitement d'une image bruitée



2

## ★ Traitement d'une image bruitée



3

## ★ Traitement d'une image bruitée



4

# ★ Traitement d'une image bruitée



1

2

3

4

	A	B	C	D
1 → 2	R	R	G	F
2 → 3	G	F	F	G
3 → 4	F	G	R	R

*F* : filtrage fréquentiel

*G* : filtrage gaussien bilatéral

*R* : retinex

# ★ Traitement d'une image bruitée



1

2

3

4

	A	B	C	D
1 → 2	R	R	G	F
2 → 3	G	F	F	G
3 → 4	F	G	R	R

*F* : filtrage fréquentiel

*G* : filtrage gaussien bilatéral

*R* : retinex

# ★ Traitement d'une image bruitée



1

2

3

4

	A	B	C	D
1 → 2	R	R	G	F
2 → 3	G	F	F	G
3 → 4	F	G	R	R

*F* : filtrage fréquentiel

*G* : filtrage gaussien bilatéral

*R* : retinex

# 3. Traitement du bruit impulsif

## 3.1. Principe des filtres d'ordre

- ▶ bruit impulsif : impacte **seulement un sous-ensemble** de pixels
  - ▶ autour de chaque pixel : combinaison de pixels corrects et bruités
  - ▶ principe : identifier et remplacer les **valeurs extrêmes** (potentiellement aberrantes)
  - ▶ moyenne pondérée sur un voisinage **trié**
- ▶ **moyenne tronquée** :
  - ▶  $\{i_1, \dots, i_n\}$  : ensemble valeurs des pixels sur un voisinage de  $p$
  - ▶  $\{i_{(1)}, \dots, i_{(n)}\}$  : suite des valeurs **triées**
  - ▶  $\{c_1, \dots, c_n\}$  tels que  $\sum_{k=1}^n c_k = 1$

$$I'(p) = \sum_{k=1}^n c_k i_{(k)} \quad \text{avec} \quad c_i = \underbrace{\{0, \dots, 0\}}_{\frac{n-p}{2}}, \underbrace{\{1/p, \dots, 1/p\}}_p, \underbrace{\{0, \dots, 0\}}_{\frac{n-p}{2}}$$

- ▶ en pratique :
  - ▶ filtre **médian** ( $p = 1$ ) : on ne conserve que la valeur médiane  $i_{(\frac{n+1}{2})}$
  - ▶ filtre **olympique** ( $p = n - 2$ ) : on supprime seulement les valeurs extrêmes  $i_{(1)}$  et  $i_{(n)}$
  - ▶ le médian est ses variantes demeure le plus utilisé

# 3. Traitement du bruit impulsionnel

## 3.1. Principe des filtres d'ordre

- ▶ bruit impulsionnel : impacte **seulement un sous-ensemble** de pixels
  - ▶ autour de chaque pixel : combinaison de pixels corrects et bruités
  - ▶ principe : identifier et remplacer les **valeurs extrêmes** (potentiellement aberrantes)
  - ▶ moyenne pondérée sur un voisinage **trié**
- ▶ **moyenne tronquée** :
  - ▶  $\{i_1, \dots, i_n\}$  : ensemble valeurs des pixels sur un voisinage de  $p$
  - ▶  $(i_{(1)}, \dots, i_{(n)})$  : suite des valeurs **triées**
  - ▶  $\{c_1, \dots, c_n\}$  tels que  $\sum_{k=1}^n c_k = 1$

$$I'(p) = \sum_{k=1}^n c_k i_{(k)} \quad \text{avec} \quad c_i = \underbrace{\{0, \dots, 0\}}_{\frac{n-p}{2}}, \underbrace{\{1/p, \dots, 1/p\}}_p, \underbrace{\{0, \dots, 0\}}_{\frac{n-p}{2}}$$

- ▶ en pratique :
  - ▶ filtre **médian** ( $p = 1$ ) : on ne conserve que la valeur médiane  $i_{(\frac{n+1}{2})}$
  - ▶ filtre **olympique** ( $p = n - 2$ ) : on supprime seulement les valeurs extrêmes  $i_{(1)}$  et  $i_{(n)}$
  - ▶ le médian est ses variantes demeure le plus utilisé

# 3. Traitement du bruit impulsionnel

## 3.1. Principe des filtres d'ordre

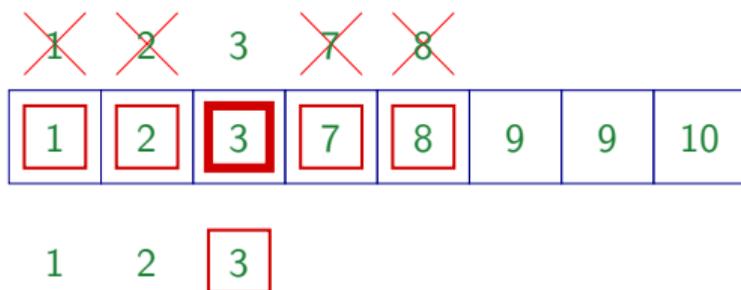
- ▶ bruit impulsionnel : impacte **seulement un sous-ensemble** de pixels
  - ▶ autour de chaque pixel : combinaison de pixels corrects et bruités
  - ▶ principe : identifier et remplacer les **valeurs extrêmes** (potentiellement aberrantes)
  - ▶ moyenne pondérée sur un voisinage **trié**
- ▶ **moyenne tronquée** :
  - ▶  $\{i_1, \dots, i_n\}$  : ensemble valeurs des pixels sur un voisinage de  $p$
  - ▶  $(i_{(1)}, \dots, i_{(n)})$  : suite des valeurs **triées**
  - ▶  $\{c_1, \dots, c_n\}$  tels que  $\sum_{k=1}^n c_k = 1$

$$I'(p) = \sum_{k=1}^n c_k i_{(k)} \quad \text{avec} \quad c_i = \underbrace{\{0, \dots, 0\}}_{\frac{n-p}{2}}, \underbrace{\{1/p, \dots, 1/p\}}_p, \underbrace{\{0, \dots, 0\}}_{\frac{n-p}{2}}$$

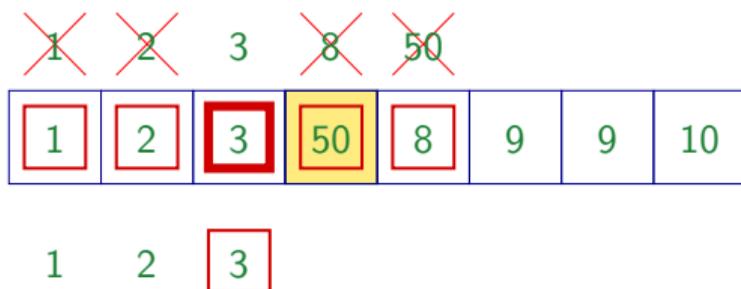
- ▶ en pratique :
  - ▶ filtre **médian** ( $p = 1$ ) : on ne conserve que la valeur médiane  $i_{(\frac{n+1}{2})}$
  - ▶ filtre **olympique** ( $p = n - 2$ ) : on supprime seulement les valeurs extrêmes  $i_{(1)}$  et  $i_{(n)}$
  - ▶ le médian et ses variantes demeure le plus utilisé

○ Exemples en dimension 1 : médian de taille 5

▶ sans bruit impulsif

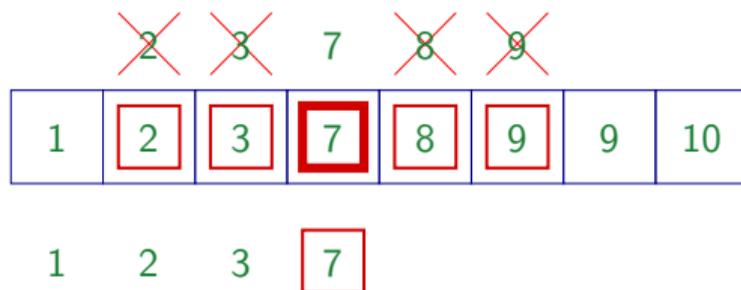


▶ avec bruit impulsif

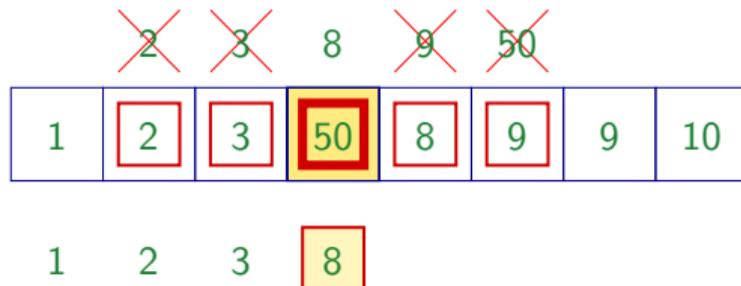


○ Exemples en dimension 1 : médian de taille 5

▶ sans bruit impulsif

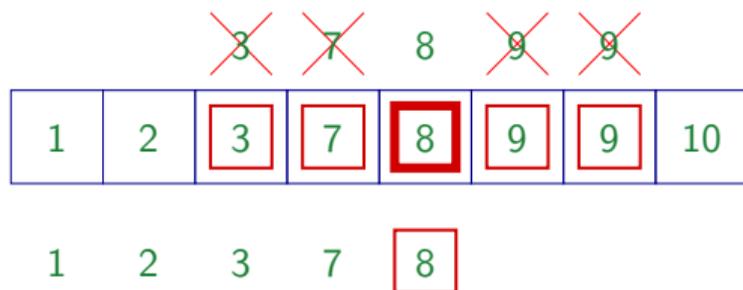


▶ avec bruit impulsif

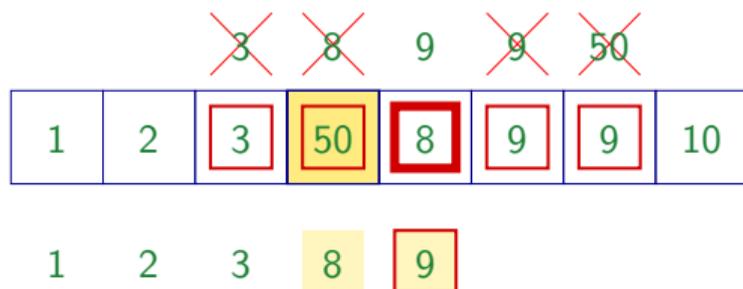


○ Exemples en dimension 1 : médian de taille 5

▶ sans bruit impulsif

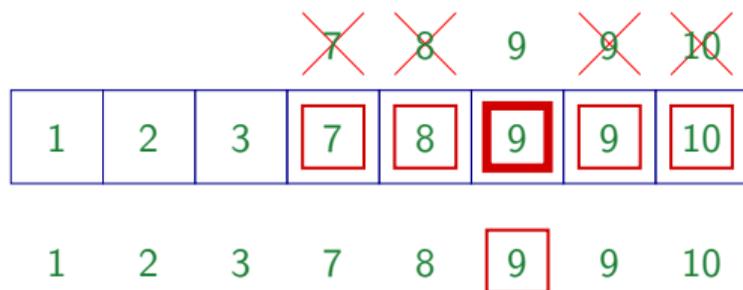


▶ avec bruit impulsif

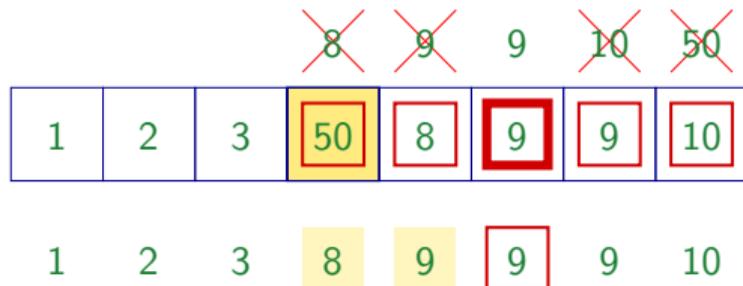


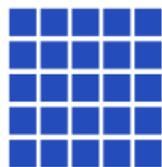
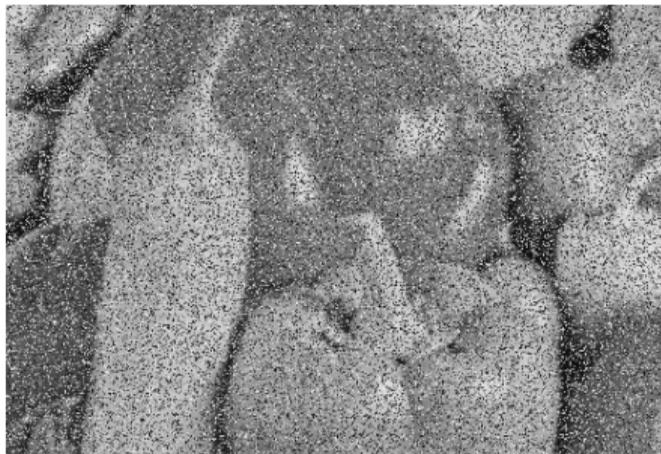
○ Exemples en dimension 1 : médian de taille 5

▶ sans bruit impulsif



▶ avec bruit impulsif

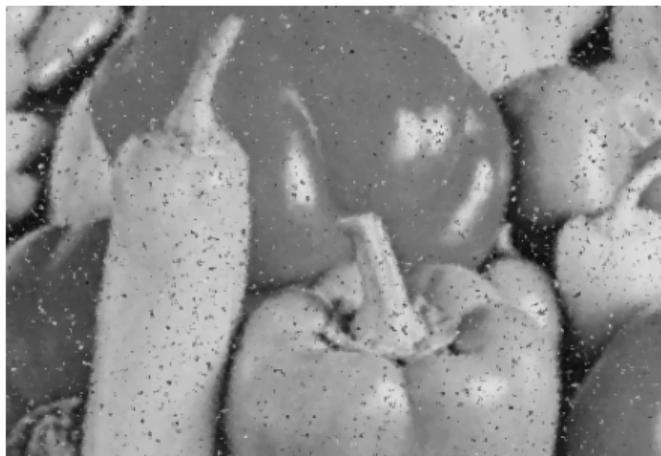




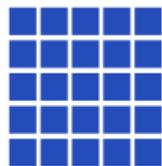
domaine carré



domaine hexagonal



*carré  
largeur 3*



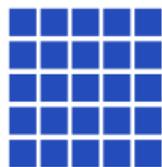
domaine carré



domaine hexagonal



*carré  
largeur 5*



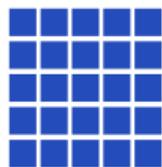
domaine carré



domaine hexagonal



*carré  
largeur 7*



domaine carré

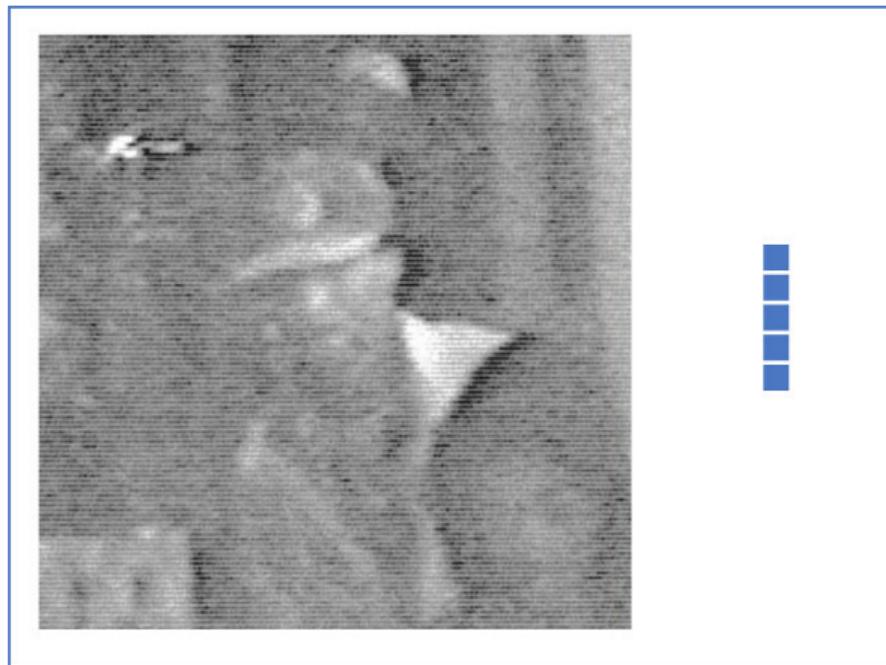


domaine hexagonal



*hexagonal  
largeur 7*

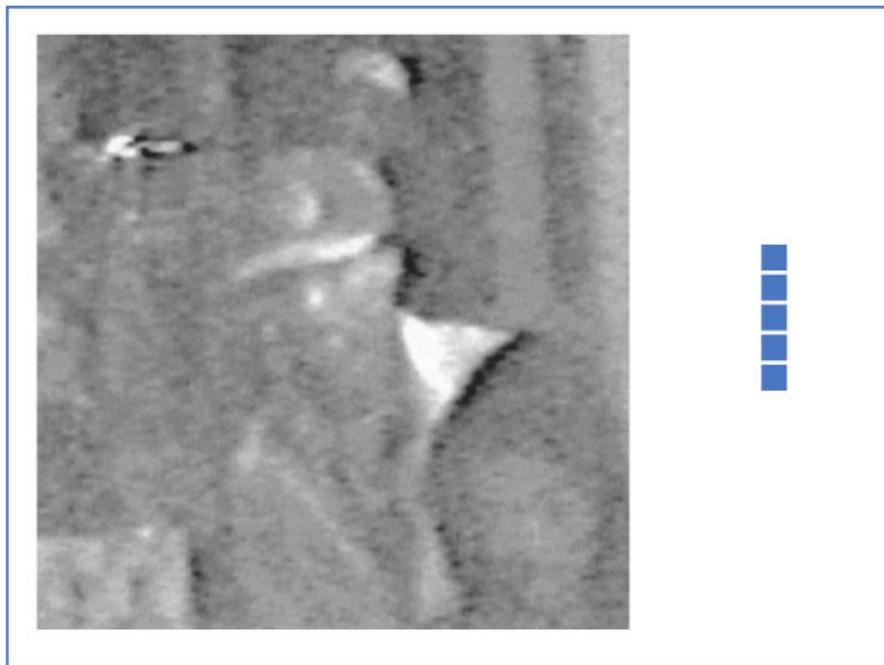
► possibilité de traitement directionnel



*Image de vidéosurveillance – J.C.Russ, Img. Process. Handbook 4<sup>ed</sup>*

- même problématique que pour les filtres moyenneurs : compromis bruit / détail
- plusieurs stratégies pour discriminer les hautes fréquences pertinentes

- ▶ possibilité de traitement directionnel



*Image de vidéosurveillance – J.C.Russ, Img. Process. Handbook 4<sup>ed</sup>*

- ▶ même problématique que pour les filtres moyenneurs : compromis bruit /détail
- ▶ plusieurs stratégies pour discriminer les hautes fréquences pertinentes

## 3.2. Filtre médian pondéré

- ▶ objectif : réduire les modifications injustifiées
  - ▶ exemple en 1D avec un voisinage de taille 5 :

$$\{ 10 \ 90 \ 15 \ 7 \ 5 \} \rightarrow \{ 5 \ 7 \ 10 \ 15 \ 90 \} \rightarrow 10$$

- ▶ stratégie : pondération par réplication des valeurs sur le voisinage
  - ▶ réplication des valeurs en fonction de coefficients  $k_1 \dots k_n$  :

$$k \diamond x : x \rightarrow \overbrace{\{x, \dots, x\}}^k$$

- ▶ exemple de coefficients sur un voisinage de taille 5 :  $\{1, 2, 3, 2, 1\}$

$$\begin{aligned} \{ 1 \diamond 10 \ 2 \diamond 90 \ 3 \diamond 15 \ 2 \diamond 7 \ 1 \diamond 5 \} &\rightarrow \{ 10 \ 90 \ 90 \ 15 \ 15 \ 15 \ 7 \ 7 \ 5 \} \\ &\rightarrow \{ 5 \ 7 \ 7 \ 10 \ 15 \ 15 \ 15 \ 90 \ 90 \} \\ &\rightarrow 15 \end{aligned}$$

## 3.2. Filtre médian pondéré

- ▶ objectif : réduire les modifications injustifiées
  - ▶ exemple en 1D avec un voisinage de taille 5 :

$$\{ 10 \ 90 \ 15 \ 7 \ 5 \} \rightarrow \{ 5 \ 7 \ 10 \ 15 \ 90 \} \rightarrow 10$$

- ▶ stratégie : pondération par réplication des valeurs sur le voisinage
  - ▶ réplication des valeurs en fonction de coefficients  $k_1 \dots k_n$  :

$$k \diamond x : x \rightarrow \overbrace{\{x, \dots, x\}}^k$$

- ▶ exemple de coefficients sur un voisinage de taille 5 :  $\{1, 2, 3, 2, 1\}$

$$\begin{aligned} \{ 1 \diamond 10 \ 2 \diamond 90 \ 3 \diamond 15 \ 2 \diamond 7 \ 1 \diamond 5 \} &\rightarrow \{ 10 \ 90 \ 90 \ 15 \ 15 \ 15 \ 7 \ 7 \ 5 \} \\ &\rightarrow \{ 5 \ 7 \ 7 \ 10 \ 15 \ 15 \ 15 \ 90 \ 90 \} \\ &\rightarrow 15 \end{aligned}$$

## 3.2. Filtre médian pondéré

- ▶ objectif : réduire les modifications injustifiées
  - ▶ exemple en 1D avec un voisinage de taille 5 :

$$\{ 10 \text{ 90 } 15 \text{ 7 } 5 \} \rightarrow \{ 5 \text{ 7 } 10 \text{ 15 } 90 \} \rightarrow 10$$

- ▶ stratégie : pondération par réplication des valeurs sur le voisinage
  - ▶ réplication des valeurs en fonction de coefficients  $k_1 \dots k_n$  :

$$k \diamond x : x \rightarrow \overbrace{\{x, \dots, x\}}^k$$

- ▶ exemple de coefficients sur un voisinage de taille 5 :  $\{1, 2, 3, 2, 1\}$

$$\begin{aligned} \{ 1 \diamond 10 \text{ 2} \diamond 90 \text{ 3} \diamond 15 \text{ 2} \diamond 7 \text{ 1} \diamond 5 \} &\rightarrow \{ 10 \text{ 90 } 90 \text{ 15 } 15 \text{ 15 } 7 \text{ 7 } 5 \} \\ &\rightarrow \{ 5 \text{ 7 } 7 \text{ 10 } 15 \text{ 15 } 15 \text{ 90 } 90 \} \\ &\rightarrow 15 \end{aligned}$$

## 3.2. Filtre médian pondéré

- ▶ objectif : réduire les modifications injustifiées
  - ▶ exemple en 1D avec un voisinage de taille 5 :

$$\{ 10 \ 90 \ 15 \ 7 \ 5 \} \rightarrow \{ 5 \ 7 \ 10 \ 15 \ 90 \} \rightarrow 10$$

- ▶ stratégie : pondération par réplication des valeurs sur le voisinage
  - ▶ réplication des valeurs en fonction de coefficients  $k_1 \dots k_n$  :

$$k \diamond x : x \rightarrow \overbrace{\{x, \dots, x\}}^k$$

- ▶ exemple de coefficients sur un voisinage de taille 5 :  $\{1, 2, 3, 2, 1\}$

$$\begin{aligned} \{ 1 \diamond 10 \ 2 \diamond 90 \ 3 \diamond 15 \ 2 \diamond 7 \ 1 \diamond 5 \} &\rightarrow \{ 10 \ 90 \ 90 \ 15 \ 15 \ 15 \ 7 \ 7 \ 5 \} \\ &\rightarrow \{ 5 \ 7 \ 7 \ 10 \ 15 \ 15 \ 15 \ 90 \ 90 \} \\ &\rightarrow 15 \end{aligned}$$

## 3.2. Filtre médian pondéré

- ▶ objectif : réduire les modifications injustifiées
  - ▶ exemple en 1D avec un voisinage de taille 5 :

$$\{ 10 \ 90 \ 15 \ 7 \ 5 \} \rightarrow \{ 5 \ 7 \ 10 \ 15 \ 90 \} \rightarrow 10$$

- ▶ stratégie : pondération par réplication des valeurs sur le voisinage
  - ▶ réplication des valeurs en fonction de coefficients  $k_1 \dots k_n$  :

$$k \diamond x : x \rightarrow \overbrace{\{x, \dots, x\}}^k$$

- ▶ exemple de coefficients sur un voisinage de taille 5 :  $\{1, 2, 3, 2, 1\}$

$$\begin{aligned} \{ 1 \diamond 10 \ 2 \diamond 90 \ 3 \diamond 15 \ 2 \diamond 7 \ 1 \diamond 5 \} &\rightarrow \{ 10 \ 90 \ 90 \ 15 \ 15 \ 15 \ 7 \ 7 \ 5 \} \\ &\rightarrow \{ 5 \ 7 \ 7 \ 10 \ 15 \ 15 \ 15 \ 90 \ 90 \} \\ &\rightarrow 15 \end{aligned}$$

## 3.2. Filtre médian pondéré

- ▶ objectif : réduire les modifications injustifiées
  - ▶ exemple en 1D avec un voisinage de taille 5 :

$$\{ 10 \ 90 \ 15 \ 7 \ 5 \} \rightarrow \{ 5 \ 7 \ 10 \ 15 \ 90 \} \rightarrow 10$$

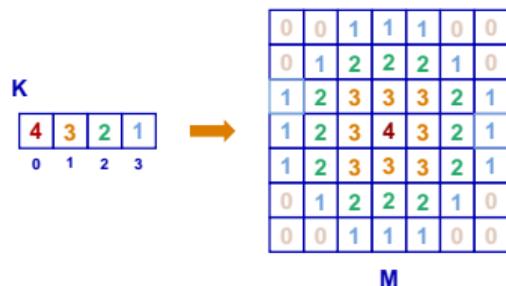
- ▶ stratégie : pondération par réplication des valeurs sur le voisinage
  - ▶ réplication des valeurs en fonction de coefficients  $k_1 \dots k_n$  :

$$k \diamond x : x \rightarrow \overbrace{\{x, \dots, x\}}^k$$

- ▶ exemple de coefficients sur un voisinage de taille 5 :  $\{1, 2, 3, 2, 1\}$

$$\begin{aligned} \{ 1 \diamond 10 \ 2 \diamond 90 \ 3 \diamond 15 \ 2 \diamond 7 \ 1 \diamond 5 \} &\rightarrow \{ 10 \ 90 \ 90 \ 15 \ 15 \ 15 \ 7 \ 7 \ 5 \} \\ &\rightarrow \{ 5 \ 7 \ 7 \ 10 \ 15 \ 15 \ 15 \ 90 \ 90 \} \\ &\rightarrow 15 \end{aligned}$$

- ▶ en dimension 2, on peut construire un tableau  $M$  de coefficients en fonction de la distance au centre du filtre
- ▶ exemple pour une largeur de filtre  $L = 3$  :



pour  $i \in [-3..3]$ , pour  $j \in [-3..3]$   

$$M[i,j] = K[\sqrt{i^2 + j^2}]$$

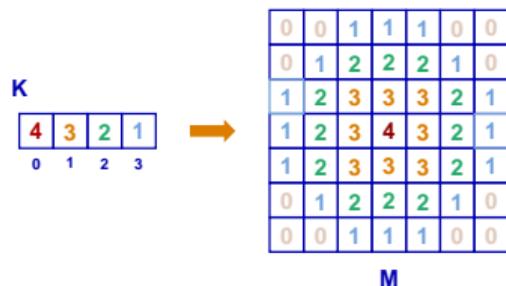
- ▶ filtrage de  $I[i,j]$  :

pour  $d_i \in [-L..L]$ , pour  $d_j \in [-L..L]$

contruction de la liste  $\{M[d_i, d_j] \diamond I[i + d_i, j + d_j]\}$

- ▶ remarque : si tous les coefficients du vecteur  $K$  valent 1, on retrouve le médian standard avec domaine hexagonal

- ▶ en dimension 2, on peut construire un tableau  $M$  de coefficients en fonction de la distance au centre du filtre
- ▶ exemple pour une largeur de filtre  $L = 3$  :



pour  $i \in [-3..3]$ , pour  $j \in [-3..3]$

$$M[i,j] = K[\sqrt{i^2 + j^2}]$$

- ▶ filtrage de  $I[i,j]$  :

pour  $d_i \in [-L..L]$ , pour  $d_j \in [-L..L]$

contruction de la liste  $\{M[d_i, d_j] \diamond I[i + d_i, j + d_j]\}$

- ▶ remarque : si tous les coefficients du vecteur  $K$  valent 1, on retrouve le médian standard avec domaine hexagonal

- ▶ *exemples de filtrages médian pondérés :*



*médian  
largeur 3*



*médian  
pondéré  
largeur 5*

- ▶ *exemples de filtrages médian pondérés :*



*médian  
largeur 5*



*médian  
pondéré  
largeur 7*

- ▶ *exemples de filtrages médian pondérés :*



*médian  
largeur 7*



*médian  
pondéré  
largeur 9*

### 3.3. Filtre multimédian

- ▶ combinaison de plusieurs filtrage : réduit le lissage
- ▶ première étape
  - application de deux ou quatre filtres médians sur des voisinages **directionnels**



- ▶ seconde étape
  - médian des valeurs calculées à l'étape 1 (deux ou quatre) et de la valeur du pixel à traiter

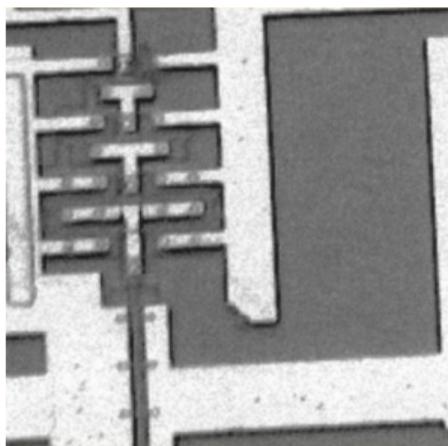
### 3.3. Filtre multimédian

- ▶ combinaison de plusieurs filtrage : réduit le lissage
- ▶ première étape
  - application de deux ou quatre filtres médians sur des voisinages **directionnels**

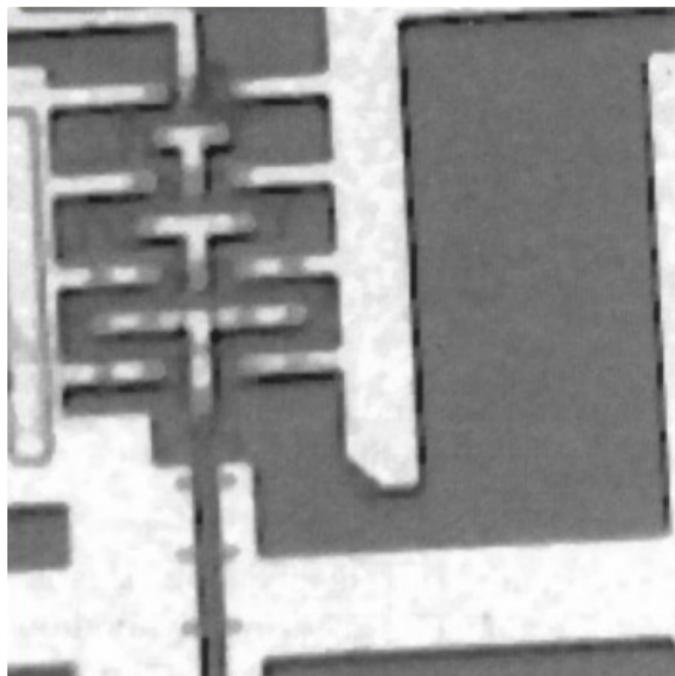
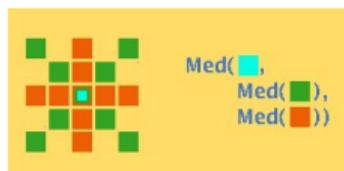


- ▶ seconde étape
  - médian des valeurs calculées à l'étape 1 (deux ou quatre) et de la valeur du pixel à traiter

► exemple de filtrage médian médian (multimédian) :

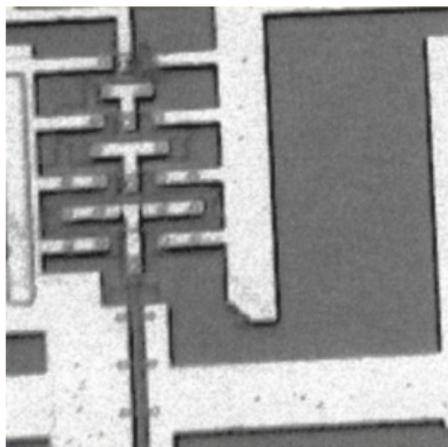


/

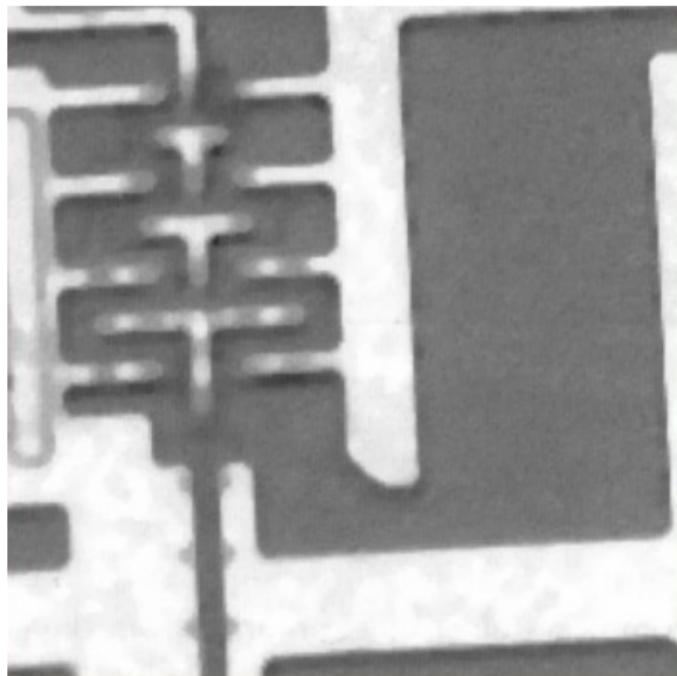
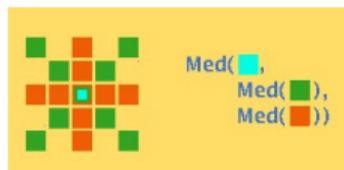


*médian largeur 3*

► exemple de filtrage médian médian (multimédian) :

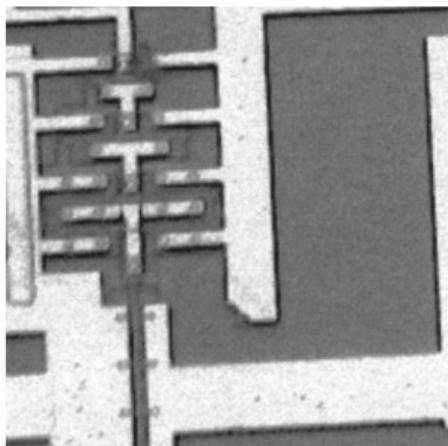


/

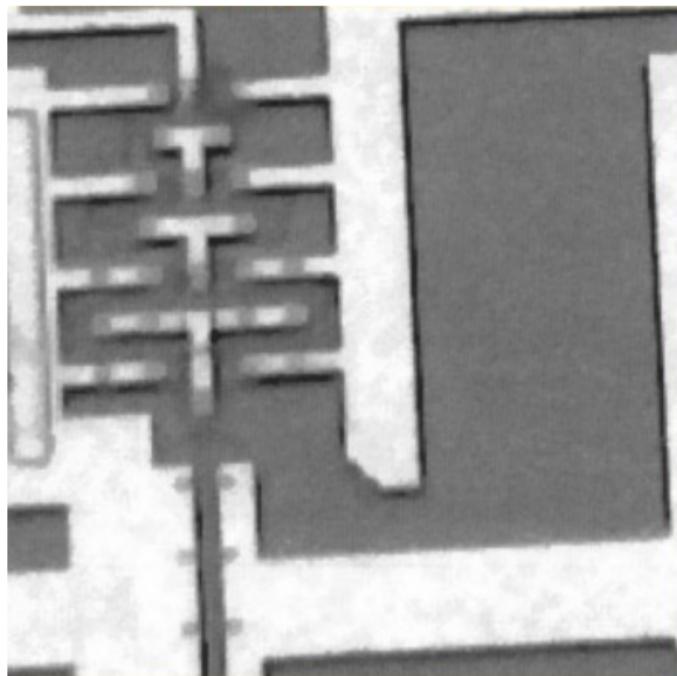
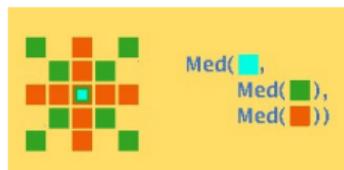


*médian largeur 5*

► exemple de filtrage médian médian (multimédian) :



/



*médian median largeur 5*

### 3.4. Filtre médian adaptatif

- ▶ hypothèse :
  - ▶ sur le voisinage de chaque pixel, le bruit
    - ▶ correspond soit à la valeur *minimale* soit à la valeur *maximale*
    - ▶ est distant de la médiane
- ▶ stratégie :
  - ▶ critère
    - ▶ un pixel  $p$  est filtré seulement si son intensité est *minimale* ou *maximale*
    - ▶ relativement à un voisinage adaptatif  $V(p)$
  - ▶ choix du voisinage
    - ▶  $V(p)$  est le plus petit voisinage t.q. la *médiane diffère* des valeurs minimale et maximale
    - ▶  $V(p)$  est de *taille bornée*

- ▶ si  $V(p)$  est un voisinage de  $p$ , on note
  - ▶  $med_V(p)$  la valeur médiane
  - ▶  $min_V(p)$  la valeur minimale
  - ▶  $max_V(p)$  la valeur maximale

*initialiser  $V(p)$  avec  $p$  et ses huit voisins*

*tant que  $med_V(p) = min_V(p)$  ou  $med_V(p) = max_V(p)$*

*{*

*accroître de 1 la taille de  $V$*

*si la taille de  $V$  est maximale*

*retourner  $l(p)$*

*}*

*si  $l(p) \neq min_V(p)$  et  $l(p) \neq max_V(p)$*

*retourner  $l(p)$*

*sinon*

*retourner  $med_V(p)$*

► exemple de filtrage médian adaptatif :

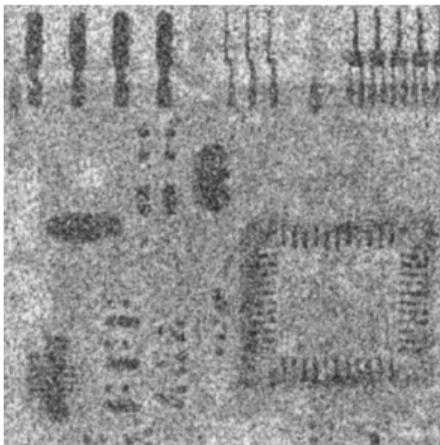
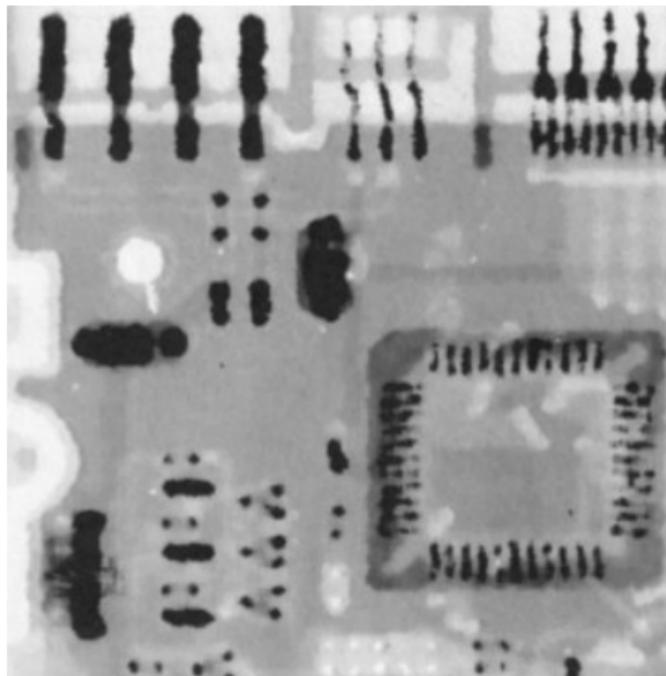
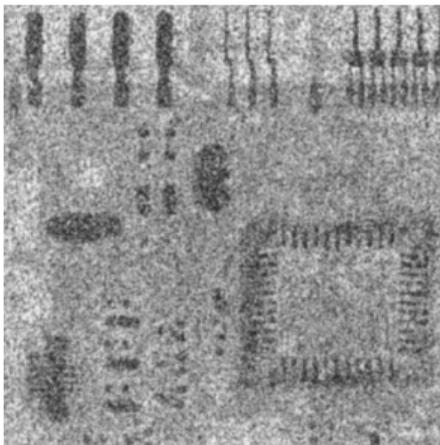


Image rayon X d'une carte électronique – Gonzalez & Wood,  
DIP 4<sup>ed</sup> © J.E.Pascente, Lixi Inc.

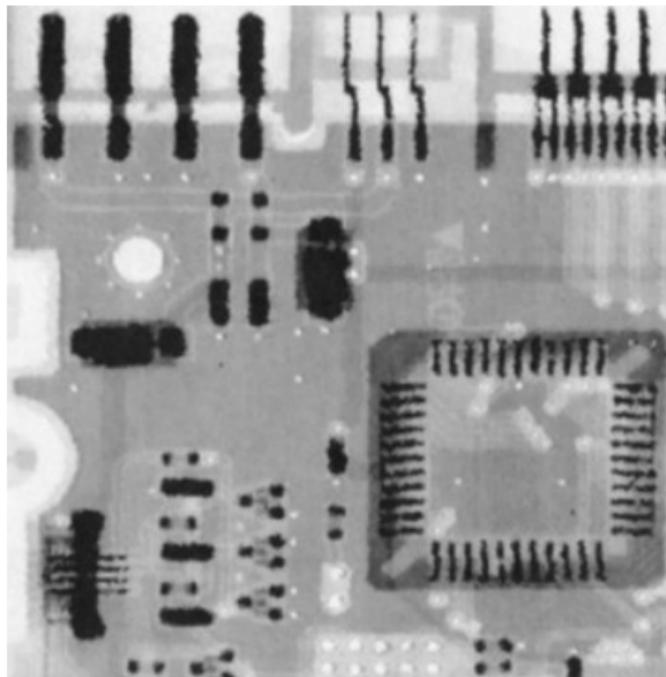


*médian largeur 7*

► exemple de filtrage médian adaptatif :



*Image rayon X d'une carte électronique – Gonzalez & Wood,  
DIP 4<sup>ed</sup> © J.E.Pascente, Lixi Inc.*

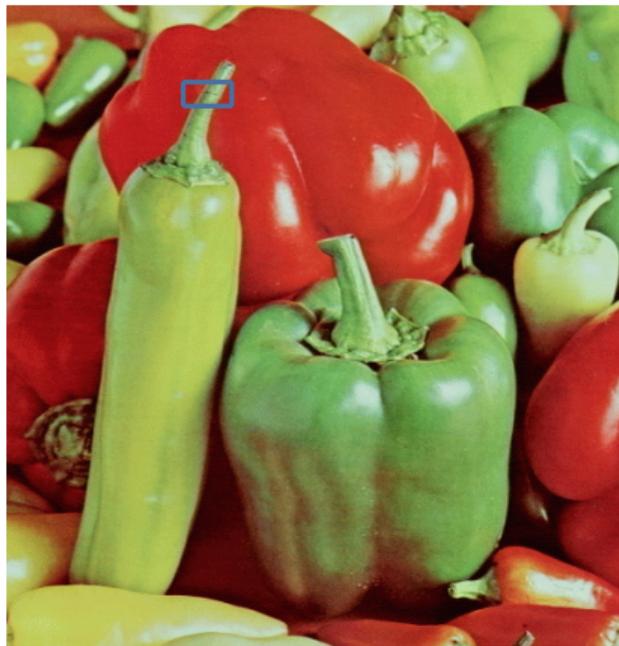


*médian adaptatif, taille maximale 7*

### 3.5. Filtrage médian d'images couleurs

#### ○ Filtrage marginal

- ▶ bruit de capteurs : pertinent de corriger chaque canal indépendamment
- ▶ mise en œuvre vectorielle possible avec la relation d'ordre entrelacée sur les couleurs

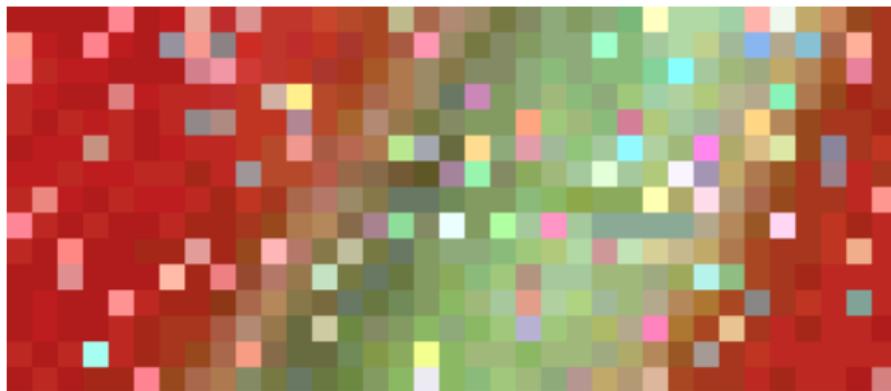
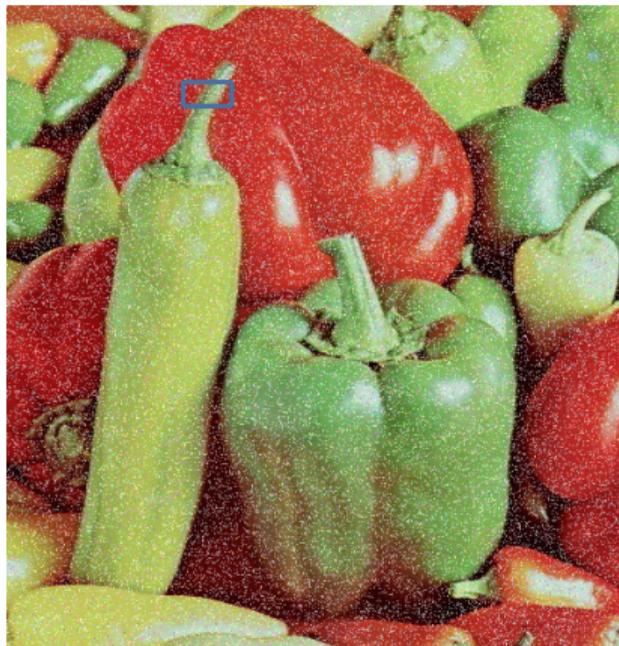


*image test*

### 3.5. Filtrage médian d'images couleurs

- Filtrage marginal

- ▶ bruit de capteurs : pertinent de corriger chaque canal indépendamment
- ▶ mise en œuvre vectorielle possible avec la relation d'ordre entrelacée sur les couleurs



*image test bruitée*

### 3.5. Filtrage médian d'images couleurs

- Filtrage marginal

- ▶ bruit de capteurs : pertinent de corriger chaque canal indépendamment
- ▶ mise en œuvre vectorielle possible avec la relation d'ordre entrelacée sur les couleurs

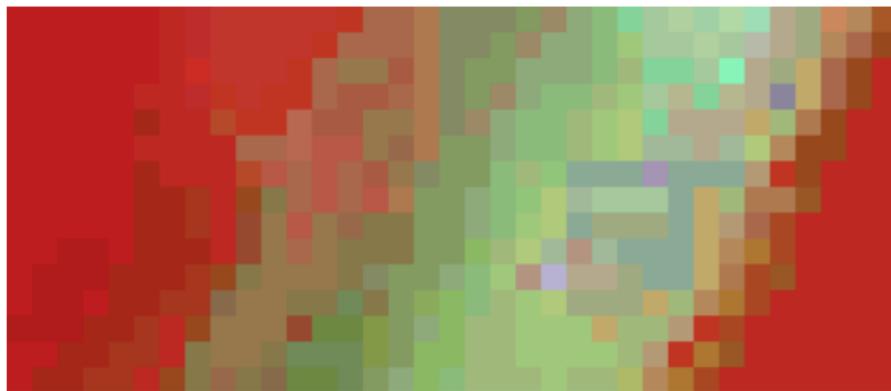


*médian  $3 \times 3$*

### 3.5. Filtrage médian d'images couleurs

#### ○ Filtrage marginal

- ▶ bruit de capteurs : pertinent de corriger chaque canal indépendamment
- ▶ mise en œuvre vectorielle possible avec la relation d'ordre entrelacée sur les couleurs

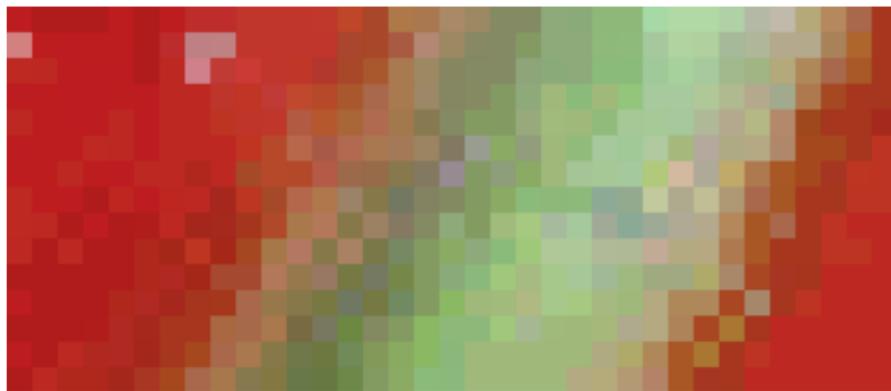
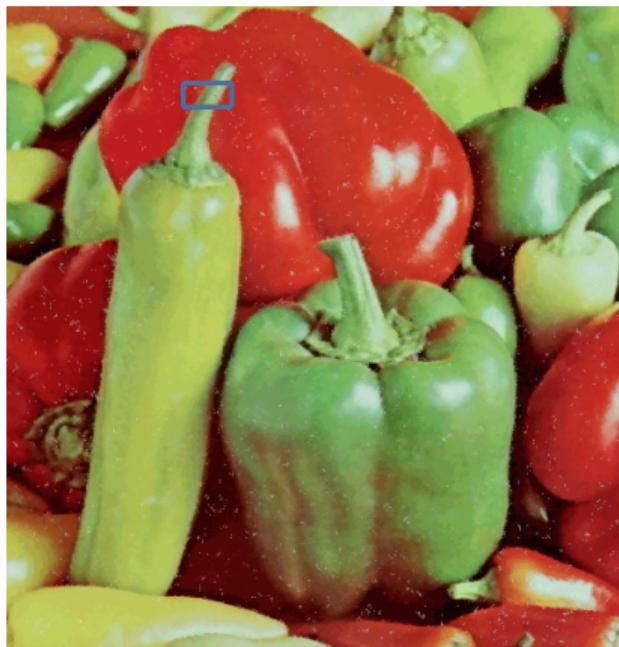


*médian  $5 \times 5$*

### 3.5. Filtrage médian d'images couleurs

#### ○ Filtrage marginal

- ▶ bruit de capteurs : pertinent de corriger chaque canal indépendamment
- ▶ mise en œuvre vectorielle possible avec la relation d'ordre entrelacée sur les couleurs

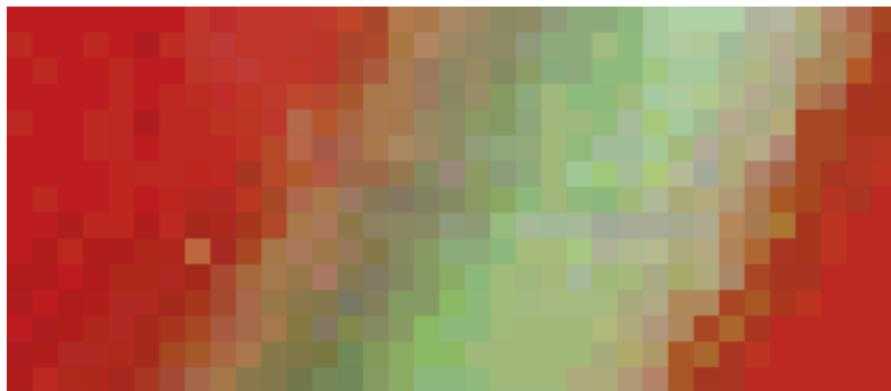


*médian/médian 5 × 5*

### 3.5. Filtrage médian d'images couleurs

#### ○ Filtrage marginal

- ▶ bruit de capteurs : pertinent de corriger chaque canal indépendamment
- ▶ mise en œuvre vectorielle possible avec la relation d'ordre entrelacée sur les couleurs



*médian/médian  $9 \times 9$*

# ANNEXE A

## Caractérisation statistique des autres bruits usuels

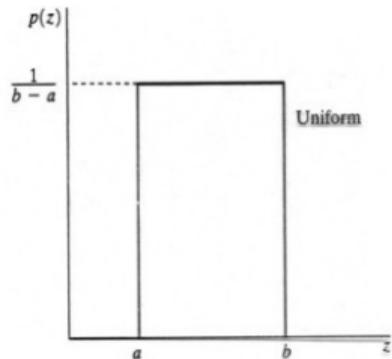
## ○ Bruit uniforme

- ▶ toutes les valeurs ont la même probabilité d'apparition

$$p_v(x) = \begin{cases} \frac{1}{b-a} & \text{si } a \leq x \leq b \\ 0 & \text{sinon.} \end{cases}$$

$$\mu = \frac{a+b}{2}$$

$$\sigma^2 = \frac{(a-b)^2}{12}$$



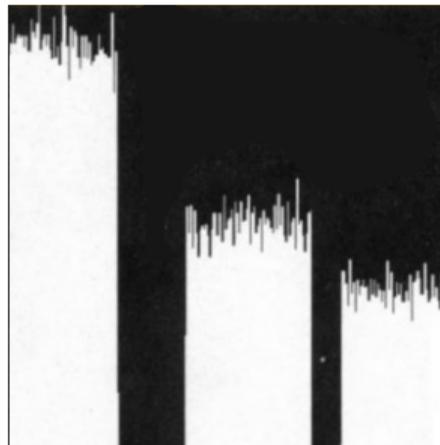
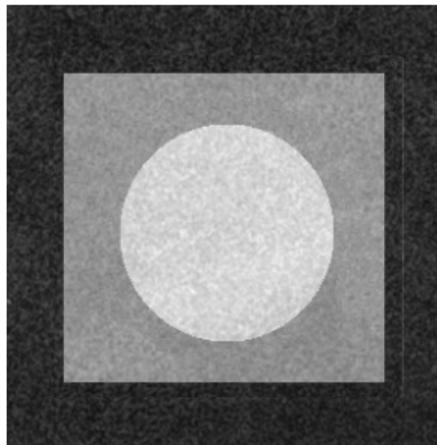
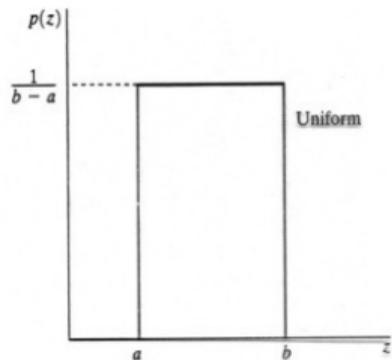
## ○ Bruit uniforme

- ▶ toutes les valeurs ont la même probabilité d'apparition

$$p_v(x) = \begin{cases} \frac{1}{b-a} & \text{si } a \leq x \leq b \\ 0 & \text{sinon.} \end{cases}$$

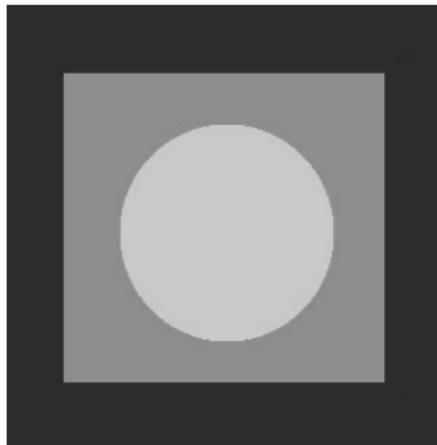
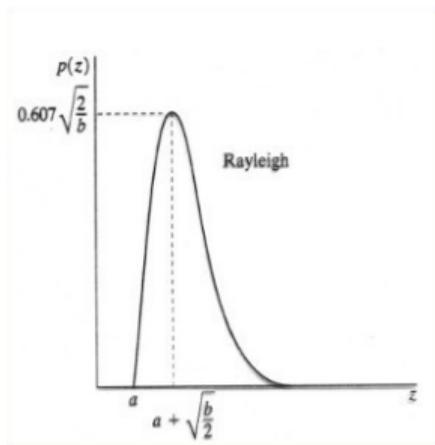
$$\mu = \frac{a+b}{2}$$

$$\sigma^2 = \frac{(a-b)^2}{12}$$



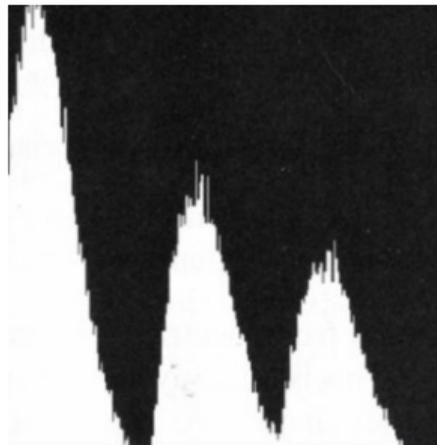
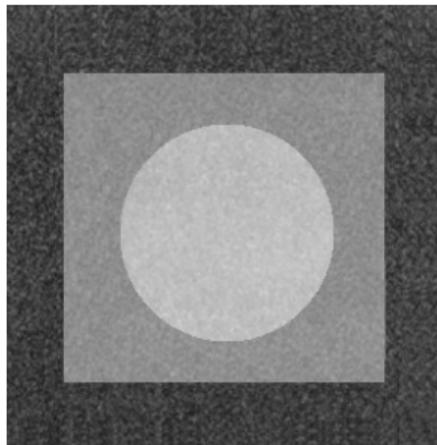
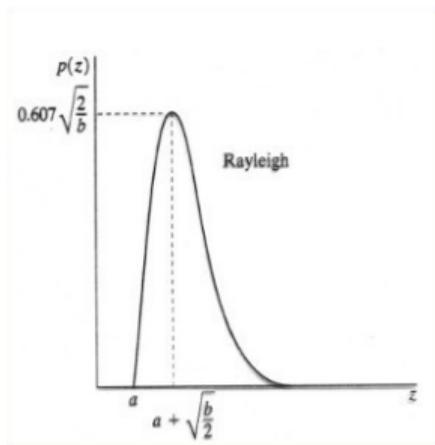
- Bruit de Rayleigh
  - ▶ imagerie ultrason

$$p_{\nu}(x) = \begin{cases} \frac{2}{b}(x-a)e^{-\frac{(x-a)^2}{b}} & \text{si } x \geq a \\ 0 & \text{sinon.} \end{cases}$$



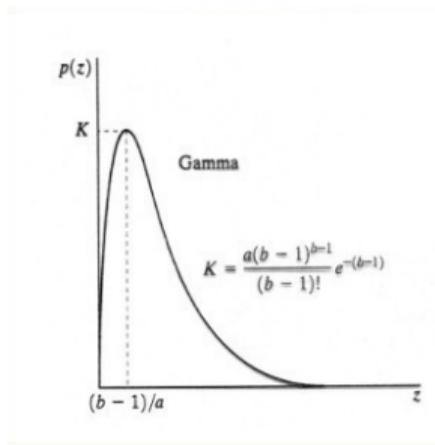
- Bruit de Rayleigh
  - ▶ imagerie ultrason

$$p_{\nu}(x) = \begin{cases} \frac{2}{b}(x-a)e^{-\frac{(x-a)^2}{b}} & \text{si } x \geq a \\ 0 & \text{sinon.} \end{cases}$$



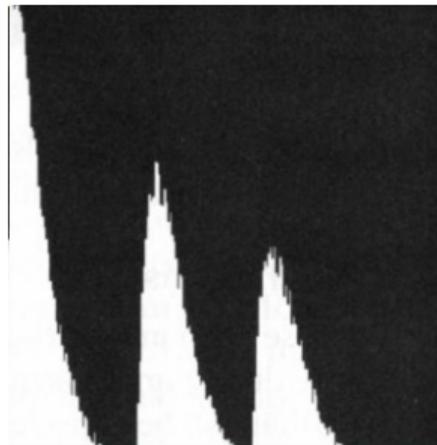
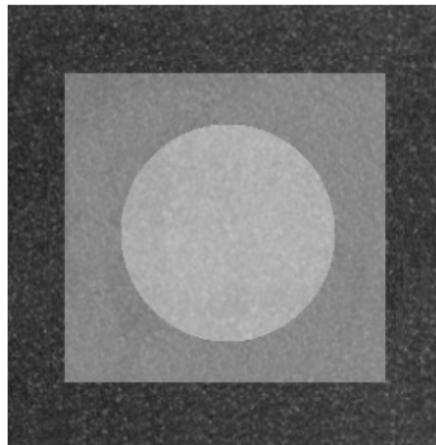
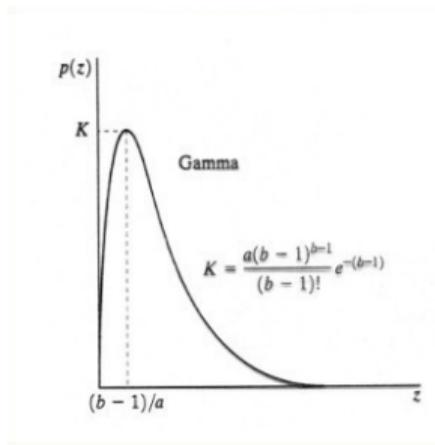
- Bruit d'Erlang ou gamma
  - ▶ imagerie laser

$$p_\nu(x) = \begin{cases} \frac{a^b x^{b-1}}{(b-1)!} e^{-ax} & \text{si } x \geq 0 \\ 0 & \text{sinon.} \end{cases}$$



- Bruit d'Erlang ou gamma
  - ▶ imagerie laser

$$p_\nu(x) = \begin{cases} \frac{a^b x^{b-1}}{(b-1)!} e^{-ax} & \text{si } x \geq 0 \\ 0 & \text{sinon.} \end{cases}$$

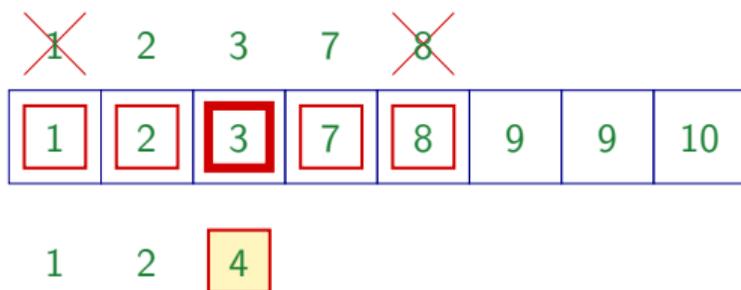


# ANNEXE B

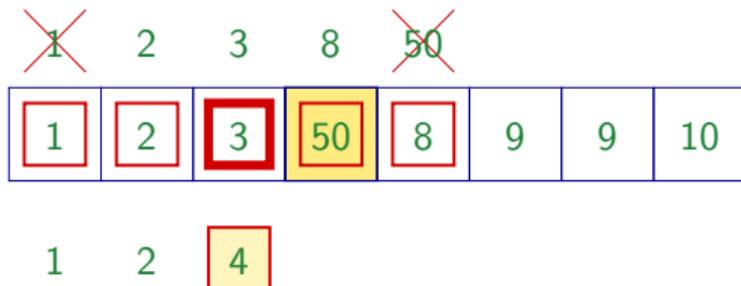
Exemple avec filtre olympique

○ Exemples en dimension 1 avec le filtre olympique

▶ sans bruit impulsionnel

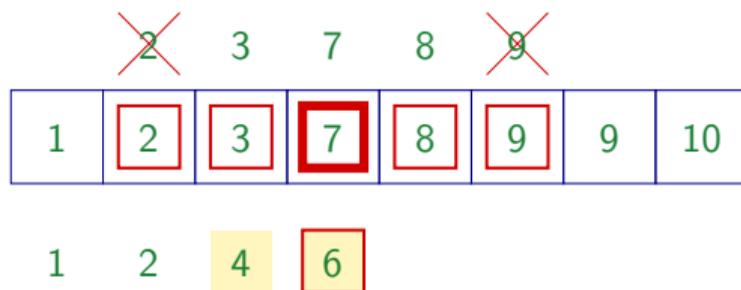


▶ avec bruit impulsionnel

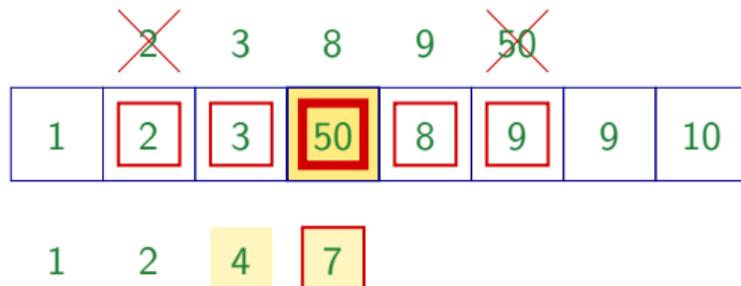


○ Exemples en dimension 1 avec le filtre olympique

▶ sans bruit impulsionnel

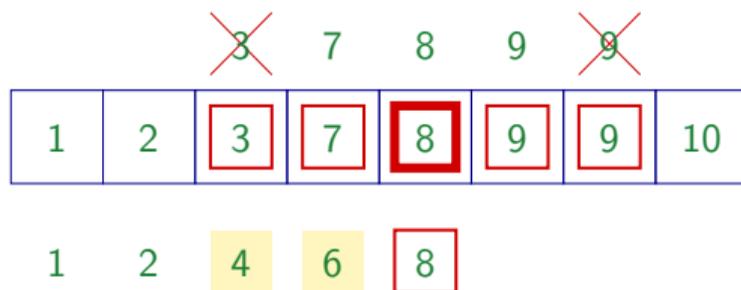


▶ avec bruit impulsionnel

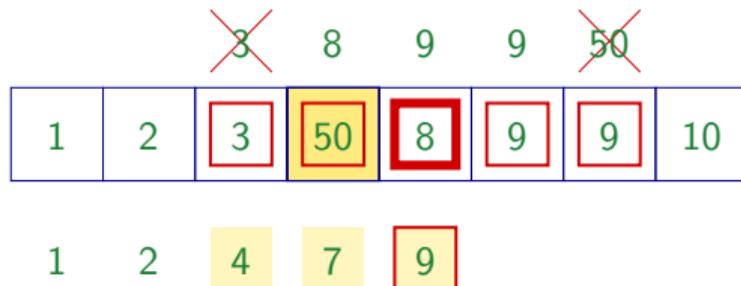


○ Exemples en dimension 1 avec le filtre olympique

▶ sans bruit impulsif

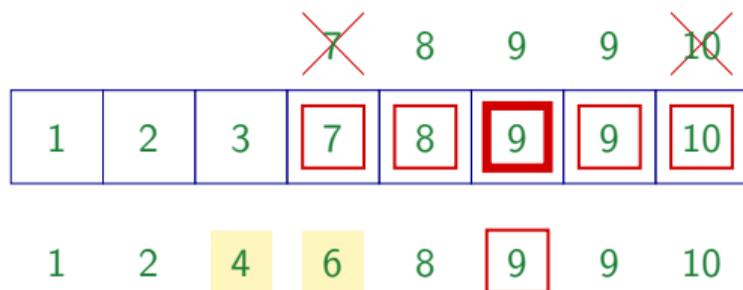


▶ avec bruit impulsif



○ Exemples en dimension 1 avec le filtre olympique

▶ sans bruit impulsifnel



▶ avec bruit impulsifnel

