



جامعة 8 ماي 1945 قالمة  
UNIVERSITE 8 MAI 1945 - GUELMA

# chapitre 2

## Les techniques de filtrage

2019-2020

# contenu

- Rappel:
  - Bruit
  - Filtrage
  - Convolution
- Les filtres:
  - Médian
  - Nagao
  - Min, max
  - Moyenneur
  - gaussien

# Bruit

- Le bruit est une altération de l'image : toute information pertinente dans l'image n'est pas simplement accessible.
- Les sources de bruits :
  - Bruits liés aux conditions de prise de vues : Le bougé , Problèmes liés à l'éclairage de la scène observée
  - Bruits liés au capteur : capteur mal réglé, capteur de mauvaise qualité (distorsion de la gamme des niveaux de gris ou en flou)
  - Bruits liés à l'échantillonnage (essentiellement des problèmes de quantification (CCD))
  - Bruits liés à la nature de la scène :
    - Exemples : nuage sur les images satellitaires, poussières dans les scènes industrielles, brouillard pour les scènes routières, etc.

# Filtrage

- Le filtrage est une opération qui consiste à réduire le bruit contenu dans une image. Il est considéré comme une transformation de l'image.
- Deux catégories de méthodes :
  - les méthodes du domaine *spatial* : ces méthodes se réfèrent à l'image elle-même, et sont basées sur la manipulation directe des pixels.
  - les méthodes du domaine *fréquentiel* : sont basées sur la modification de la transformée de Fourier de l'image.
- On parlera aussi de :
  - **filtres linéaires** exprimés sous forme de convolution
  - **filtres non linéaires**
- Dans une opération de filtrage en domaine spatial, le pixel est souvent considéré comme un individu (statistique) et on cherche son identité grâce à son voisinage.

## *Produit de convolution*

- Le produit de convolution est une opération très courante en traitement d'images.
- Ce produit représente une classe d'opérations simples à programmer et efficace dans ses résultats.
- Cette efficacité résulte directement de la linéarité de l'opération.

# Principe fondamental de la convolution

- Le produit de convolution est l'opération de voisinage **linéaire** et **invariante par translation**.
- Il n'y a pas d'autre opération ayant ces propriétés.
- Cette opération fait appel aux outils disponibles pour respecter la convolution:
  - multiplication du niveau de gris d'un point par une constante.
  - addition des niveaux de gris des points d'une image; dans le cas continu, cette addition des niveaux se traduit sous forme d'une intégrale de surface.

# Propriétés de la convolution

- Additivité/distributivité :  $(f * h_1) + (f * h_2) = f * (h_1 + h_2)$
- Commutativité :  $h_1 * h_2 = h_2 * h_1$
- Associativité du produit de convolution :  $(f * h_1) * h_2 = f * (h_1 * h_2)$

- Norme d'un opérateur  $\|h\| = \sum_{(p,q) \in \text{voisinage}} h(p,q)$

- Séparabilité d'un opérateur de convolution Un filtre de convolution est dit séparable si

$$h(x, y) = h_x(x) \cdot h_y(y)$$

$$f(x, y) * h(x, y) = h_y(y) * (h_x(x) * f(x, y))$$

Traitement selon x      Traitement selon y

# Convolution

$$g(x, y) = f(x, y) \otimes \text{filtre}(x, y)$$

$$g(x, y) = \sum_i \sum_j f(x+i, y+j) \cdot \text{filtre}(i, j)$$

$w_1$	$w_2$	$w_3$
$w_4$	$w_5$	$w_6$
$w_7$	$w_8$	$w_9$

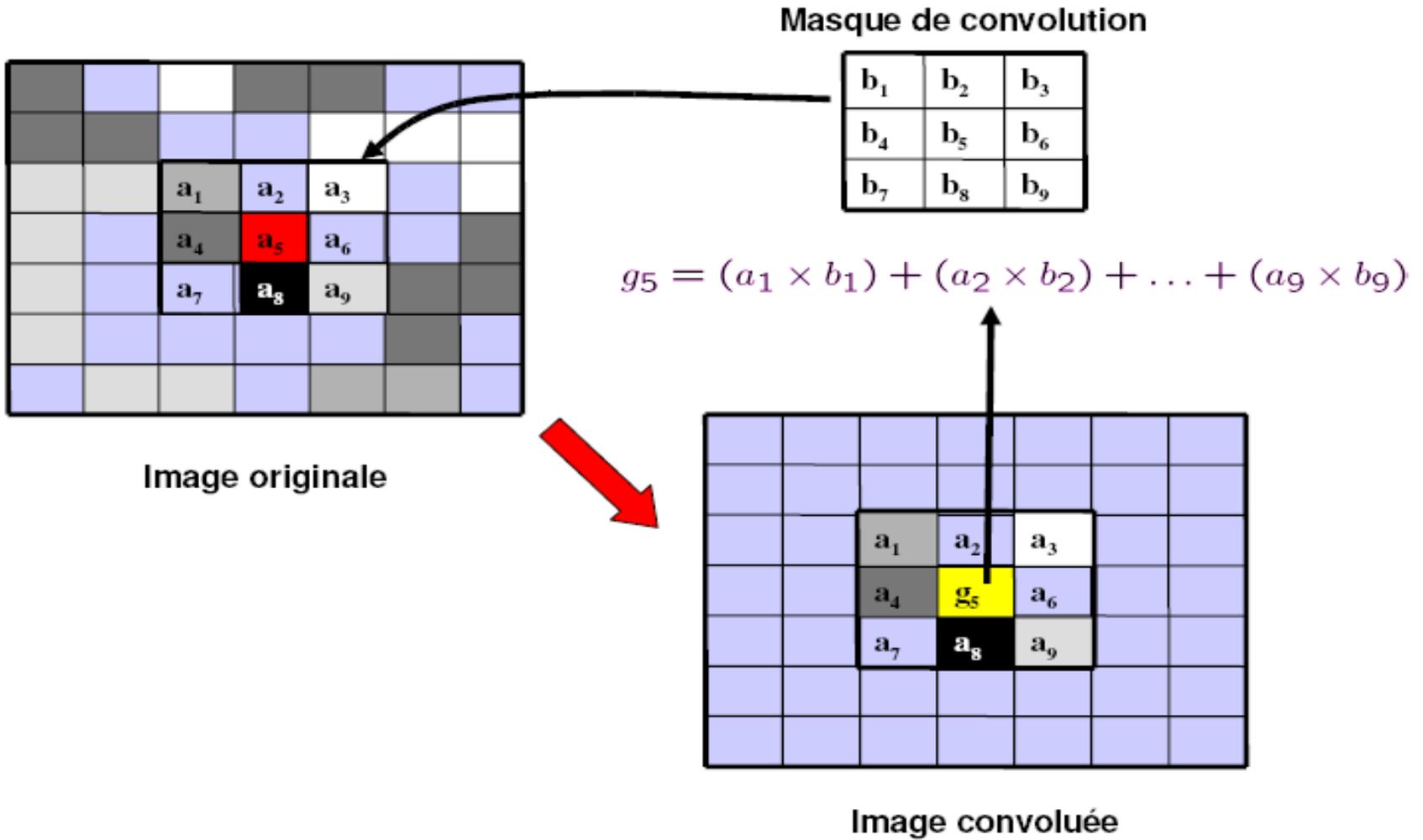
$$\text{filtre}(0, 0) = w_5$$

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

**Un pixel  $f(x, y)$  est remplacé par une somme pondérée de lui-même et des pixels de son voisinage**

**Généralement, le masque (filtre) est de dimension impaire et symétrique.**

# En Pratique



# En Pratique

100	100	100	100	100
100	100	100	100	100
100	100	150	100	100
100	100	100	100	100
100	100	100	100	100

\*

0	-1	0
-1	5	-1
0	-1	0

=

100	100	100	100	100
100	100	50	100	100
100	50	350	50	100
100	100	50	100	100
100	100	100	100	100

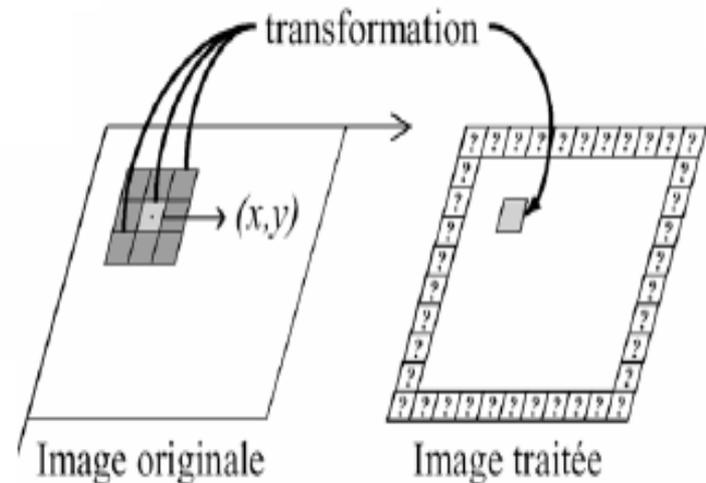
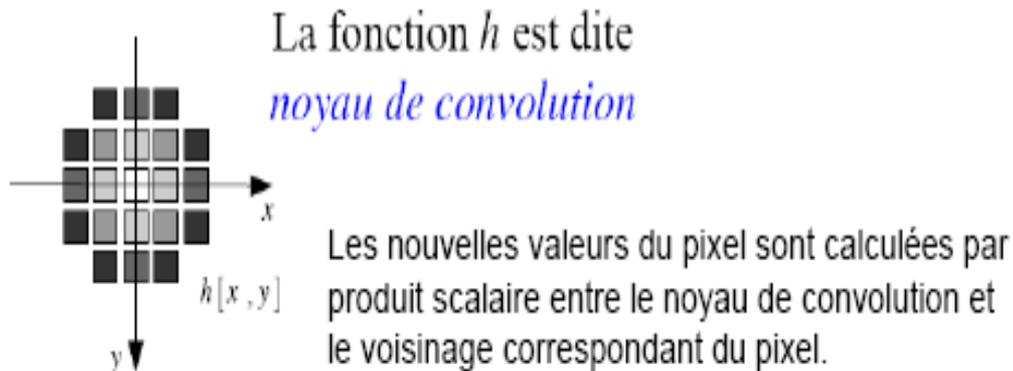
# Implantation des filtres linéaires

Soit  $I$  une image numérique.

Soit  $h$  une fonction de  $[x_1, x_2] \times [y_1, y_2]$  à valeurs réelles.

La *convolution* de  $I$  par  $h$  est définie par :

$$(I * h)[x, y] = \sum_{i=x_1}^{x_2} \sum_{j=y_1}^{y_2} h[i, j] \cdot I[x - i, y - j]$$



➤ Effet de bord ?

# Effet de bord

Ne pas filtrer le bords

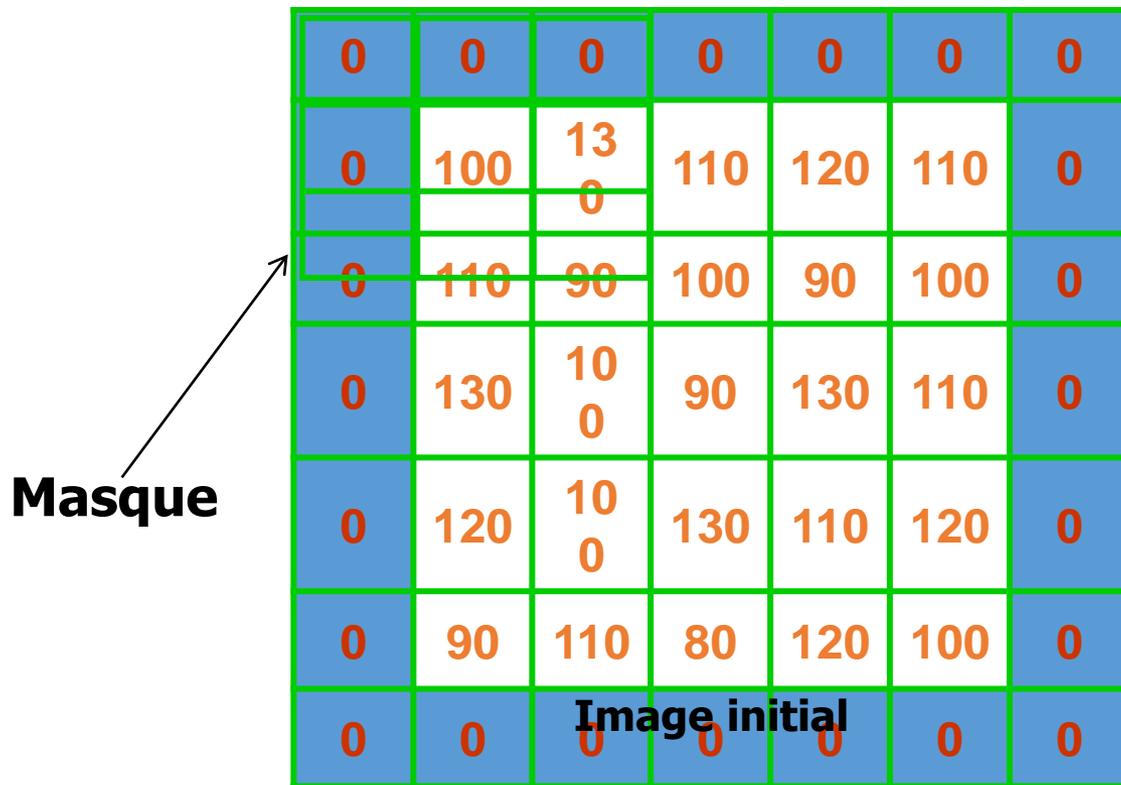
Masque

	100	130	110	120	110
	110	90	100	90	100
	130	100	90	130	110
	120	100	130	110	120
	90	110	80	120	100

Image initial

# Effet de bord

Mise à zéro de la couronne



# Effet de bord

Réaliser un effet de miroir

100	100	130	110	120	110	110
100	100	130	110	120	110	110
110	110	90	100	90	100	100
130	130	100	90	130	110	110
120	120	100	130	110	120	120
90	90	110	80	120	100	100
90	90	110	80	120	100	100

Image initial

## Bruit poivre et sel

- Un bruit poivre et sel d'ordre  $n$  est obtenu en ajoutant  $n/2$  pixels blancs et  $n/2$  pixels noirs aléatoirement dans une image.
- On le caractérise souvent par le pourcentage de pixels remplacés ( $n\%$ ).
- Ce bruit correspond à une réalité physique : Poussière sur objectif, petits objets, pertes de données

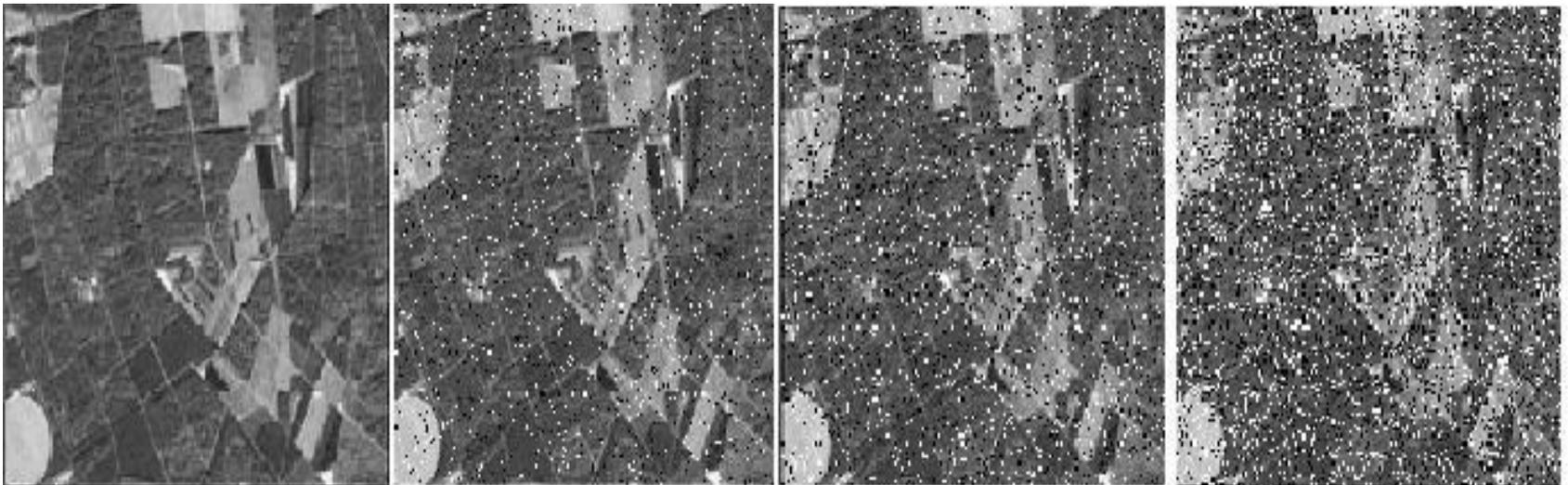


Image original

5%

10%

20%

# Bruit gaussien

- Est obtenu en ajoutant à chaque pixel une valeur aléatoire suivant une loi de probabilité Gaussienne :

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-\mu_1)^2 + (y-\mu_2)^2}{2\sigma^2}}$$

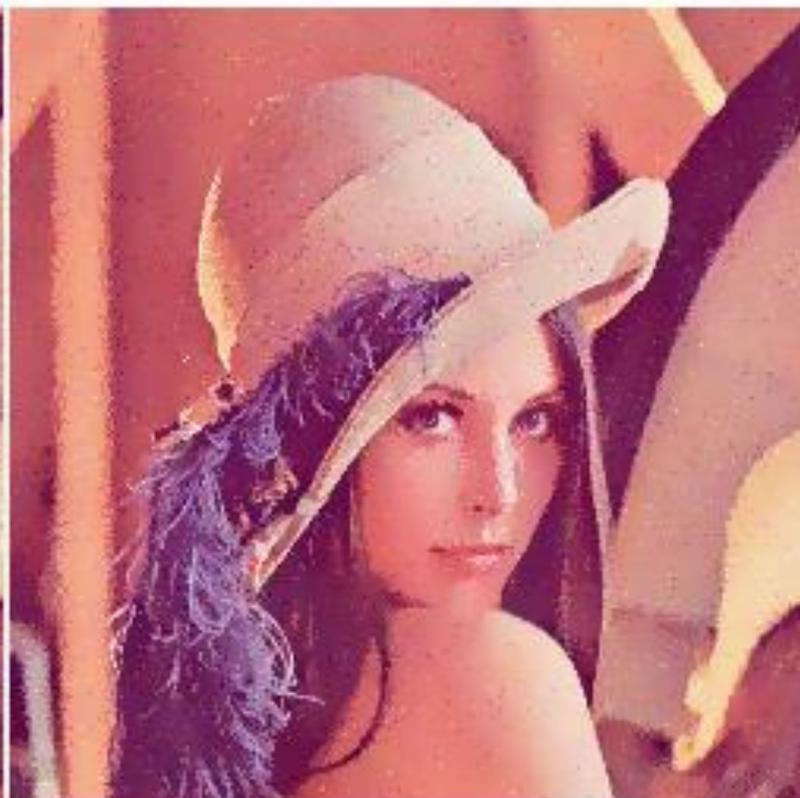
$\sigma$  : écart type

$\mu$  : moyenne

# Bruit gaussien



bruit gaussien  
sigma=20

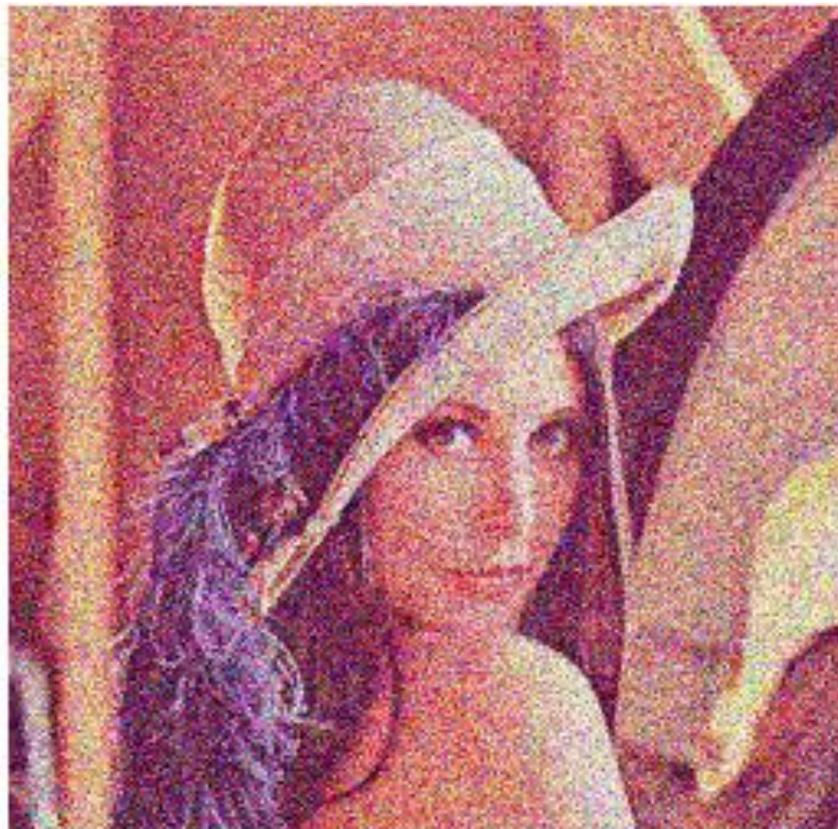


restauration

# Bruit gaussien



$\sigma=20$



$\sigma=40$

# Bruit gaussien

- Ce bruit modélise bien un grand nombre de bruits de capteurs visuels.
- Pour avoir une image **nette** il faut faire appel à des techniques permettant de « supprimer » le bruit (opération de lissage) et/ou de mettre en évidence et détecter les points frontières.

# Filtrage

-  **Filtrage passe-bas**  
Lissage et élimination du bruit
-  **Filtrage passe-haut**  
Détection des contours

# Filtrage passe-bas



## FILTRE NON LINÉAIRE

- N'est pas réalisable avec une convolution
- Médian, Max,...



## FILTRE LINÉAIRE

- Réalisable avec une convolution

$$g(x,y) = f(x,y) \otimes \text{filtre}(x, y)$$

Opérateur de convolution

- Moyenneur, Gaussien,...

# Exemple de Bruit

21	20	20	22	21
20	20	20	21	20
20	21	15	20	22
20	20	20	22	21
20	21	20	20	20



*	*	*	*	*
*	*	*	*	*
*	*	20	*	*
*	*	*	*	*
*	*	*	*	*

Bruit  
dans région homogène

5	4	20	22	21
4	5	20	21	20
4	4	15	20	22
3	5	20	22	21
4	4	20	20	20



.	.	*	*	*
.	.	*	*	*
.	.	15	*	*
.	.	*	*	*
.	.	*	*	*

Bruit  
dans zone de transition  
(contour)

5	4	20	22	21
4	5	20	21	20
4	4	20	20	22
3	5	20	22	21
4	4	20	20	20



.	.	*	*	*
.	.	*	*	*
.	.	20	*	*
.	.	*	*	*
.	.	*	*	*

Zone de transition  
non bruitée  
(respect des contours)

# FILTRE MEDIAN

- **Principe** :

Remplacer la valeur du pixel central par la valeur médiane de la répartition (luminances triées dans l'ordre croissant) des niveaux de gris des pixels situés à l'intérieur de cette fenêtre.

- **But**: Préserve les contours.

➤ Utile pour contrer l'effet d'un bruit Poivre & Sel

# FILTRE MEDIAN

$$g(x, y) = \text{médian} \left\{ f(n, m) \right\}_{(n, m) \in S}$$

		19	23	42	
		11	150	31	
		60	25	12	

médiane



11	12	19	23	25	31	42	60	150
----	----	----	----	----	----	----	----	-----



bruit

		19	23	42	
		11	25	31	
		60	25	12	

# FILTRE MEDIAN

123	143	122	167	90	171	200
70	72	75	78	254	212	199
122	134	90	93	123	128	112
156	157	167	168	198	190	80
123	110	129	135	145	126	85
130	123	125	124	156	176	90
105	101	90	80	78	67	57

Tri des élément : 90 93 123 167 168 198 129 135 145  
90 93 123 129 135 145 167 168 198

La valeur 168 ça sera remplacé par 135

# FILTRE MEDIAN

## **Intérêt du filtre médian:**

- Un pixel non représentatif dans le voisinage affectera peu la valeur médiane.
- La valeur médiane choisie étant le niveau de gris d'un des pixels considérés, on ne crée pas alors de nouveaux niveaux de gris dans l'image. Ainsi lorsque le filtre passe sur un contour très marqué il le préservera mieux.

# Exemple



*Image Originale*



*Filtrage médian  $5 \times 5$*

# Exemple



*Image Originale*



*Filtrage médian  $5 \times 5$*

# Exemple



**image originale**



**image filtrée par un filtre median de  
taille 3\*3**



**image filtrée par un filtre median de  
taille 5\*5**

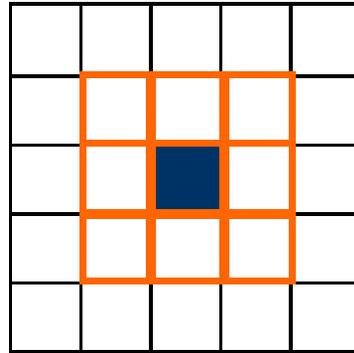


**image filtrée par un filtre median de  
taille 7\*7**

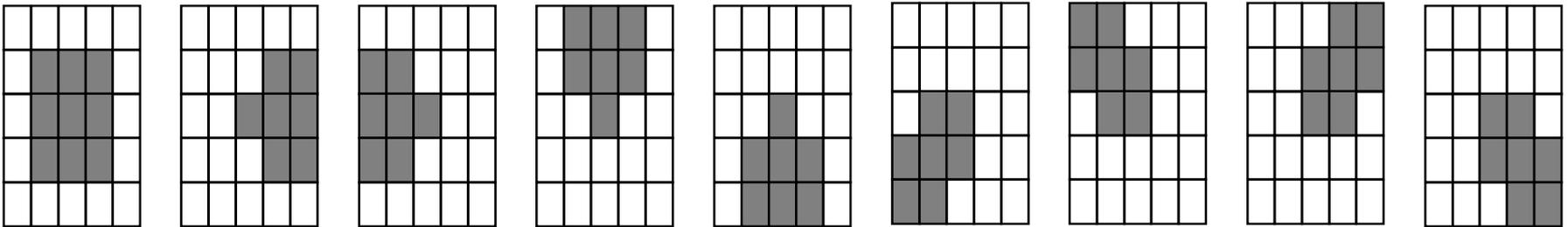
# Filtre Nagao

- **Principe:** Découpe d'une fenêtre 5x5 centrée sur le pixel en 9 fenêtres de 7 pixels. Mesure sur chacune de ces fenêtres d'une valeur de l'homogénéité (variance par exemple).
- Le pixel central est alors remplacé par la valeur de la zone la plus homogène.

# Filtre Nagao



fenêtre 5x5



9 fenêtres de 7 pixels

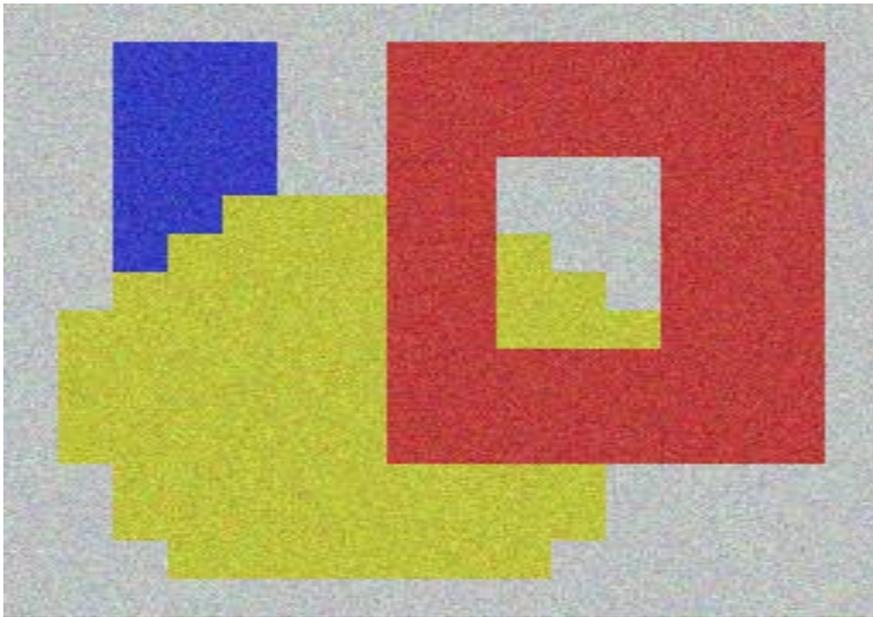
# Filtre Nagao

- Les contours sont ainsi bien conservés car le lissage ne se fait que dans sa direction tangentielle c'est-à-dire dans la direction où la modification est la moins visible.
- C'est la raison pour laquelle on dit que le filtre de Nagao est qualifié de lissage avec conservation et même accentuation des contours.
- Les contours sont remarquablement conservés par ce filtre
- on peut d'ailleurs montrer que le Nagao renforce donc le contraste.
- Les détails fins sont nettement plus apparents qu'avec le médian

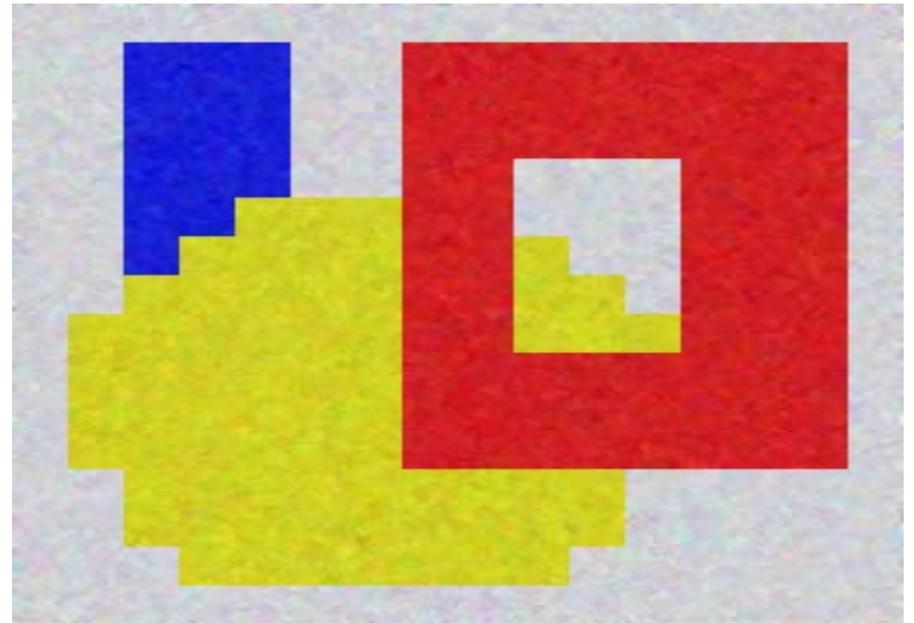
# Filtre Nagao

- Ce filtrage peut être réitéré plusieurs fois sur la matrice déjà traitée.
- On observe que ce filtre est pratiquement idempotent: au bout de quelques itérations, l'image (ou la matrice) ne se modifie presque pas.
- ✓ On arrête alors le processus de traitement.

# Exemple



Bruit gaussien  $\sigma=30$



Filtre de Nagao

# Exemple



*Image Originale*



*Filtrage Nagao*

# Exemple



*Image Originale*



*Filtrage Nagao*

## Filtrage Min Max

- On considère le niveau de gris du pixel à traiter, et d'autre part tous ses voisins (à l'exception de lui même).
- Sur les voisins on calcule le niveau min et le niveau max.,
- si le niveau de gris du pixel à traiter est compris entre le min et le max. alors on le laisse inchangé
- sinon on le remplace par le max(resp min).

# Filtrage Min Max

- **Principe**: Consiste à remplacer la valeur au pixel  $(x,y)$  par le minimum ou le maximum sur la fenêtre centrée en ce point selon la proximité de ces deux valeurs à la valeur centrale.

# Filtrage Min Max

123	143	122	167	90	171	200
70	72	75	78	254	212	199
122	134	90	93	123	128	112
156	157	167	208	198	190	80
123	110	129	135	145	126	85
130	123	125	124	156	176	90
105	101	90	80	78	67	57

Min=90, Max=198 on a donc 208 >Max d'où la valeur 208 ça sera remplacé par 198 dans le cas du filtrage Max.

# Filtrage Min Max



Image originale

# Filtrage Min Max



Filtrage Max

# Filtrage Min Max

- Ce filtre de lissage supprime bien le bruit de type “poivre et sel” c’est à dire qu’il “adoucit” les pixels isolés ayant un niveau de gris très différent des niveaux de gris de leur voisinage
- Il a la particularité de bien préserver les contours très marqués.
- Ce filtre s’assure en fait que tout pixel a son niveau de gris placé dans la gamme de ses voisins.

# Filtrage par la moyenne

- Le filtrage par la moyenne consiste à remplacer chaque pixel par la valeur moyenne de ses voisins (le pixel lui-même y compris).

```
120 120 0
120 255 120
120 120 120
```

Nous allons remplacer  $p_{3,3} = 255$  par :

$$p_{3,3}^{moy} = \frac{120+120+0+120+255+120+120+120+120}{9} \approx 121,67 \approx 122$$

# Filtrage par la moyenne

- Cette méthode a pour effet de modifier les niveaux de gris trop différents de leurs voisins en ce sens on peut penser « supprimer » le bruit, c'est à dire des niveaux de gris « anormaux ».

# Filtrage par la moyenne

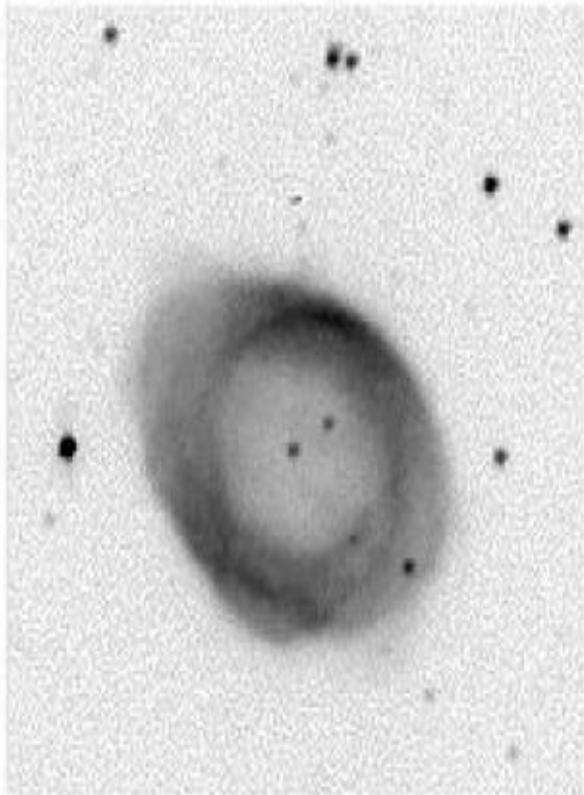
- Suivant la “violence” du lissage que l’on veut réaliser on choisira une taille de filtre plus ou moins grande (3x3, 5x5,..) mais on doit comprendre que les contours de l’image de départ deviendront alors plus « flous ».

$$W_1 = 1/9 \begin{array}{ccc} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{array}$$

$$W'_1 = 1/25 \begin{array}{ccccc} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{array}$$

$$W_2 = 1/10 \begin{array}{ccc} & 1 & 1 & 1 \\ 1 & 2 & 1 & \\ 1 & 1 & 1 & \end{array}$$

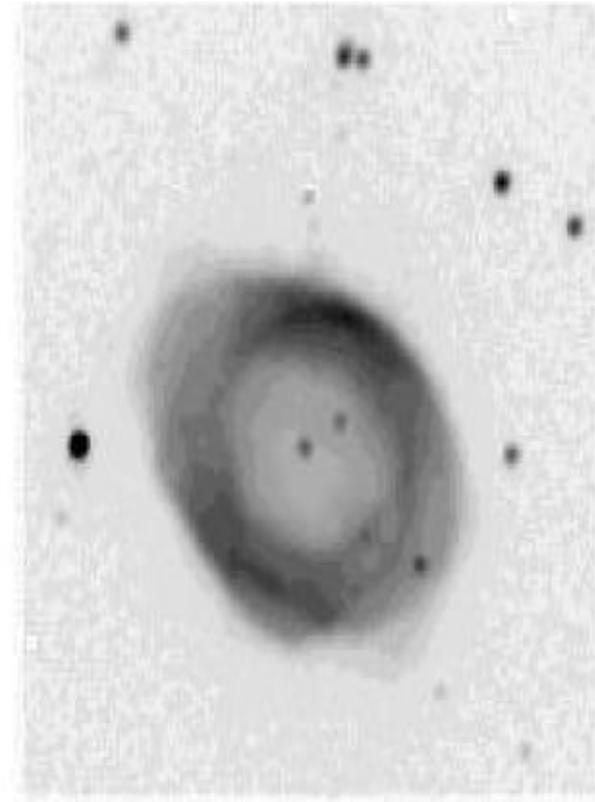
# Filtrage par la moyenne



\*

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

=



# Filtrage par la moyenne



**image originale**



**image filtrée par un filtre moyeneur  
de taille 3\*3**



**image filtrée par un filtre moyeneur  
de taille 5\*5**



**image filtrée par un filtre moyeneur  
de taille 7\*7**

# Filtrage par la moyenne

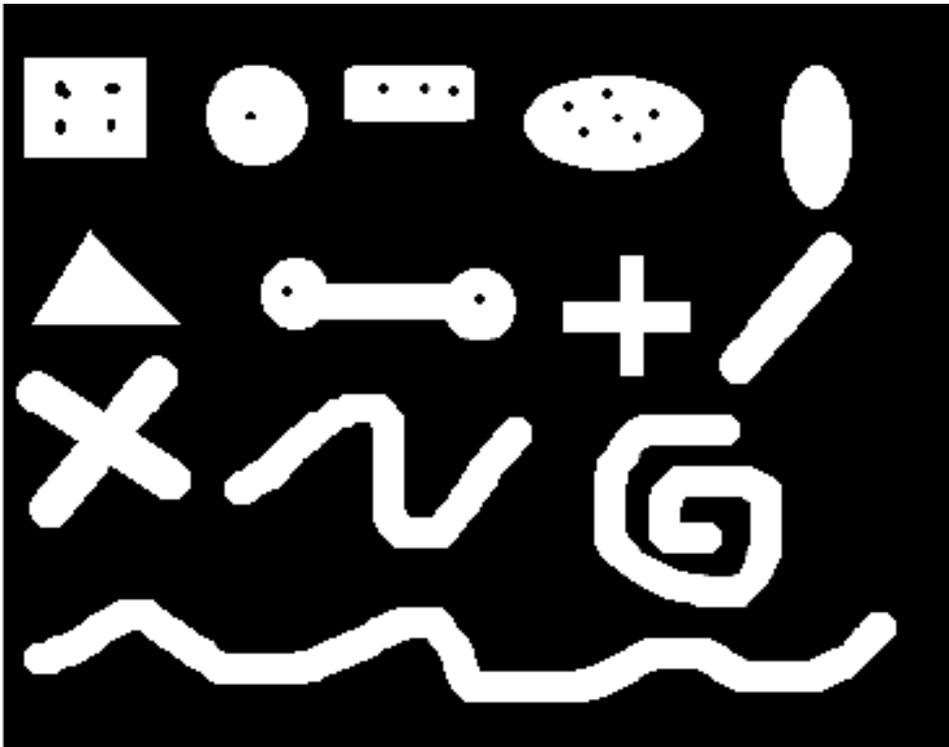
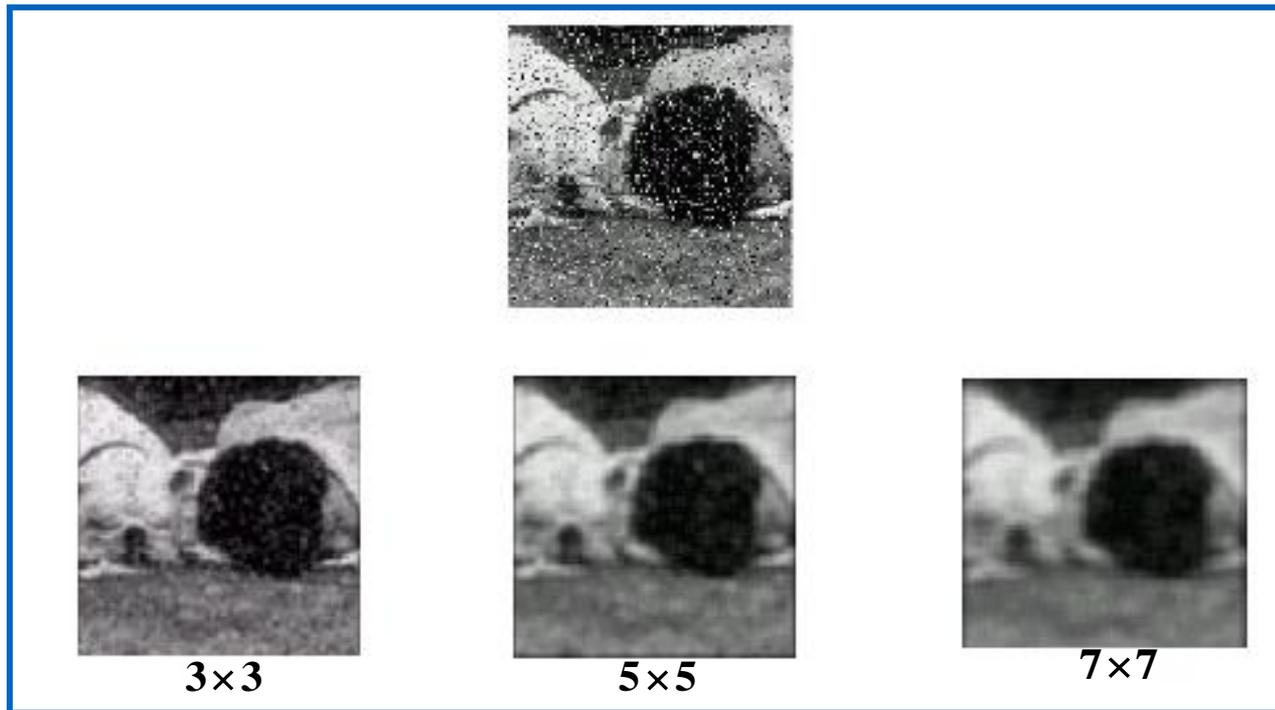


Image Originale



filtre Moyenne 3x3

# Filtrage par la moyenne



# Filtrage par la moyenne

- Un pixel isolé avec un niveau de gris “anormal” pour son voisinage va perturber les valeurs moyennes des pixels de son voisinage.
- Sur une frontière de régions le filtre va estomper le contour et le rendre flou, ce qui est gênant en visualisation bien sûr mais éventuellement aussi pour un traitement ultérieur qui nécessiterait des frontières nettes.

# Filtrage par la moyenne

- Il est possible de moduler ces effets néfastes en réalisant en chaque pixel une convolution “conditionnelle”
- Par exemple en un pixel de niveau de gris NG1 on applique le filtre et on obtient une valeur NG2, alors on décidera d’appliquer le filtre que si :

$$|NG1-NG2| \geq SEUIL$$

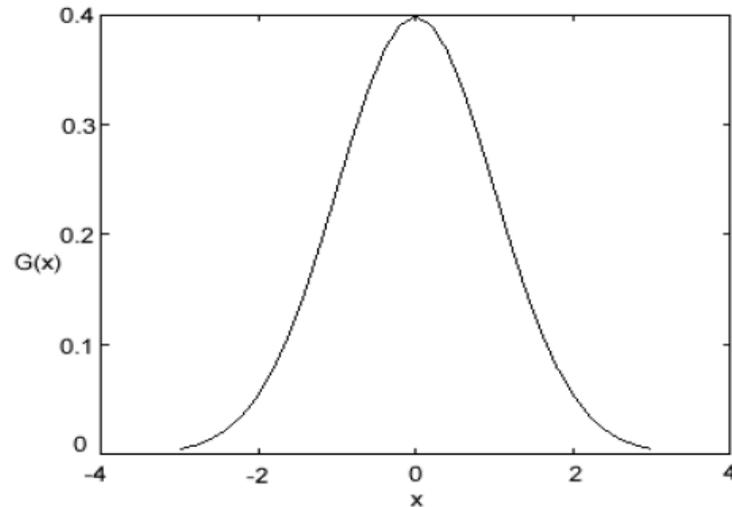
# LE FILTRE GAUSSIEN

- **L**e filtre gaussien est un opérateur de lissage utilisé pour estomper les «détails» et le bruit. Ce filtre a une logique analogue au filtre moyenne.

# LE FILTRE GAUSSIEN

- La fonction Gaussienne est aussi souvent utilisée dans les distributions statistiques, elle est définie par la fonction

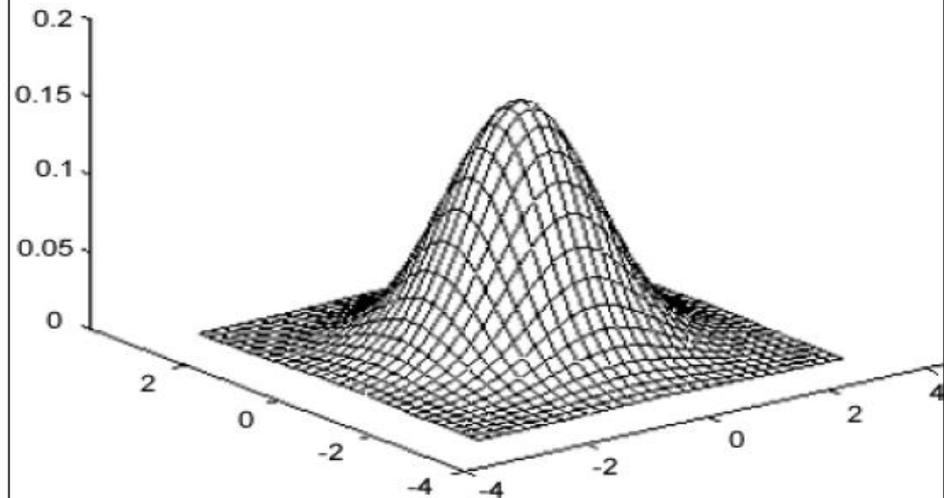
$$G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}$$



# LE FILTRE GAUSSIEN

- Dans le traitement d'images on traite des données à deux dimensions ( X et Y ), on introduit donc une fonction gaussienne à deux dimensions  $G(x, y)$  :

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



# LE FILTRE GAUSSIEN

- Le filtrage Gaussien va utiliser cette distribution pour définir un filtre de convolution.
- Comme on travaille sur des images discrètes on utilise une approximation discrète de la distribution gaussienne dans un filtre fini de convolution.

1 / 115\*

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

Fenêtre de taille  $5 \times 5$  d'écart  
type égal à 1.4

# LE FILTRE GAUSSIEN

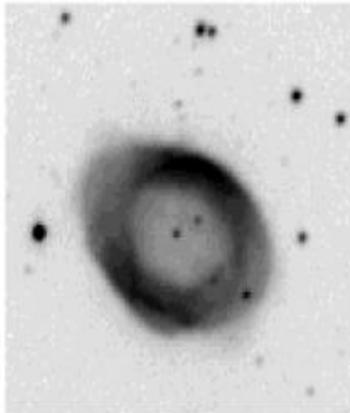
$$\frac{1}{16}$$

1	2	1
2	4	2
1	2	1

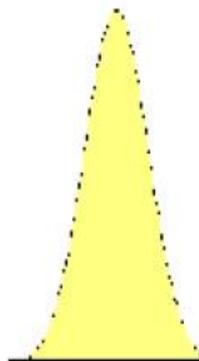
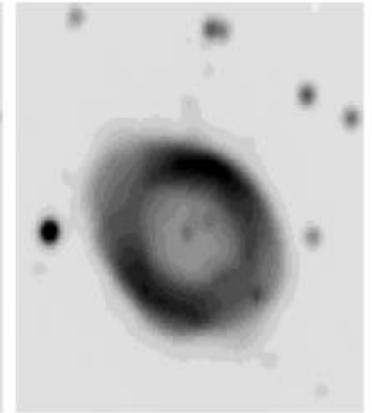
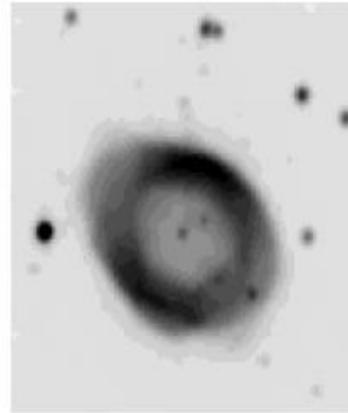
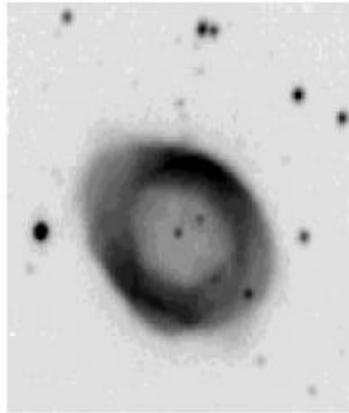
$$\frac{1}{246}$$

1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1

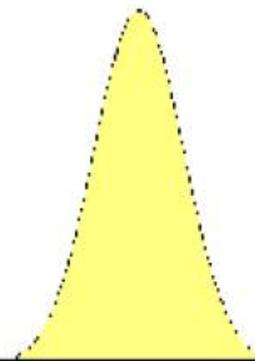
# LE FILTRE GAUSSIEN



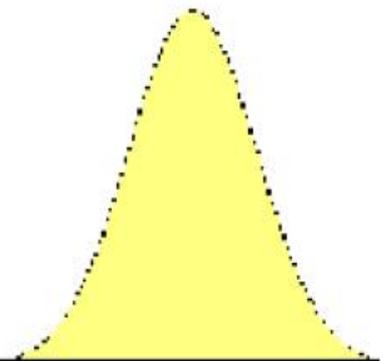
Gaussian  
\* Kernel =



$\sigma=0.5$

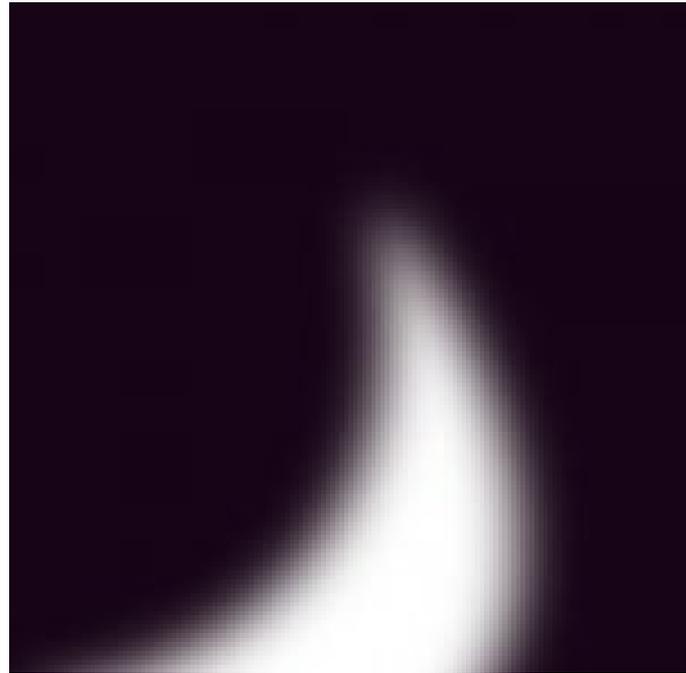
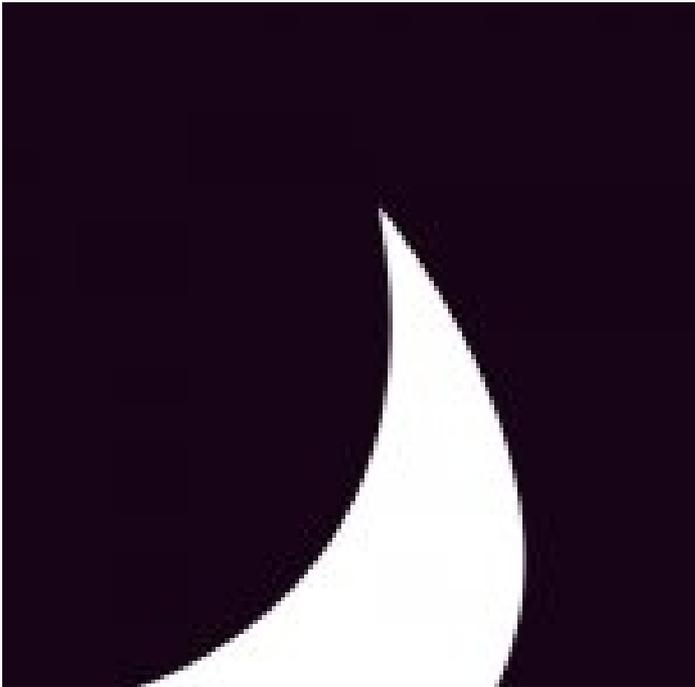


$\sigma=1.0$



$\sigma=1.5$

# LE FILTRE GAUSSIEN



# LE FILTRE GAUSSIEN



Image originale



Flou gaussien

# LE FILTRE GAUSSIEN

- Le degré de lissage est déterminé par l'écart-type associé au filtre gaussien mais alors plus l'écart-type est élevé plus il faut prendre un filtre grand (5x5, 7x7).

# LE FILTRE GAUSSIEN

- Accorde plus d'importance au pixel central
- Filtre séparable:
  - Économie de calcul (filtrage des colonnes et des lignes séparément)
  - Exemple: filtre 5x5
    - Normalement: 5x5 multiplications + 24 additions par pixel.
    - Séparé: 5x2 multiplications + 4x2 additions par pixel.
- Atténuation du bruit

# Filtrage : exercices

- Que font les filtres à noyau de convolution suivants (prenez un exemple numérique si nécessaire)

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\frac{1}{16} \begin{bmatrix} -1 & -2 & -1 \\ -2 & 12 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

- Quelle est la condition sur les coefficients pour que le filtrage soit passe-bas

- Décomposer le filtre 2D de noyau  $\frac{1}{b^2 + 4a(a+b)} \begin{bmatrix} a^2 & ab & a^2 \\ ab & b^2 & ab \\ a^2 & ab & a^2 \end{bmatrix}$

sous forme du produit de convolution de 2 filtres 1D

- En déduire un moyen efficace, en nombre d'opérations par pixel, d'implémenter les filtres précédents