Filtrage linéaire

- Filtrage spatial
- Filtrage fréquentiel
- Filtrage linéaire adaptatif

Rappels et définition

- Un filtre numérique linéaire 1D se représente par une équation du type:

$$y(n) = \sum_{i=-\infty}^{+\infty} h_i x(n-i)$$

  - Si i : 0 à $+\infty$ alors le signal est causal : h_i=0 pour i<0, $y(n) = \sum_{i=0}^{+\infty} h_i x(n-i)$, sinon on dit qu'il est anti-causal
  - Si il existe une limite tq h_i=0 pour i>M alors le filtre est dit à réponse impulsionnelle finie RIF, $y(n) = \sum_{i=0}^{+M} h_i x(n-i)$ sinon il est à réponse impulsionnelle infinie RII
  - Il existe donc des filtres RIF causaux et anti-causaux, des filtres RII causaux et anti-causaux
  - L'étude de la stabilité s'effectue en étudiant les pôles et les zéros de la TZ
  - Le comportement en fréquence s'étudie à partir de la TFTD ou TZ pour $z=e^{2i\pi\lambda}$ de l'équation de récurrence.
  - La TFD est une discrétisation de la TFTD

(Revoir cours de TS numérique)

Résumé

	Causal	Anti-causal
RIF	$y(n) = \sum_{k=0}^{+M} h_k x(n-k)$	$y(n) = \sum_{k=-N}^{+N} h_k x(n-k)$
RII	$y(n) = \sum_{k=0}^{+\infty} h_k x(n-k)$	$y(n) = \sum_{k=-\infty}^{+\infty} h_k x(n-k)$

Généralisation du 1D au cas 2D

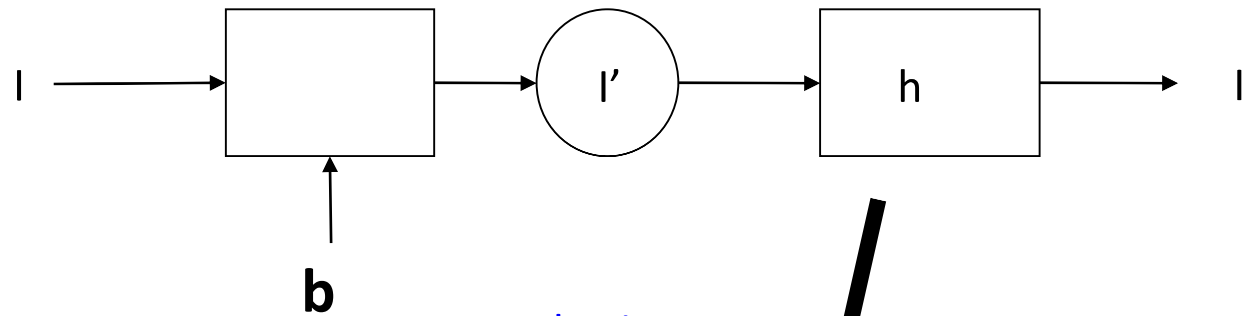
	Causal
RIF	$y(i,j) = \sum_{k=0}^{+M} \sum_{l=0}^{+N} h_{kl} x(i-k, j-l)$
RII	$y(i,j) = \sum_{k=0}^{+M} \sum_{l=0}^{+N} h_{kl} x(i-k, j-l)$

Le filtrage spatial s'intéresse à la détermination des coefficients h_{kl}

Filtrage spatial

- Filtres passe-bas

Objectif habituel de ces filtres : réduire le bruit haute fréquence.



[Le bruit doit être modélisé]

Le bruit peut être :

-Additif -> **$I'=I+b$**

-Multiplicatif -> $I'=I.b$

-Convolutif -> $I'=I*b$

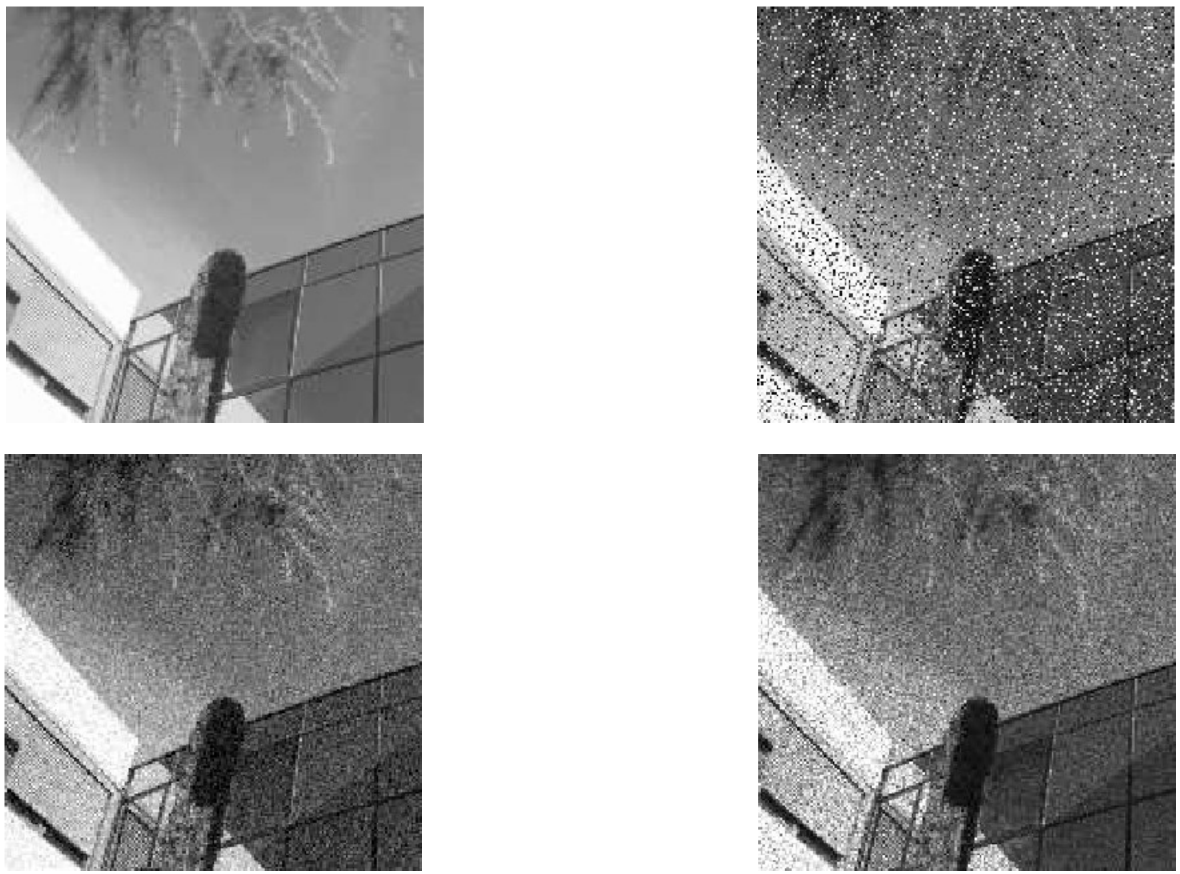
<u>Hyp</u> : Une Intensité constante doit rester constante après filtrage

[***Plusieurs solutions h***]

Modèle de bruit

<u>Caractéristiques du bruit</u> ?

- bruit indépendant ou non de l'image ?
- bruit additif ou multiplicatif ?
- bruit spatialement variable ?



Modèle de bruit

Cas du bruit additif gaussien

<u>Définition</u> : un bruit gaussien est obtenu en ajoutant à chaque pixel une valeur aléatoire selon une loi gaussienne de moyenne μ et de variance σ^2

$$G_{\mu,\sigma}(p) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(p-\mu)^2}{2\sigma^2}}$$

Le bruit blanc gaussien centré modélise un grand nombre de bruits associés aux capteurs visuels

sans bruit

μ=0, σ=40

μ=0, σ=60

Modèle de bruit

Cas du bruit poivre et sel

<u>Définition</u> : un bruit poivre et sel d'ordre *n* consiste à ajouter *n* pixels blancs et *n* pixels noirs de manière aléatoire dans une image (paramètre : % de pixels remplacés). En général, peu de pixels sont affectés mais grandement.
Ex : poussière sur objectif, objets de la taille du pixel, perte de données lors d'une transmission…


sans bruit


5%


15%

Les solutions pour h

Soit h(k,l) la réponse impulsionnelle définit sur une fenêtre [-M;+M]*[-N;+N], les filtres suivants satisfont aux critères définis précédemment :

$$\sum \sum h(k,l) = 1$$

Respect intensité constante

- filtre moyenne $h(k,l) = \frac{1}{L}$ avec $L = 2M * 2N$

- filtre gaussien $h(k,l) = C.\exp(-\frac{k^2 + l^2}{2\sigma^2})$

- filtre Shen-Castan $h(k,l) = K.\exp(-\frac{|k| + |l|}{a})$

avec C et K qui sont des constantes de normalisation rendant la somme des coefficients égal à 1, cette constante est à recalculer en fonction de la taille choisie du filtre

Le filtre étant déterminé l'image filtrée sera obtenue par une simple convolution avec ce noyau suivant l'équation générale de convolution 2D.

Soit I l'image, I(i,j)=l la luminance au pixel (i , j)

$$I_f(i,j) = I * h = \sum_{k=-M}^{M} \sum_{l=-N}^{N} h(k,l) I(i-k, j-l)$$

Exemple de calcul : filtre gaussien

$$h(k,l) = C.\exp(-\frac{k^2+l^2}{2\sigma^2})$$

Cas 3*3 : M=1, N=1

k=-1,0,1 et l=-1,0,1 et σ=0.8

$$C.\begin{pmatrix} \exp(-\frac{2}{2\sigma^2}) & \exp(-\frac{1}{2\sigma^2}) & \exp(-\frac{2}{2\sigma^2}) \\ \exp(-\frac{1}{2\sigma^2}) & 1 & \exp(-\frac{1}{2\sigma^2}) \\ \exp(-\frac{2}{2\sigma^2}) & \exp(-\frac{1}{2\sigma^2}) & \exp(-\frac{2}{2\sigma^2}) \end{pmatrix}$$

$$\begin{pmatrix} 0{,}21 & 0{,}458 & 0{,}21 \\ 0{,}458 & 1 & 0{,}458 \\ 0{,}21 & 0{,}458 & 0{,}21 \end{pmatrix}$$

$$C = \frac{1}{\sum_{k=-1}^{k=1}\sum_{l=-1}^{l=1}\exp(-\frac{k^2+l^2}{2\sigma^2})} = 0{,}272$$

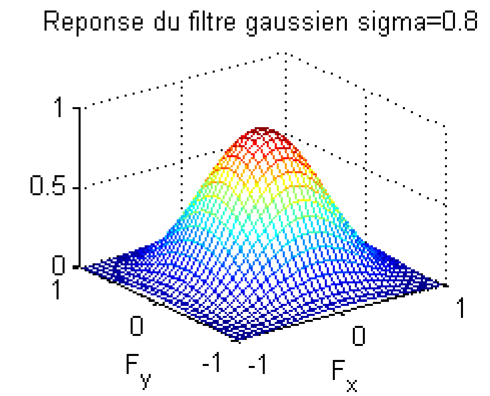
$$C \approx \frac{1}{2\pi\sigma^2} = 0{,}2486..$$

Rmqs : très souvent la largeur du filtre est fonction du sigma (pour plus de précisions) W=5*sigma (ex W=3 pour i=-1:1, 5*0.8=4 donc W=3 ou W=5 car doit être impaire)

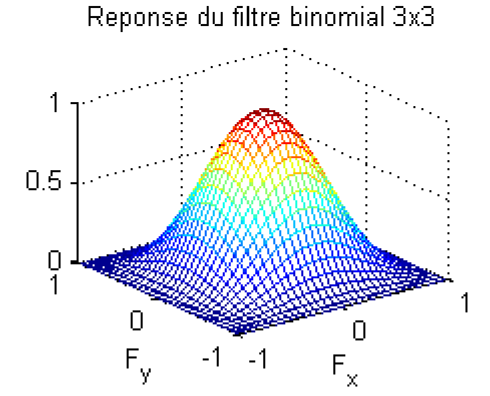
$$\begin{pmatrix} 0{,}057 & 0{,}125 & 0{,}057 \\ 0{,}125 & 0{,}458 & 0{,}125 \\ 0{,}057 & 0{,}125 & 0{,}057 \end{pmatrix}$$

Comparaison gaussien / binômial

$$h_{gaussien\_\sigma=0.8} = \begin{pmatrix} 0.052 & 0.114 & 0.052 \\ 0.114 & 0.249 & 0.114 \\ 0.052 & 0.114 & 0.052 \end{pmatrix}$$



$$h_{binomial} = \frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} = \begin{pmatrix} 0.0625 & 0.125 & 0.0625 \\ 0.125 & 0.25 & 0.125 \\ 0.0625 & 0.125 & 0.0625 \end{pmatrix}$$



- Implémentation pratique : masque de convolution

Exemple pour une fenêtre d'analyse (taille du masque de convolution : 3*3)

h(1,1)	h(1,0)	h(1,-1)
h(0,1)	h(0,0)	h(0,-1)
h(-1,1)	h(-1,0)	h(-1,-1)

I(i-2,j-2)	I(i-2,j-1)	I(i-2,j)	I(i-2,j+1)	I(i-2,j+2)
I(i-1,j-2)	I(i-1,j-1)	I(i-1,j)	I(i-1,j+1)	I(i-1,j+2)
I(i,j-2)	I(i,j-1)	I(i,j)	I(i,j+1)	I(i,j+2)
I(i+1,j-2)	I(i+1,j-1)	I(i+1,j)	I(i+1,j+1)	I(i+1,j+2)
I(i+2,j-2)	I(i+2,j-1)	I(i+2,j)	I(i+2,j+1)	I(i+2,j+2)

$$I_f(m,n) = I * h = \sum_{p=-M}^{M} \sum_{q=-N}^{N} h(p,q).I(m-p,n-q)$$

$$I_f(i,j) = h(1,1)I(i-1,j-1) + h(1,0)I(i-1,j) + h(1,-1)I(i-1,j+1)$$
$$+ h(0,1)I(i,j-1) + h(0,0)I(i,j) + h(-1,-1)I(i+1,j+1)$$
$$+ h(-1,1)I(i+1,j-1) + h(-1,0)I(i+1,j) + h(-1,-1)I(i+1,j+1)$$

Rmq : équivalent à la corrélation quand le masque est symétrique

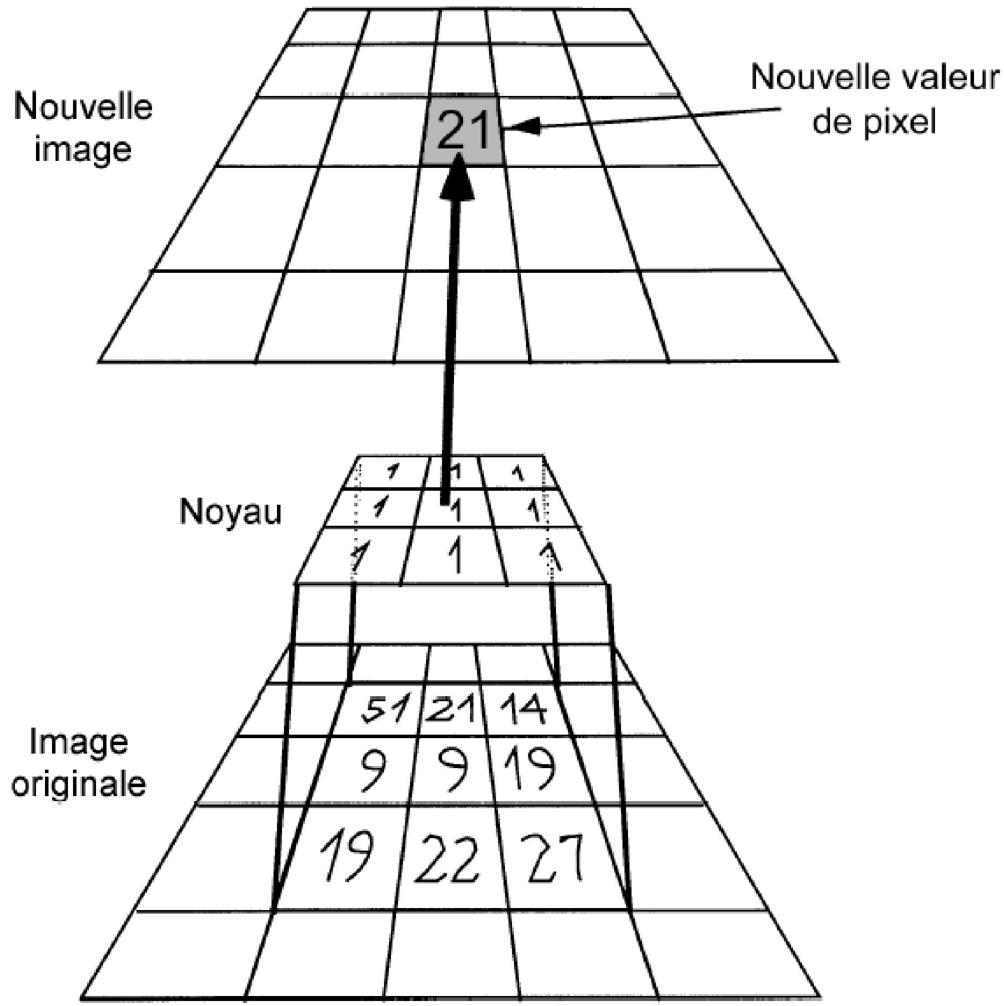
- Exemple pratique

Image

10	12	40	16	19	10
14	22	52	10	55	41
10	14	51	21	14	10
32	22	9	$\boxed{\mathbf{9}}$	19	14
41	18	19	22	27	11
10	7	8	8	4	5

Noyau

$$\frac{1}{9} * \begin{array}{c|ccc} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ \hline & -1 & 0 & 1 \end{array}$$



$$\text{Nouvelle valeur de pixel} = \left[\begin{array}{l} 1*51 + 1*21 + 1*14 + \\ 1*9 \;\; + 1*9 \;\; + 1*19 + \\ 1*19 + 1*22 + 1*27 \end{array} \right] / 9 = 21$$

Image

10	12	40	16	19	10
14	22	52	10	55	41
10	14	^{1}51	^{1}21	^{1}14	10
32	22	^{1}9	^{1}9	^{1}19	14
41	18	^{1}19	^{1}22	^{1}27	11
10	7	8	8	4	5

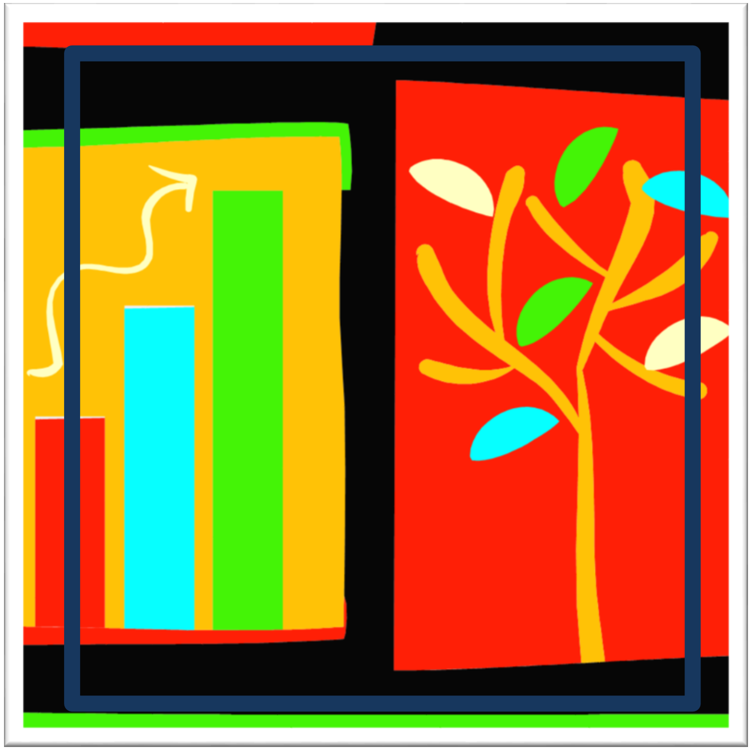
Sortie

	25				
		23			
			21		

Le traitement des bords de l’image

Première solution faire une image résultat de taille plus petite : taille initiale – taille filtre

-> embêtant…si plusieurs filtres sont appliqués successivement…



Le traitement des bords de l'image

- on complète par des 0 (imfilter avec aucune option)

-Répliquer simplement les bords

(option 'replicate')

-option 'symmetric' : on agrandit l'image par continuité en réalisant une symétrie par miroir (voir illustration)

-option 'circular', : cette fois le signal est complété en réalisant une périodisation de l'image

Cas 'replicate'

Le traitement des bords de l'image

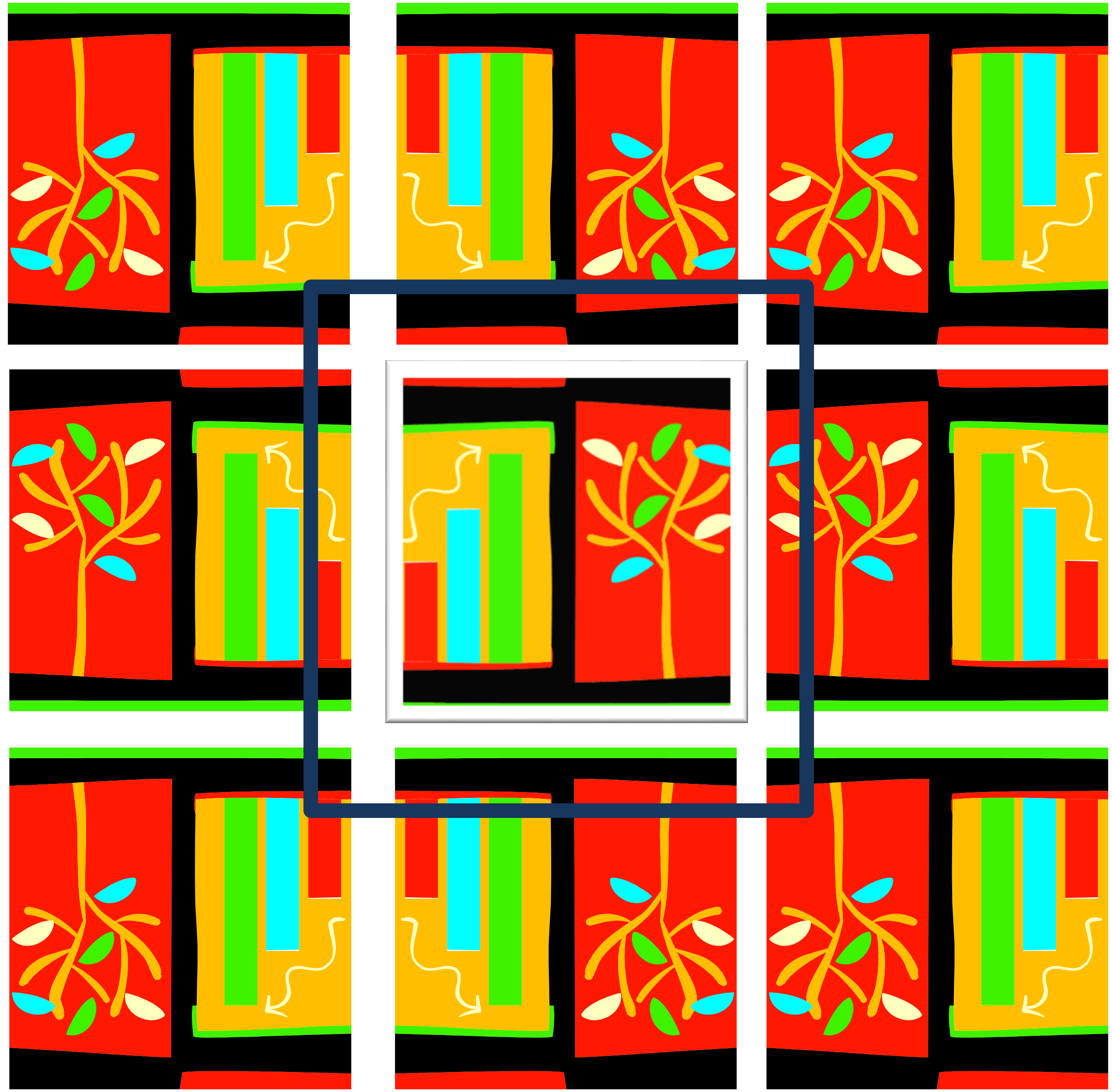
- on complète par des 0 (aucune option)

-Répliquer simplement les bords

(option 'replicate')

-option 'symmetric' : on agrandit l'image par continuité en réalisant une symétrie par miroir (voir illustration)

-option 'circular', : cette fois le signal est complété en réalisant une périodisation de l'image



Cas 'symmetric'

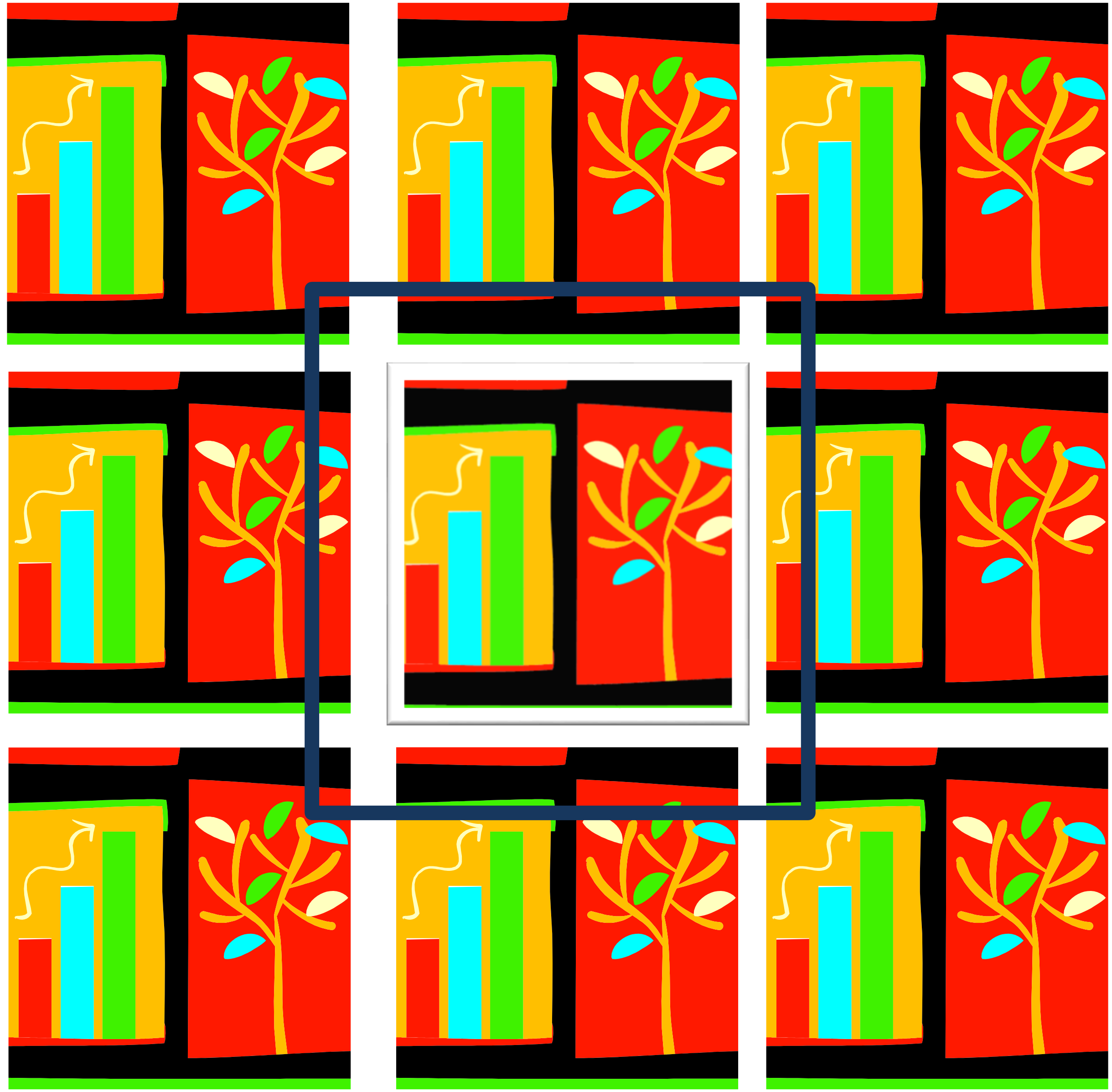
Le traitement des bords de l'image

- Aucune option : on complète par des 0
- Répliquer simplement les bords

(option 'replicate')

-option 'symmetric' : on agrandit l'image par continuité en réalisant une symétrie par miroir (voir illustration)

-option 'circular', : cette fois le signal est complété en réalisant une périodisation de l'image



Cas 'circular'

- Masques Passe-bas les plus courants

moyenne 3*3 et 5*5

$$\frac{1}{9}\begin{pmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{pmatrix} \qquad \frac{1}{25}\begin{pmatrix}1 & 1 & 1 & 1 & 1\\ 1 & 1 & 1 & 1 & 1\\ 1 & 1 & 1 & 1 & 1\\ 1 & 1 & 1 & 1 & 1\\ 1 & 1 & 1 & 1 & 1\end{pmatrix}$$

$$\frac{1}{5}\begin{pmatrix}0 & 1 & 0\\ 1 & 1 & 1\\ 0 & 1 & 0\end{pmatrix} \qquad \frac{1}{10}\begin{pmatrix}1 & 1 & 1\\ 1 & 2 & 1\\ 1 & 1 & 1\end{pmatrix}$$

Binômial (proche Gaussien)
3*3 et 5*5

$$\frac{1}{16}\begin{pmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{pmatrix} \qquad \frac{1}{140}\begin{pmatrix}1 & 1 & 6 & 1 & 1\\ 1 & 6 & 15 & 6 & 1\\ 6 & 15 & 20 & 15 & 6\\ 1 & 6 & 15 & 6 & 1\\ 1 & 1 & 6 & 1 & 1\end{pmatrix}$$

Exemple

Bruit salt & pepper

Filtrage moyenne

Filtrage median

Filtrage gaussien

Analyse quantitative usage du MSE ou PSNR

Pour analyser l'efficacité du filtre, souvent on étudie le PSNR à partir d'une image de référence que l'on bruite à différents niveaux, le filtre sera d'autant meilleur que le MSR sera faible et donc le PSNR fort

Mean Squarre Error : $$MSE = \frac{1}{M*N}\sum_i \sum_j \left(I_{ref}(i,j) - I(i,j)\right)^2$$

Peak Signal Noise Ratio: $$PSNR = 10\log_{10}\frac{(Max(I))^2}{MSE} = 20\log_{10}\frac{Max(I)}{\sqrt{MSE}}$$

Pour une image codée sur 8 bits, soit 256 niveaux de gris, Max(I)=255

Attention : $\neq$ de l' analyse qualitative

Exemple : Quantitatif $\neq$ Qualitatif

PSNR19.6899

PSNR19.8408

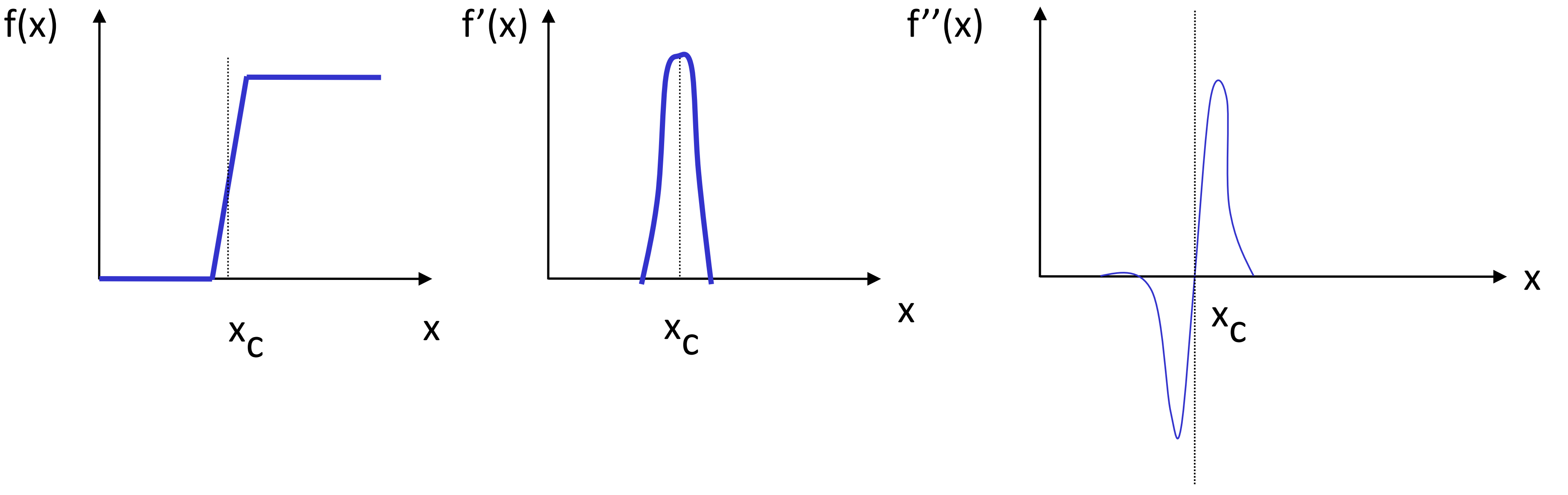
D’autres critères d’évaluations existent, SSIM etc…

• Filtres passe-haut

Les HF dans une image sont représentatives des contours.

Contour : changement rapide et d’amplitude importante de l’intensité.

Soit un échelon représentatif de ce changement de luminance : f(x)



La dérivée première d’une fonction ou/et la dérivée seconde peuvent être vues comme le résultat d’un filtrage passe-haut de l’image initiale.

! Attention le modèle de contour peut être différent, l’échelon est plus courant, on peut imaginer par exemple une rampe comme modèle

Par définition, la dérivée d'une fonction f(x) est :

$$f'(x) = \frac{df}{dx} = \lim_{h \to 0} \frac{f(x) - f(x-h)}{h} \approx f(x) - f(x-1)$$

développement au premier ordre de la dérivée.

(dans le cas numérique x est échantillonné : h=1)

Masque : h= [0 1 -1]

La dérivée première du second ordre peut être approximée par :

$$f'(x) = \frac{1}{2}\left[f(x+1) - f(x-1)\right]$$

Masque : h= [1/2 0 -1/2]

La généralisation au cas 2D est évidente : on peut toujours dériver partiellement suivant une direction puis suivant l'autre. Les masques précédents s'écrivent donc :

$$\begin{pmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \frac{1}{2}\begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

Ces masques correspondent à une dérivée suivant l'axe des y, on obtiendra donc les contours verticaux.

- Masques de dérivée première les plus courants (FIR)

$$G_x = \begin{pmatrix} -1 & -c & -1 \\ 0 & 0 & 0 \\ 1 & c & 1 \end{pmatrix} \qquad G_y = \begin{pmatrix} -1 & 0 & 1 \\ -c & 0 & c \\ -1 & 0 & 1 \end{pmatrix}$$

(Lissage suivant un axe pendant qu'on dérive suivant l'autre)

On les appelle masques de Prewitt : c=1

ou masques de Sobel : c=2

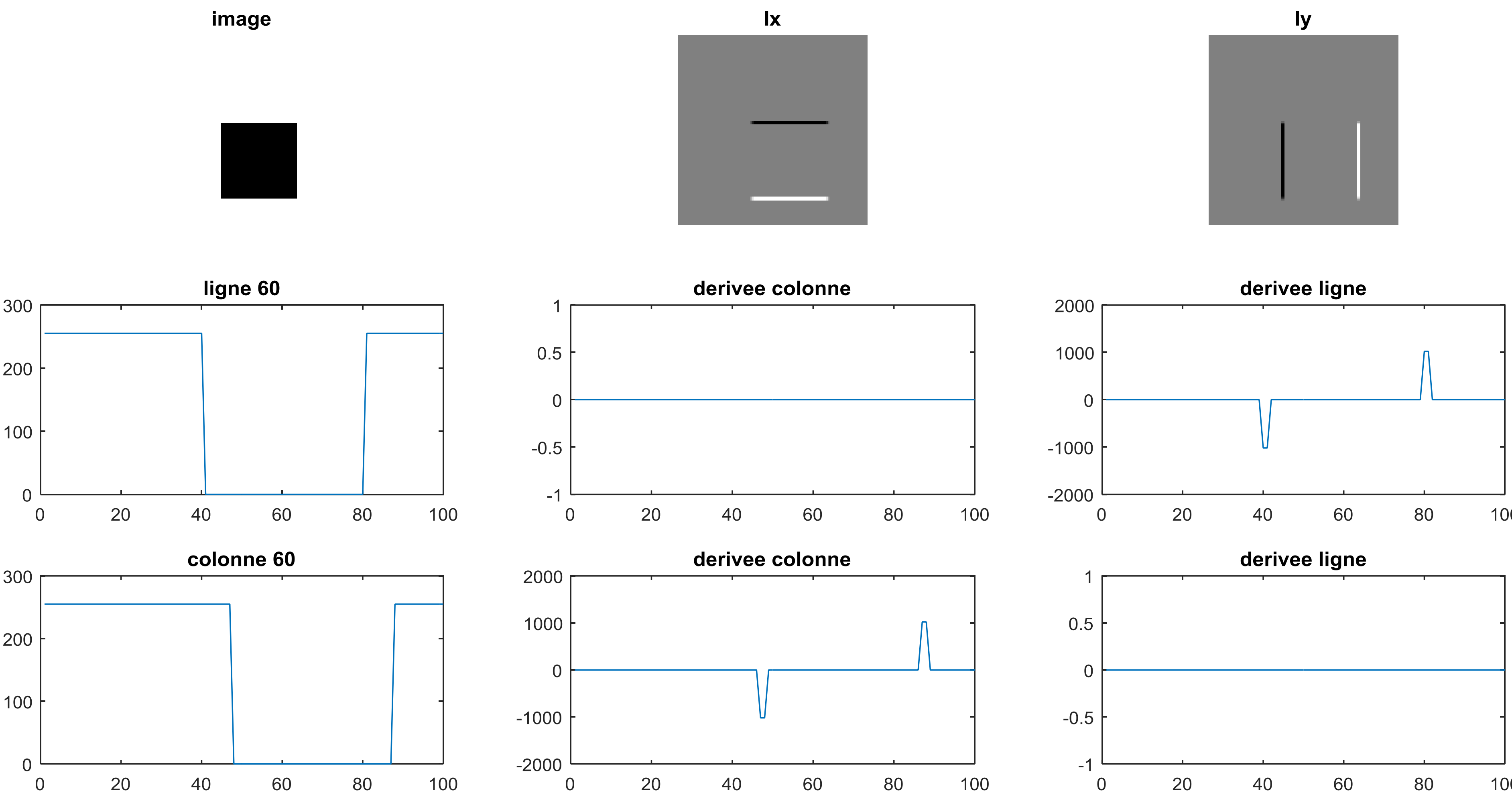
Il est possible d'utiliser des filtres de taille plus grande (5*5) : moins précis spatialement, ils ont toutefois l'avantage de diminuer le bruit : compromis.

La somme des coefficients pour un filtre extracteur de contours doit toujours être nulle.

$$\begin{pmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

(On verra plus tard dans l'étude de détection de contours les filtres iir)

Exemple : Filtre de Sobel



(Attention à l'implémentation en matlab, imfilter ou filter2 utilisent la corrélation et non la convolution, il faut mettre l'option 'conv')

- Masques de dérivée seconde

Développement de Taylor : $f(x+1)=f(x)+f'(x)+\frac{1}{2}f''(x)$ autour de x

$\Rightarrow f''(x)=2f(x+1)-4f(x)+2f(x-1)$ (En reprenant l'expression de f' vue précédemment)

$$\Rightarrow \text{masque de convoultion} \; : \; \frac{\partial^2 f}{\partial y^2}=2*\begin{pmatrix}0 & 0 & 0\\ 1 & -2 & 1\\ 0 & 0 & 0\end{pmatrix} \text{et} \; \frac{\partial^2 f}{\partial x^2}=2*\begin{pmatrix}0 & 1 & 0\\ 0 & -2 & 0\\ 0 & 1 & 0\end{pmatrix}$$

$$\Rightarrow \text{masque du Laplacien} \; : \; \frac{\partial^2 f}{\partial x^2}+\frac{\partial^2 f}{\partial y^2}=2*\begin{pmatrix}0 & 1 & 0\\ 1 & -4 & 1\\ 0 & 1 & 0\end{pmatrix}$$

$$\text{autres, par ex} : \begin{pmatrix}1 & -2 & 1\\ -2 & 4 & -2\\ 1 & -2 & 1\end{pmatrix}$$

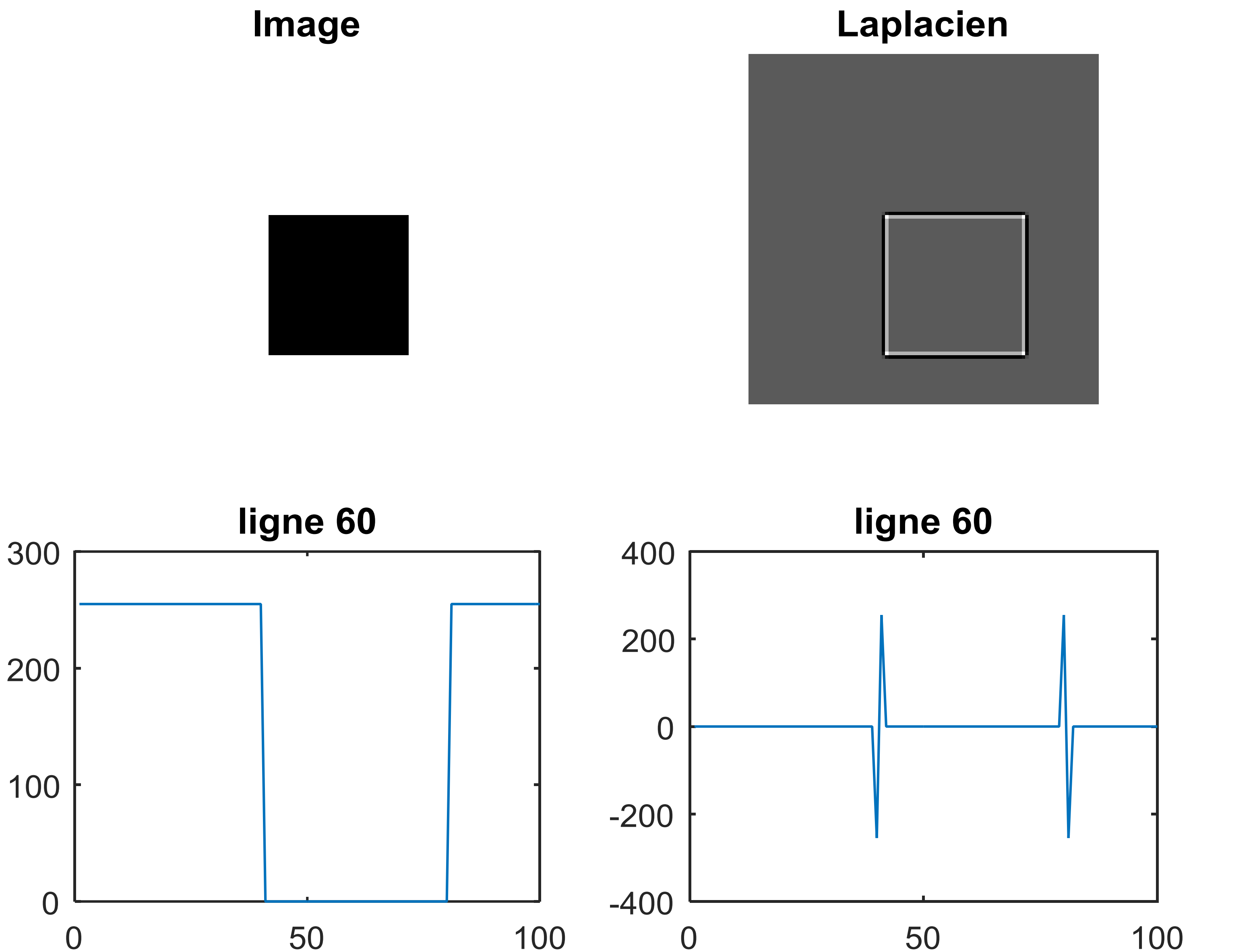
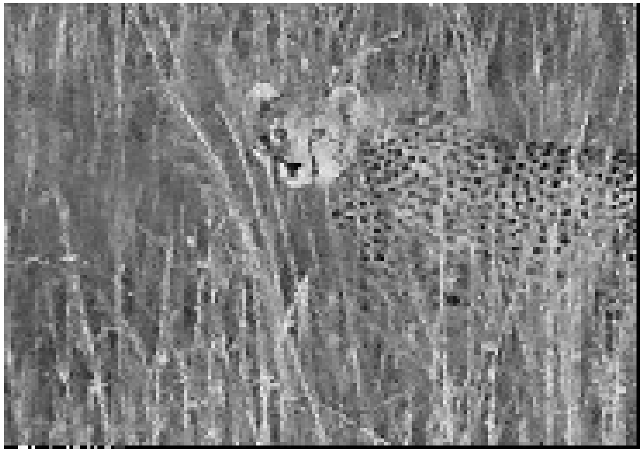
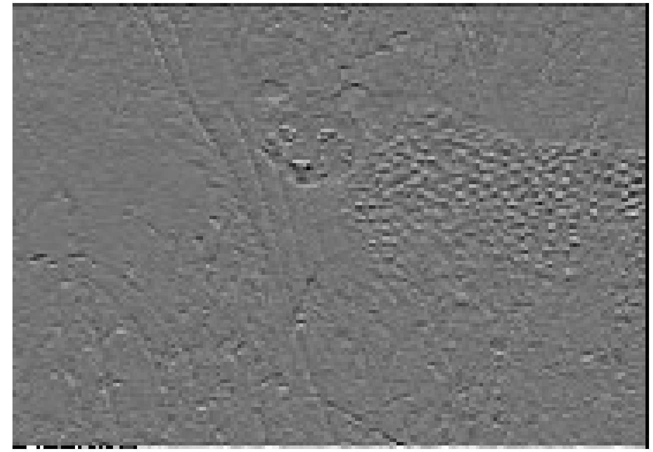


Image
Laplacien
ligne 60
300
200
100
0
0
50
100
ligne 60
400
200
0
-200
-400
0
50
100

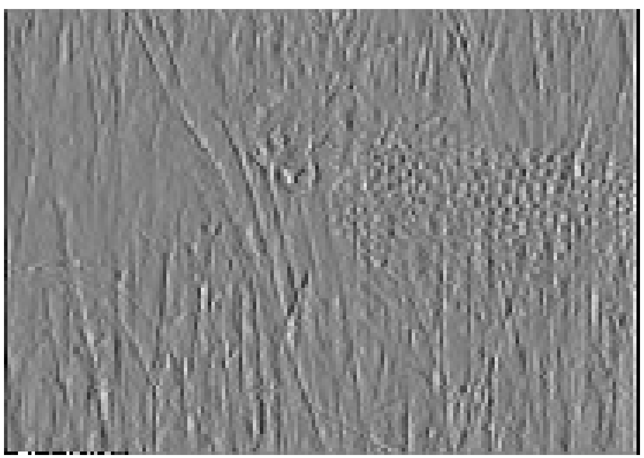
original



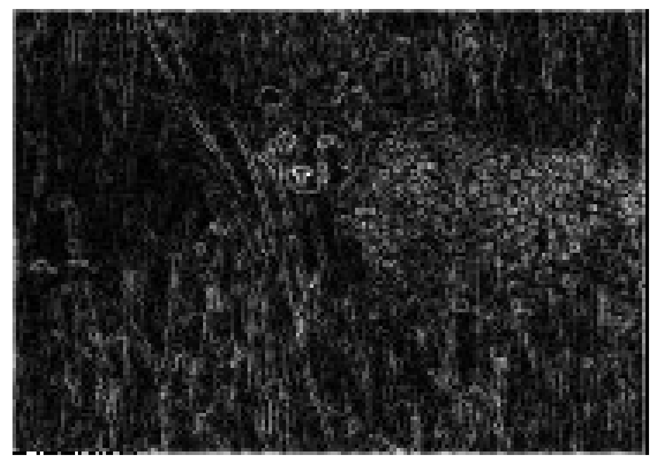
filtre Gy



filtre Gx

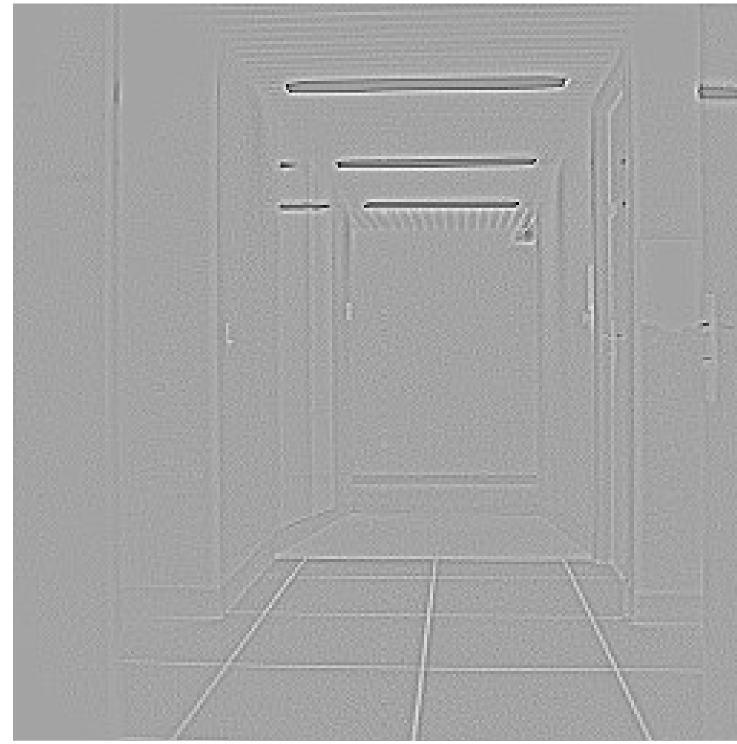


norme de G



originale

Laplacien



- Exercices
  - Séparabilité d’un filtre

•Un filtre 2D est dit séparable si on peut appliquer 2 filtres 1D, un vertical et un horizontal.

–Ex : montrer qu’il est équivalent d’appliquer le masque 1/3*[1 1 1] en horizontal suivi du vertical que d’appliquer le masque moyenne 3*3.

  - Cascade de filtres

•Ex : vous appliquez le filtre moyenne 3*3 plusieurs fois (par ex 2 fois). Quel est le masque équivalent? (raisonnez en 1D)

•Ex : faire de même pour le Gaussien : que remarquez-vous pour l’obtention des coefficients?

- Application : le réhaussement de contour

Le réhaussement de contour se comprend facilement dans le domaine spatial. On désire que l' image ait des contours plus marqués : on peut donc dire que une méthode simple de réhaussement de contour consiste à rajouter à l'image ses propres contours. On utilise le Laplacien.

$$I_f = h3 * I = h1 * I + h2 * I = (h_1 + h_2) * I$$

$$h_2 = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

$$h_3 = h_1 + h_2 = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

$$h_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Attention : renormalisez les valeurs après calculs…

Exemple

Image originale

Image contours réhaussés

image originale

image réhaussée passe-haut linéaire

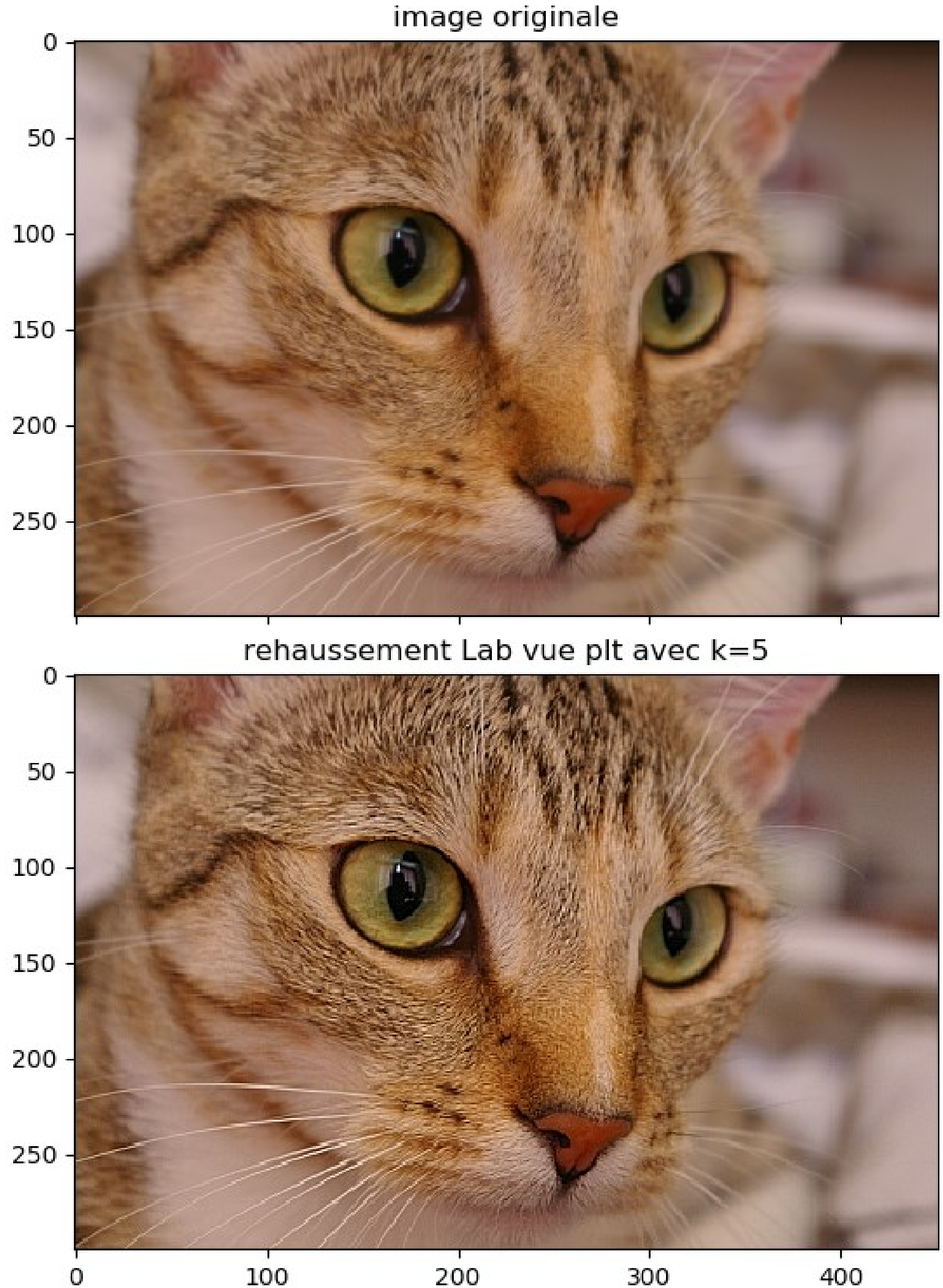


image originale
0
50
100
150
200
250
rehaussement Lab vue plt avec k=5
0
50
100
150
200
250
0
100
200
300
400

Filtres fréquentiels

Définition

Soit f(x,y) une fonction 2D (par exemple l'image), F(u,v) est sa TF. Soit g(x,y) (G(u,v) sa TF) l'image filtrée par un filtre de réponse impulsionnelle h(x,y), soit H(u,v) le filtre fréquentiel alors :

$$\mathbf{G(u,v)=H(u,v).F(u,v)\ \text{->}\ g(x,y)=h(x,y)*f(x,y)}$$

Equivalent à la convolution dans le domaine spatial, il est parfois plus pratique de travailler dans le domaine fréquentiel (ex : on connaît les fréquences indésirables etc…). De plus l'implémentation dans le domaine fréquentiel est souvent plus aisée (simple multiplication).

- Filtres passe-bas

Le passe-bas idéal

$$H(u,v)=\begin{vmatrix} 1 & \text{si}\, D(u,v) \leq D_0 \\ 0 & \text{si}\, D(u,v) > D_0 \end{vmatrix}$$

idéal en fréquence mais rebonds très importants dans le domaine spatial

D_0 est une constante non négative : fréquence de coupure
D(u,v) représente la distance entre un point (u,v)
et l'origine (0,0).

$$D(u,v)=\sqrt{u^2+v^2}$$

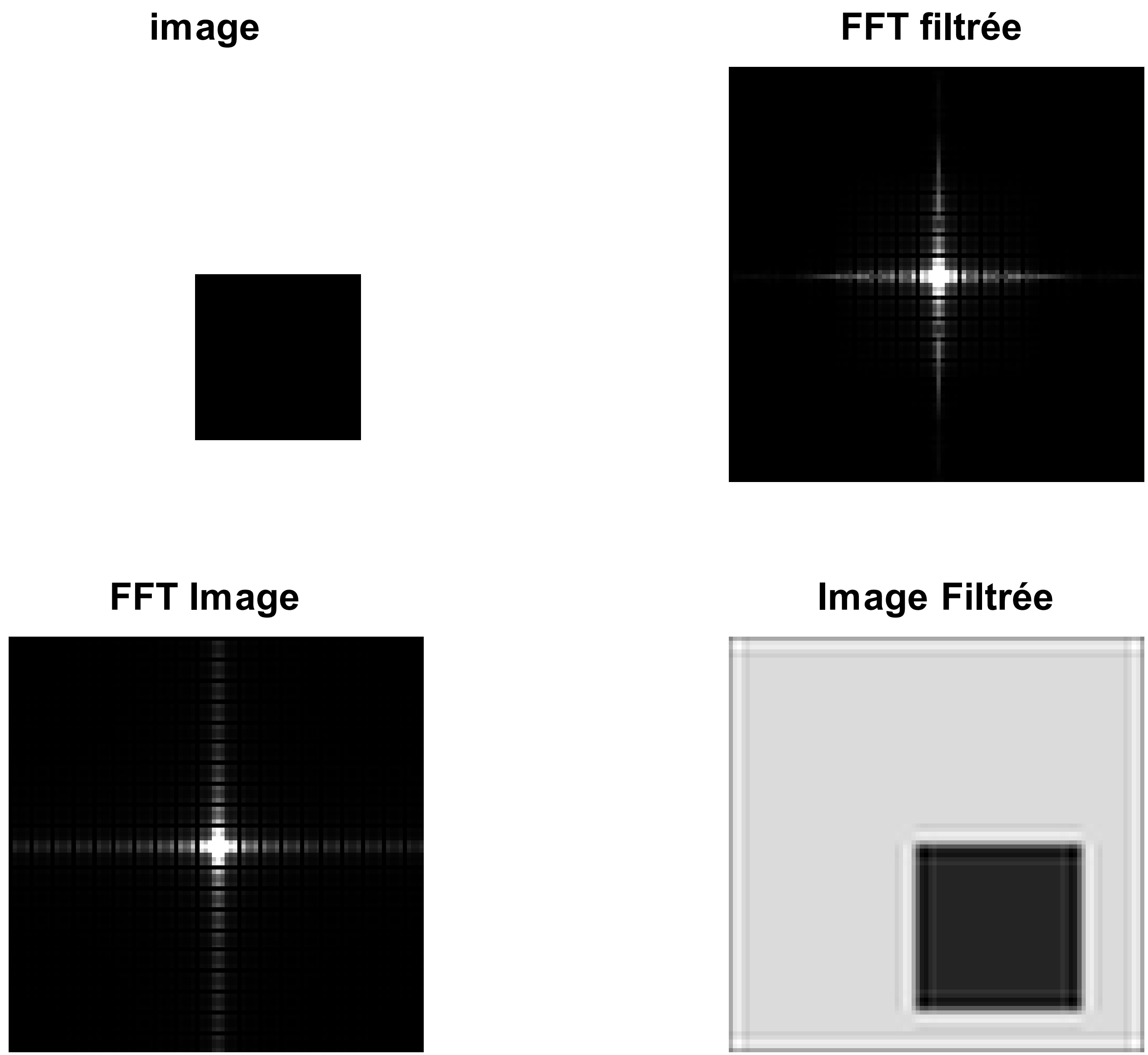


image
FFT filtrée
FFT Image
Image Filtrée

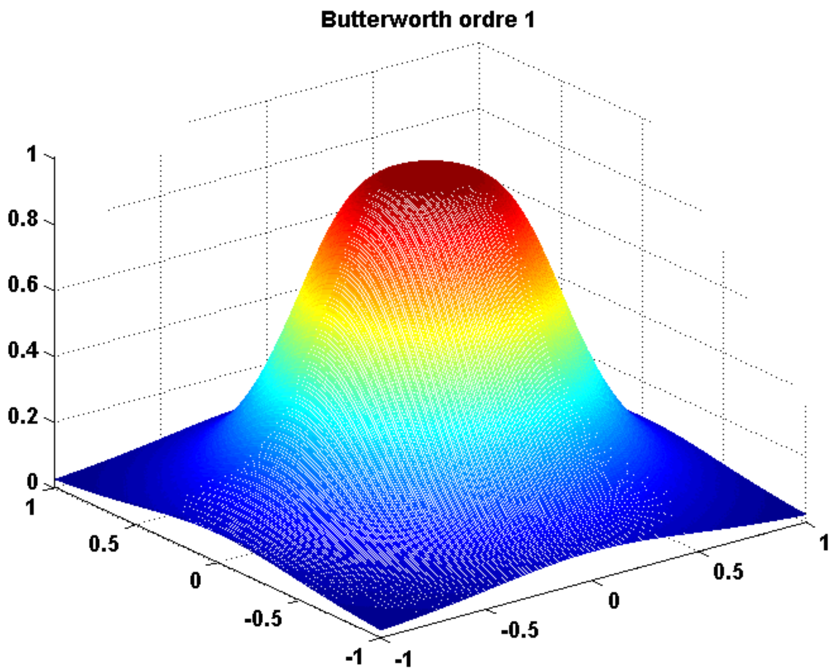
•Filtre passe-bas de Butterworth

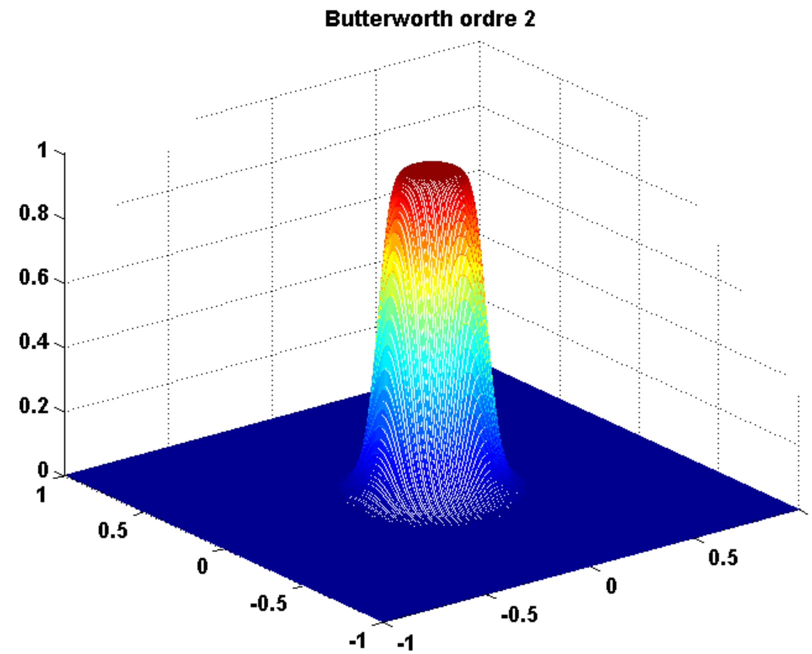
–Filtre optimal suffisamment lissé dans le domaine fréquentiel pour limiter les rebonds dans le domaine spatial.

$$\mathbf{H}(\mathbf{u},\mathbf{v}) = \frac{1}{1+0.414 * [\mathbf{D}(\mathbf{u},\mathbf{v}) / \mathbf{D}_0]^{2\mathbf{n}}}$$

où D(u,v) suit l'équation précédente du filtre idéale, c'est donc la distance par rapport à l'origine.
D_0 est la fréquence de coupure désirée.

Les filtres de Butterworth permettent de régler le compromis entre sélectivité de la fréquence de coupure et diminution des effets de rebonds


Butterworth ordre 1
1
0.8
0.6
0.4
0.2
0
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1


Butterworth ordre 2
1
0.8
0.6
0.4
0.2
0
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

Sous Matlab

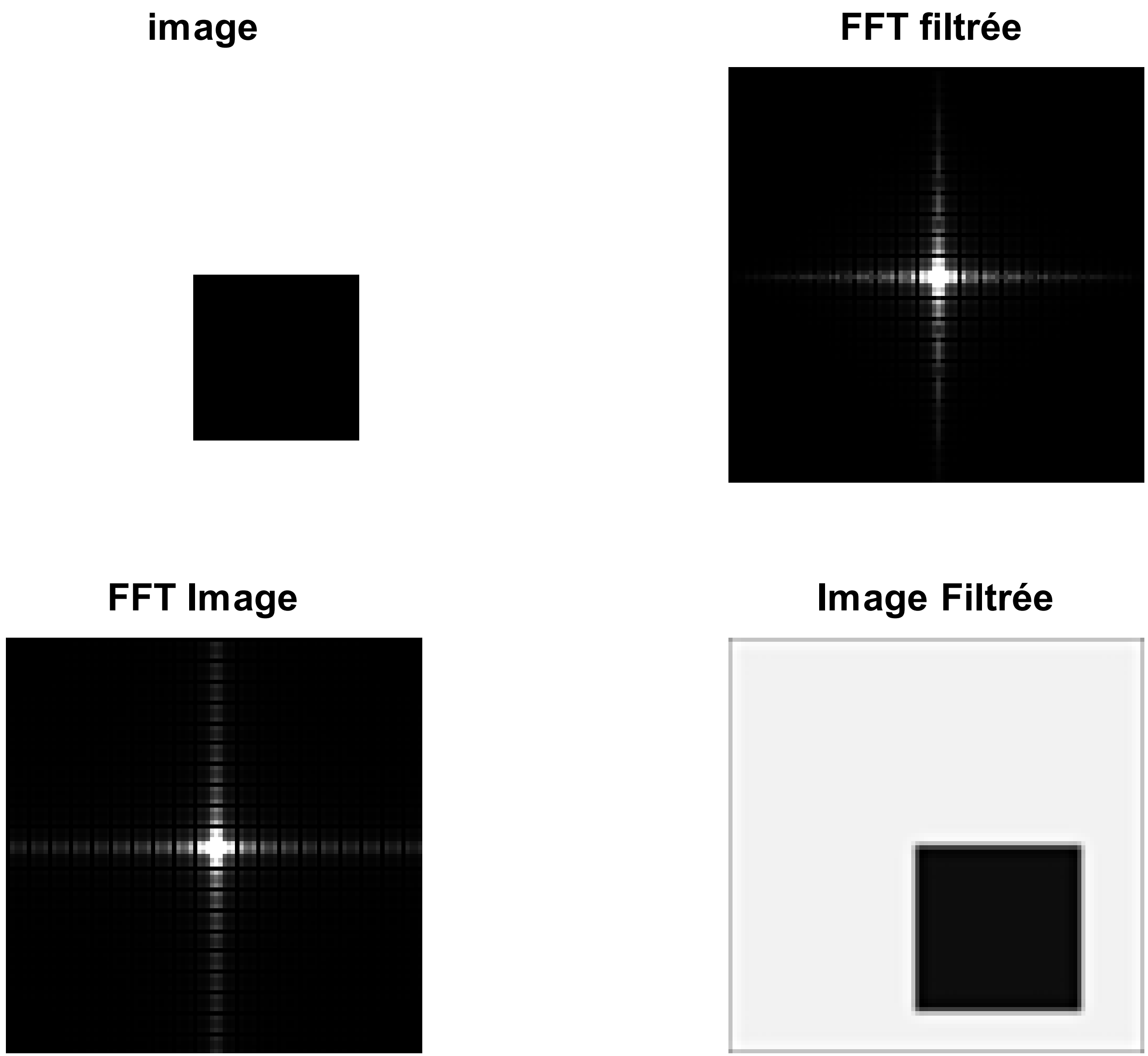
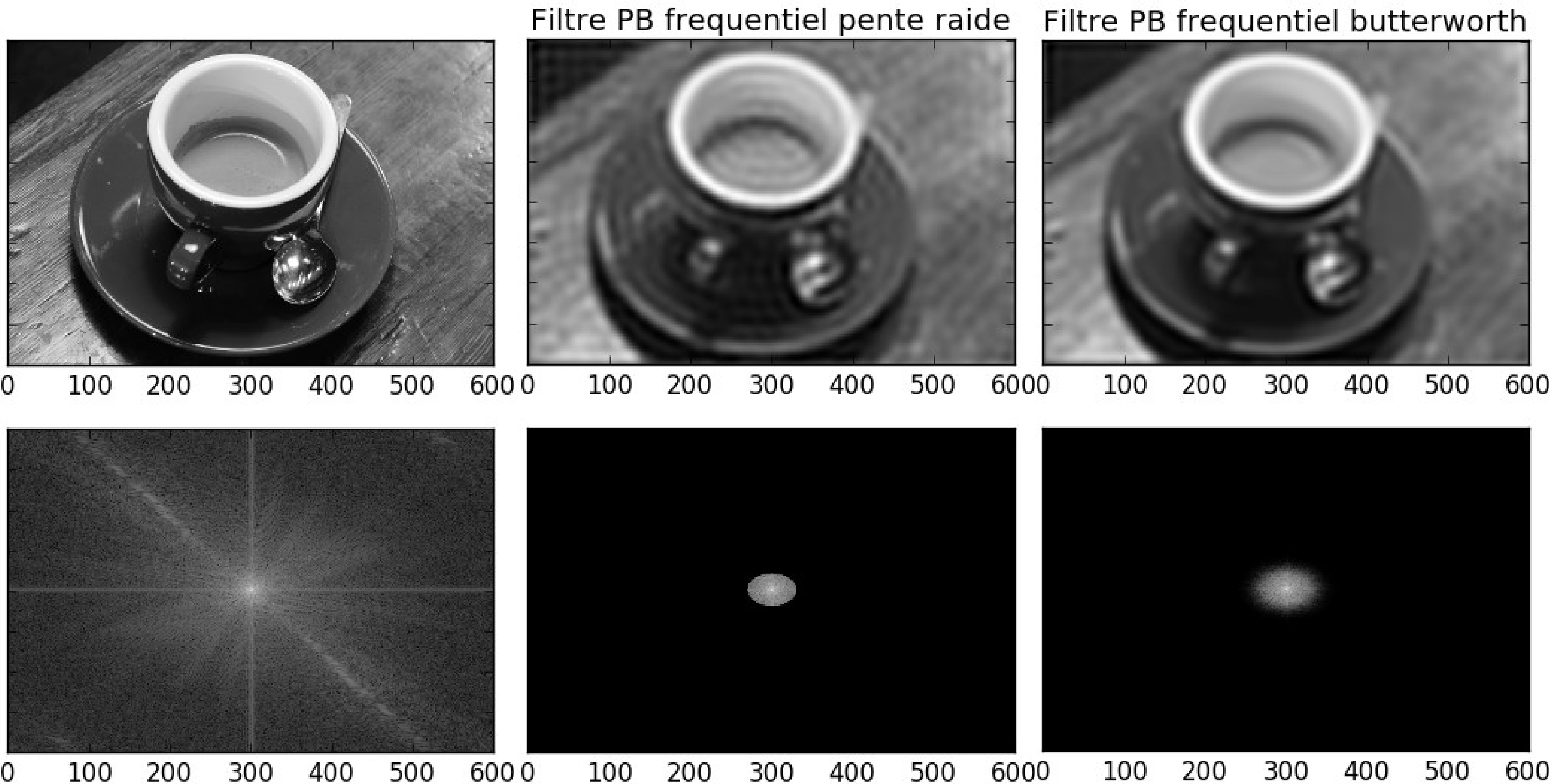


image
FFT filtrée
FFT Image
Image Filtrée

Exemple

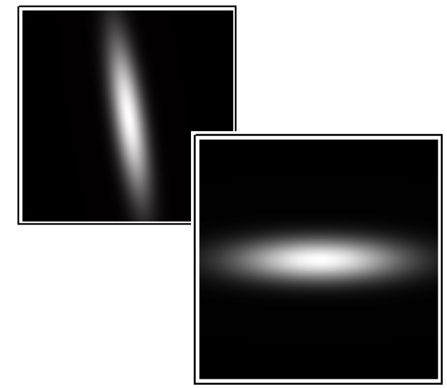


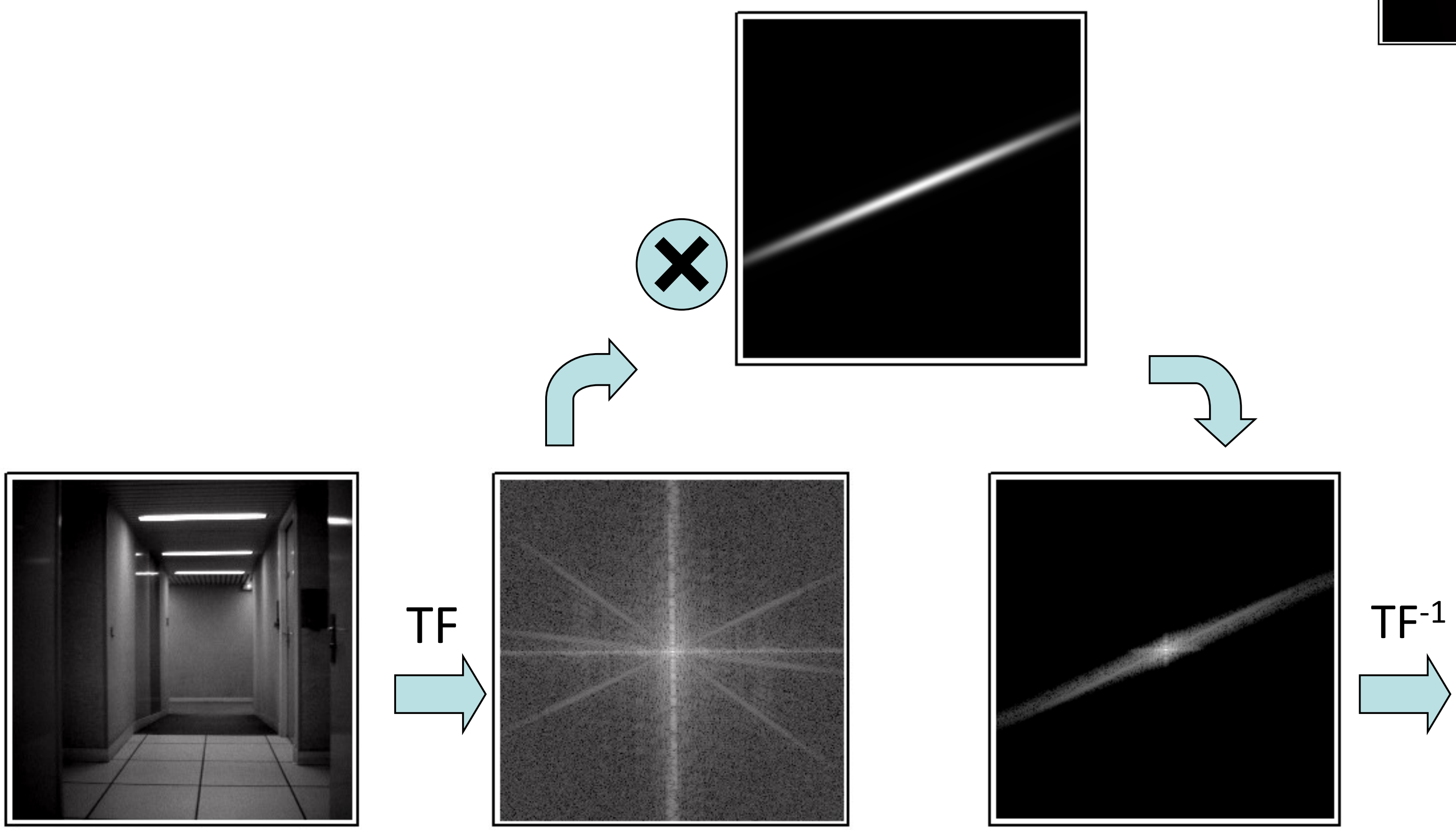
Fc=0.05

Butterworth ordre 1

Filtrage fréquentiel

Sélection d'une fréquence : exemple les filtres de Gabor -> extraction d’une direction privilégiée





– Relation entre masque de convolution et réponse fréquentielle

Très simple, si on connaît le filtre spatial h, il suffit d'appliquer la transformée de Fourier du filtre.

$$\textit{Soit}\ h(k,l)\ \text{la réponse impulsionnelle}\ (k,l) : [-n;+n]X[-m,+m]$$

$$TF(h(k,l)) = H(u,v) = \sum_{-\infty}^{\infty}\sum_{-\infty}^{\infty} h(k,l)e^{-2i\pi(uk+vl)}$$

- Exercice
  - Quel est le filtre fréquentiel du masque h=1/3[1 1 1]
- Exercice
  - Soit I_2 l’image obtenue en soustrayant de l’image originale I_0 l’image I_1 filtrée à l’aide d’un filtre moyenne 3*3.

•Donner l’expression matricielle du filtre (masque de convolution) permettant de passer directement de I_0 à I_2.

•Calculer sa fonction de transfert.

•De quel type est-il?

Implémenter un filtre fréquentiel

- Soit H filtre défini dans le domaine fréquentiel
  - Implémentation dans le domaine fréquentiel : simple multiplication avec la TF de l'image
  - Implémentation dans le domaine spatial : TF inverse du filtre H -> h mais h peut être infini…revoir la synthèse de filtres numérique en TS!

Synthèse de filtres RIF discret, implémentation dans le domaine spatial

Simple extension du cas 1D, il existe plusieurs méthodes :

* Echantillonnage en fréquence

A partir d'une fonction connue que l'on considère comme une fonction idéale $H_i(u,v)$, on prélève un nombre d'échantillons G(m,n) assimilés aux coefficients de la TFDB. La TF inverse de ces échantillons donnera alors les coefficients h(k,l).

Mais la réponse fréquentielle ne s'identifie vraiment qu'aux échantillons H(m,n) => on considère souvent les H(m,n) comme des paramètres que l'on cherche à ajuster par des méthodes d'optimisation (programmation linéaire) qui cherchent à minimiser l'erreur entre $H_i(u,v)$ et la réponse effective du filtre.

* Limitation de l'étendue du filtre dans le domaine spatial par une fonction fenêtre

Hi(u,v) très souvent contient des transitions rapides entre bande de fréquence => hi(k,l) est forcément à étendue infinie => elle ne peut donc pas provenir d'un filtre RIF (par def.) => hi(k,l) doit être multipliée par une fenêtre d'apodisation.

$$h(k,l) = h_i(k,l)w(k,l)$$
$$H(u,v) = H_i(u,v) * W(u,v)$$

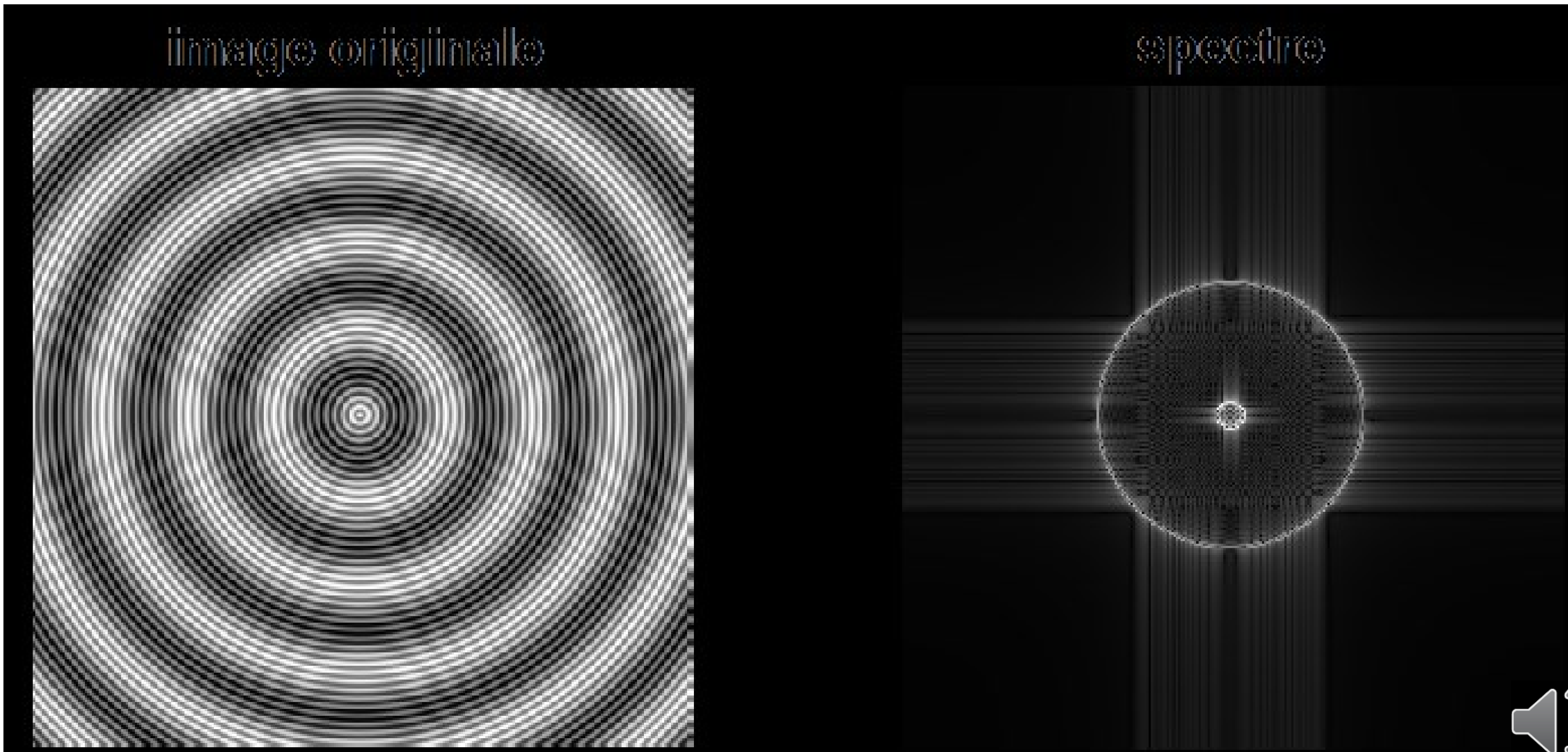
Le pb : trouver W qui minimise l’écart entre H et Hi.
On montre que l’on peut généraliser les résultats 1D : si u(k) est 1 bonne fenêtre 1D, alors :

$$\mathbf{w(k,l)} = \mathbf{u}(\sqrt{\mathbf{k}^2 + \mathbf{l}^2})$$ est une bonne fenêtre 2D

En pratique, reprendre Hamming, Hanning etc…et les généraliser comme précédemment

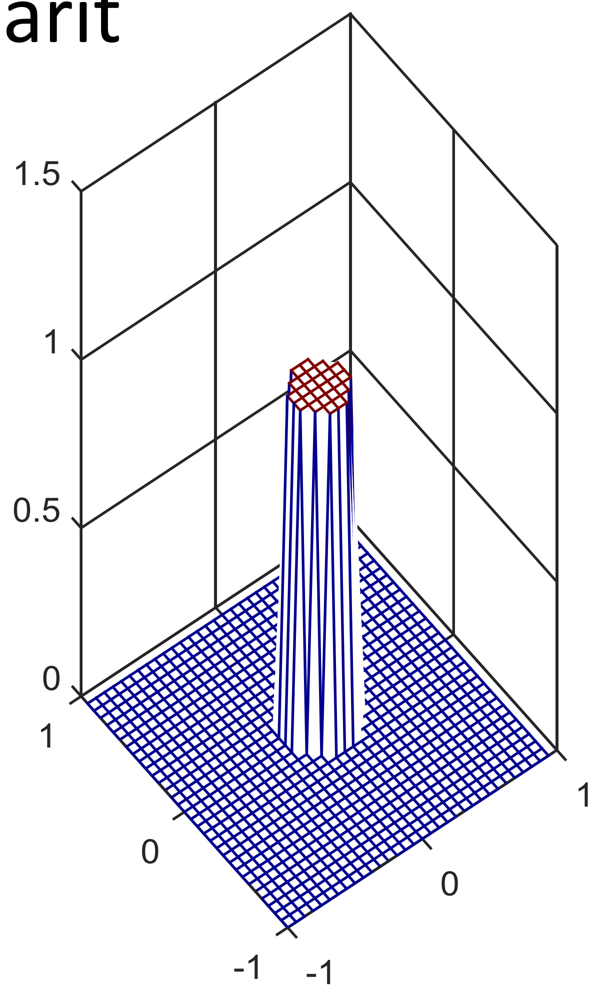
Exemple synthèse de filtre

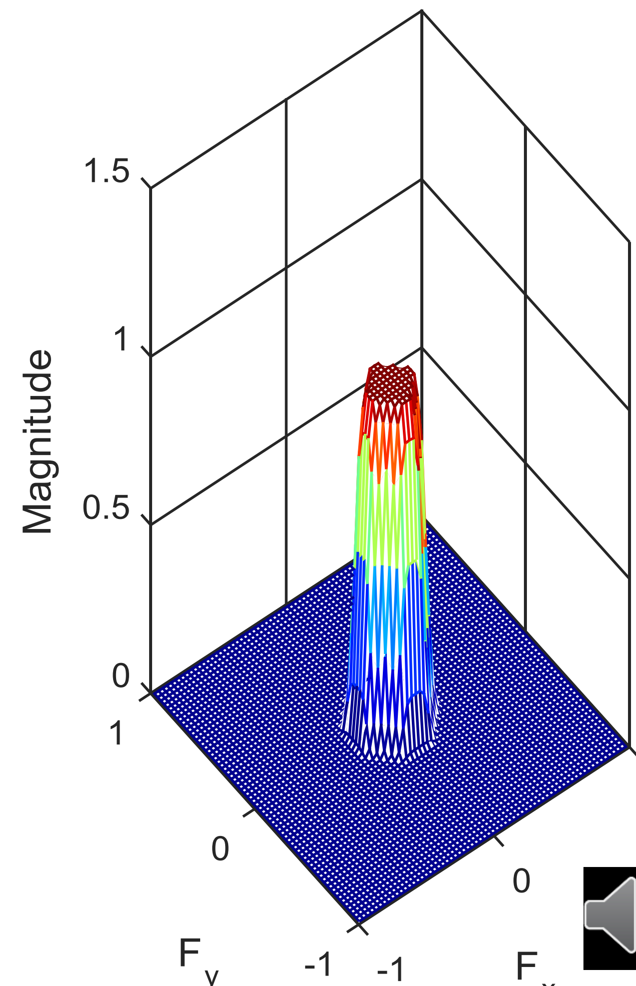
– Une image comportant deux fréquences 0.02 et 0.2 doit être sous-échantillonnée avec Te=4 pixels



Définition du filtre

- Fc=1/4/2 (shannon)
- Définir le gabarit
- Réponse fréquentielle du filtre





Résultat

image sous-echantillonnée

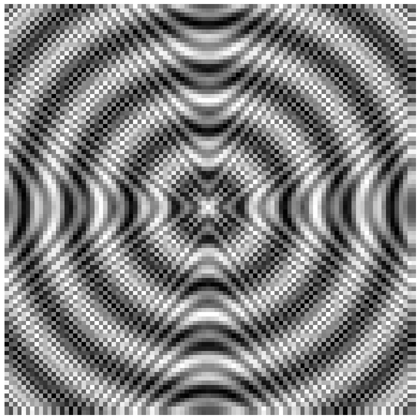
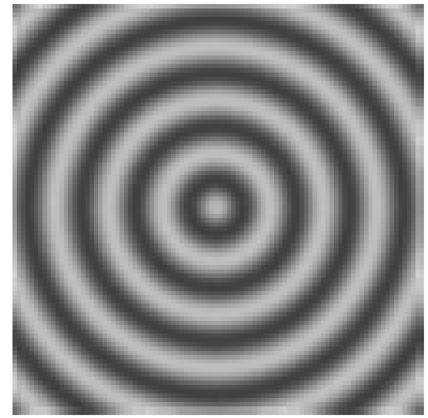
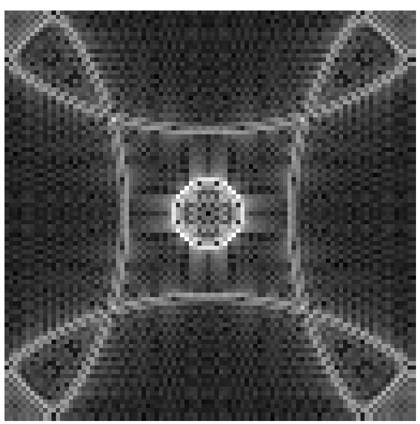


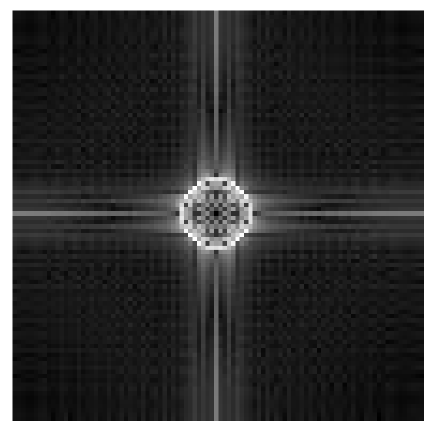
image filtrée sous-échantillonnée



spectre



spectre



Filtres passe-haut fréquentiel

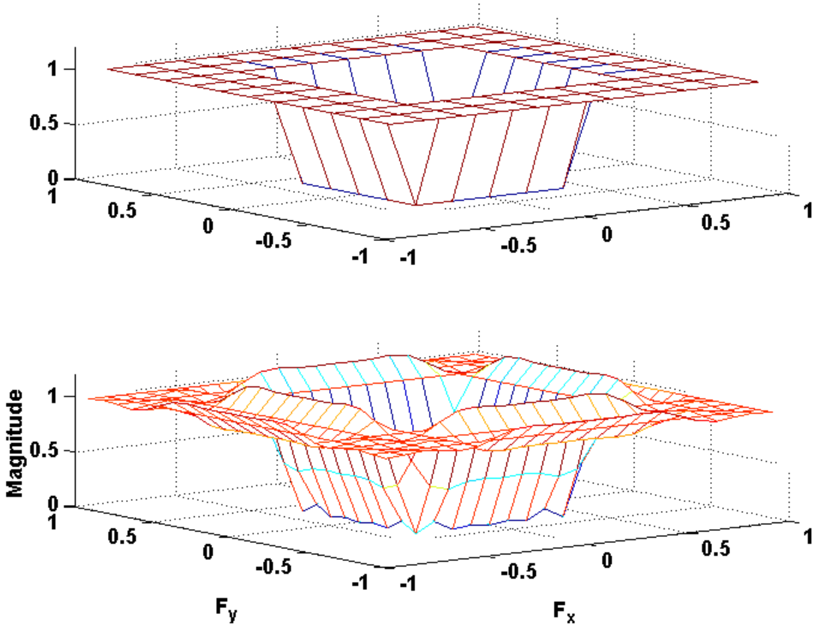
•Filtre passe-haut idéal

$$H(u,v) = \begin{vmatrix} 1 & \text{si } D(u,v) \geq D_0 \\ 0 & \text{si } D(u,v) < D_0 \end{vmatrix}$$

idéal en fréquence mais rebonds très importants dans le domaine spatial

D_0 est une constante non négative : fréquence de coupure
D(u,v) représente la distance entre un point (u,v)
et l'origine (0,0).

$$D(u,v) = \sqrt{u^2 + v^2}$$



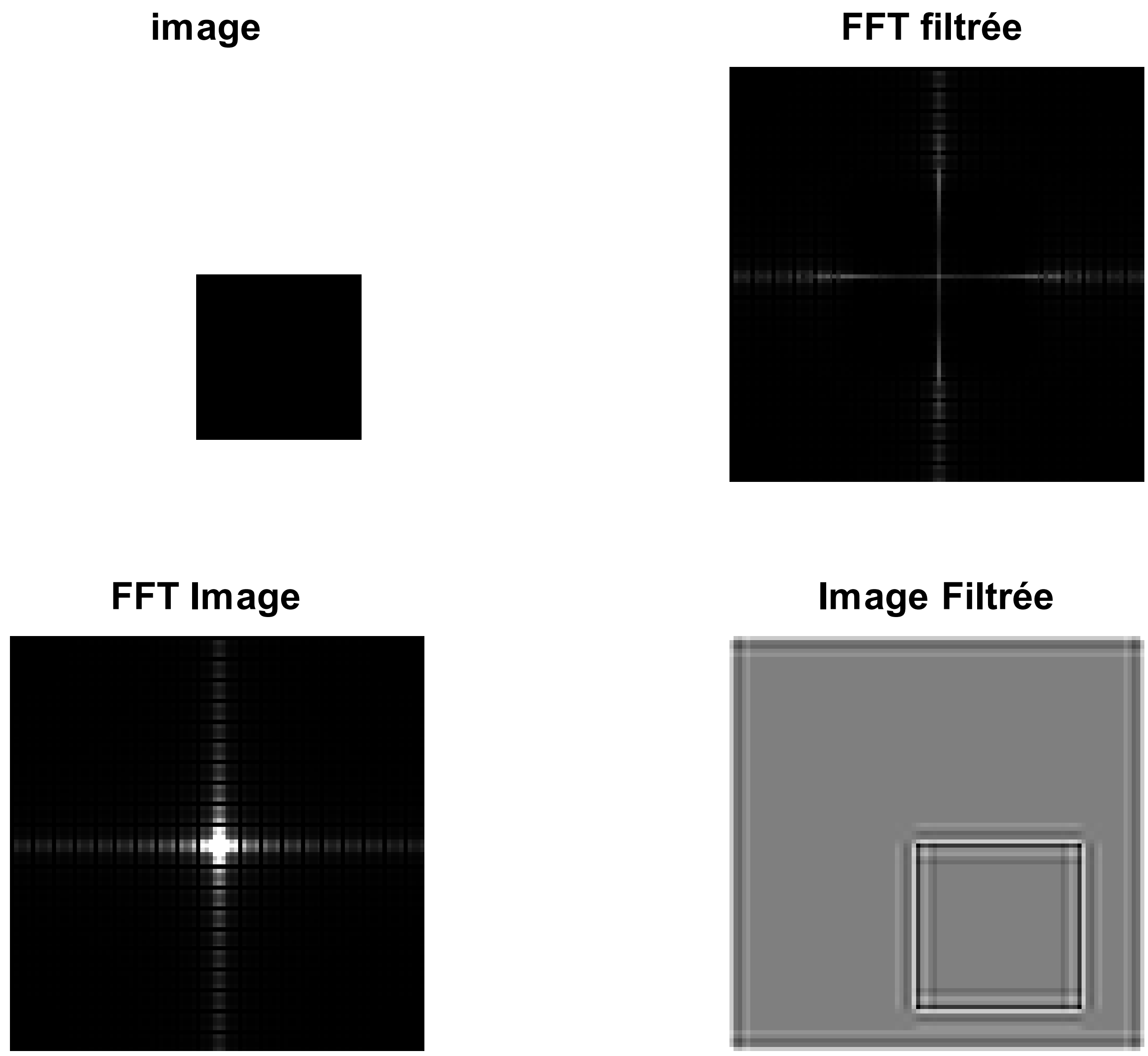


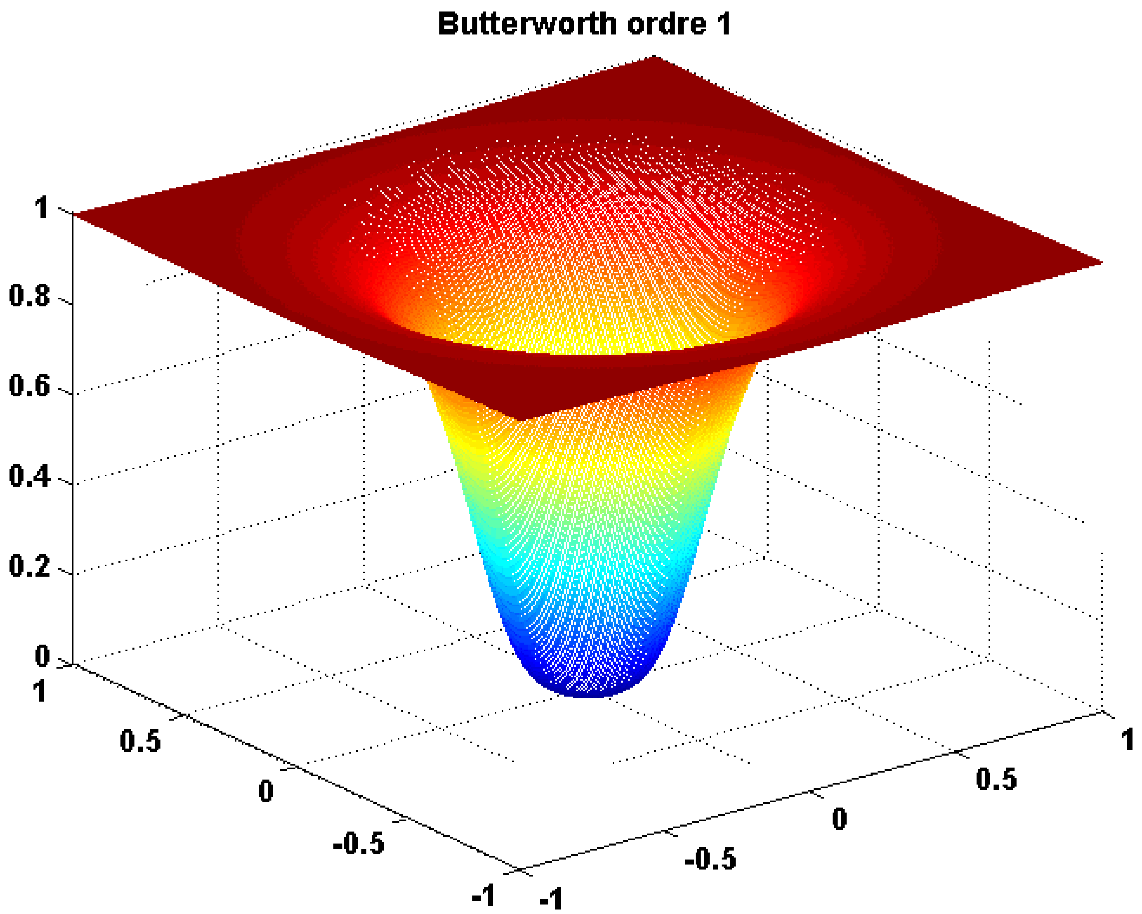
image
FFT filtrée
FFT Image
Image Filtrée

- Filtre passe-haut de Butterworth
  - Filtre optimal suffisamment lissé dans le domaine fréquentiel pour limiter les rebonds dans le domaine spatial.

$$\mathbf{H}(\mathbf{u},\mathbf{v}) = \frac{1}{1+0.414*\left[\mathbf{D}_0/\mathbf{D}(\mathbf{u},\mathbf{v})\right]^{2\mathbf{n}}}$$

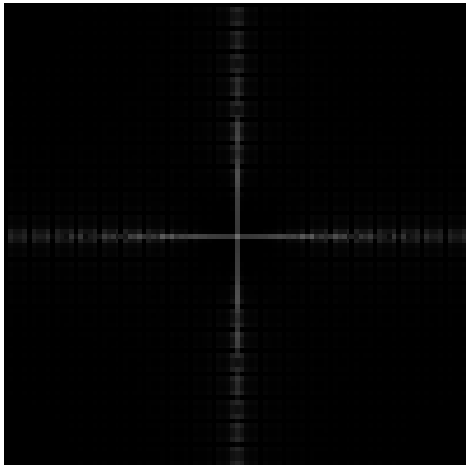
où D(u,v) suit l'équation précédente du filtre idéale, c'est donc la distance par rapport à l'origine.
D_0 est la fréquence de coupure désirée.

Les filtres de Butterworth permettent de régler le compromis entre sélectivité de la fréquence de coupure et diminution des effets de rebonds


Butterworth ordre 1
1
0.8
0.6
0.4
0.2
0
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1

image

FFT filtrée



FFT Image

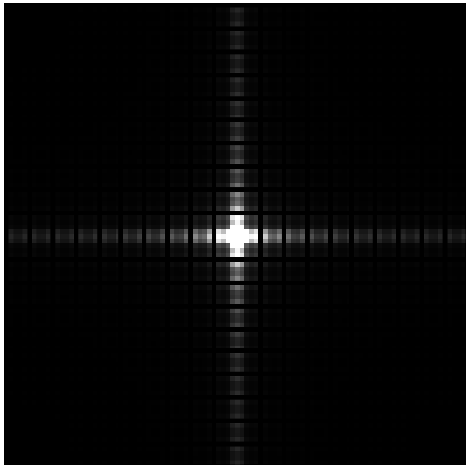
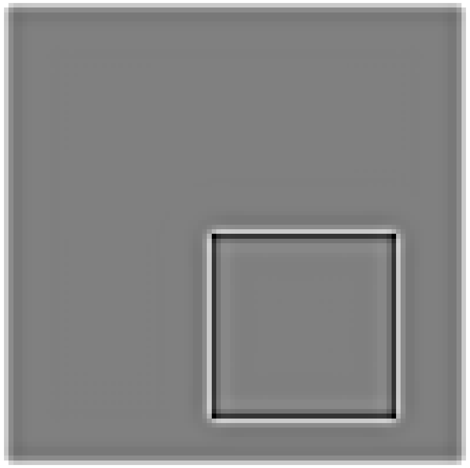


Image Filtrée



Filtrage linéaire adaptatif

L'image est un processus que l'on ne peut pas classer de stationnaire => il est naturel d'envisager de traiter les images avec des opérateurs adaptatifs, par exemple selon la position dans l'image :

- comportement passe-bas
- comportement filtre identité (préservation des frontières)
- comportement passe-haut (réhaussement des frontières)

En Résumé : Opérateurs linéaires mais décision locale.

– Moyenne adaptative

$$I'(s) = \frac{\sum_{p=1}^{P} c_p I_p}{\sum_{p=1}^{P} c_p}$$

Avec p : pixel et s est le pixel courant (i,j)
P : la fenêtre d’analyse
I(p) : intensité du pixel p

$$\begin{vmatrix} c_p = 1 \quad \text{si} \quad |I_p - I_s| \le \tau \\ c_p = 0 \quad \text{sinon} \end{vmatrix}$$

s est le pixel courant

On ne fait que la moyenne des pixels « proches »
Le seuil doit être assez grand pour que la moyenne fonctionne
Le seuil doit être inférieur à la valeur de transition des régions

Filtre bilatéral

- Filtre passe-bas dont les poids des pixels appartenant au noyau sont calculés en fonction de la distance au pixel traité (comme pour les filtres linéaires classiques) et en fonction de la différence de niveaux de gris (ou de couleur)

Résultats filtre bilatéral