

Chapitre 2

Filtrage et restauration

2.1 Introduction

Dans ce chapitre, on s'intéresse à l'*amélioration* d'une image numérique, pour augmenter la qualité de son rendu visuel, ou à sa *simplification*, dans le but de faciliter une analyse ultérieure. Ces deux problèmes sont différents mais font appel aux mêmes outils. Dans un cas, on cherche à atténuer, sinon supprimer, une certaine dégradation, dans l'autre on cherche à supprimer des structures inutiles (du point de vue de l'analyse visée).

Plaçons-nous d'abord dans le cas d'une dégradation due à un *bruit additif* : On observe une image g qui est la version dégradée d'une image f par l'ajout d'un signal aléatoire b : $g(x) = f(x) + b(x)$. L'objectif est donc de retrouver, autant que possible, l'image initiale f .

Le filtrage linéaire dit *idéal* repose sur le principe suivant : Si on ne peut pas «soustraire» $b(x)$ à $g(x)$ dans le domaine spatial (b étant aléatoire, donc inconnu), on peut tenter de le faire dans le domaine fréquentiel, moyennant des hypothèses sur la répartition des composantes fréquentielles de b par rapport à celles de f . Par linéarité de la transformée de Fourier, on a : $G(u) = F(u) + B(u)$. Le filtrage idéal consiste à faire l'hypothèse que le support de B est connu, et qu'il est distinct du support de F . Dans ce cas, multiplier G par la fonction indicatrice du complémentaire du support de B (fonction notée $\mathbb{1}_{\text{Supp}(B)^c}$) est équivalent à soustraire B à G , et donc par transformée inverse, à retrouver f :

$$\text{Supp}(B) \cap \text{Supp}(F) = \emptyset \Rightarrow G(u) \cdot \mathbb{1}_{\text{Supp}(B)^c}(u) = G(u) - B(u) = F(u) \quad (2.1)$$

Cette multiplication dans le domaine fréquentiel correspond à une opération de filtrage linéaire. La même opération peut aussi être réalisée directement dans le domaine spatial par la *convolution*. Soit g une image numérique. h une fonction discrète bidimensionnelle de support $\{a_1, a_2\} \times \{b_1, b_2\}$. La convolution de g par h est définie comme suit :

$$(g \star h)[x, y] = \sum_{i=a_1}^{a_2} \sum_{j=b_1}^{b_2} h[i, j] \cdot g[x - i, y - j] \quad (2.2)$$

L'opération de convolution de l'image g par la fonction h , dite *noyau de convolution* consiste donc à remplacer la valeur de chaque pixel $g[x, y]$ par une combinaison linéaire de ses voisins, où le coefficient attaché à chaque voisin $(x - i, y - j)$ est déterminé par la valeur $h[i, j]$ du noyau de convolution.

La convolution est une opération commutative :

$$g \star h = h \star g \quad (2.3)$$

La convolution est associative :

$$(g \star h) \star k = g \star (h \star k) \quad (2.4)$$

La convolution est distributive par rapport à l'addition :

$$h \star (g + k) = (h \star g) + (h \star k) \quad (2.5)$$

La convolution dans le domaine spatial correspond au produit dans le domaine fréquentiel, et réciproquement. Si on note F_1, F_2 , et G les transformées de Fourier de f_1, f_2 , et g , on a :

$$g[x, y] = f_1[x, y] \star f_2[x, y] \Rightarrow G[u, v] = F_1[u, v].F_2[u, v] \quad (2.6)$$

$$g[x, y] = f_1[x, y].f_2[x, y] \Rightarrow G[u, v] = F_1[u, v] \star F_2[u, v] \quad (2.7)$$

On peut donc également justifier l'utilisation du filtrage linéaire pour réduire le bruit en raisonnant directement dans le domaine spatial : la combinaison linéaire réalisée par la convolution réduit la variance du bruit additif.

Dans certains cas, la perturbation liée au bruit peut être non plus additive, mais multiplicative : $g(x) = f(x).b(x)$. Dans ce cas, on ne pourra pas tenter de retrouver f par filtrage linéaire. Néanmoins, on se ramènera au cas précédent en passant au logarithme.

On rencontre également des perturbations sur les images où la valeur observée $g(x)$ ne dépend pas de la valeur initiale $f(x)$, mais remplace (avec une certaine probabilité) $f(x)$ par une valeur quelconque. Il s'agit du bruit *impulsionnel*. Dans ce cas, le filtrage linéaire n'est plus adaptée, et on a plutôt recours à d'autres opérations de voisinage qu'on désigne sous le terme de *filtres non linéaires*.

Enfin, le dernier type de perturbation que nous considérerons dans ce chapitre est la dégradation de type convolutive où le défaut apparaît sous la forme d'un noyau de convolution : $g(x) = f(x) \star d(x)$. Le principe de base consiste à inverser le défaut supposé connu, ce qui se traduit par une division dans le domaine des fréquences : $F(u) = G(u).D(u)^{-1}$. Mais cette technique n'est pas exploitable directement en pratique, elle doit être conditionnée en fonction d'une estimation du bruit additif qui s'ajoute au défaut convolutif.

La partie 2.2 traite des filtres *lisseurs*, c'est-à-dire des opérateurs chargés d'améliorer ou de simplifier les images en les rendant plus régulières. Par différence (domaine spatiale) ou par complémentation (domaine fréquentiel), on obtiendra les filtres *différenciateurs* (Partie 2.3), dont la vocation est au contraire de faire ressortir les régions de contraste dans les images, ou encore d'estimer des grandeurs différentielles, utilisées par de nombreux opérateurs d'analyse d'images. Enfin, la partie 2.4 traite de la *restauration* d'images, c'est-à-dire du traitement des défauts convolutifs.

2.2 Filtres lisseurs

Les filtres de lissage sont des opérateurs qui éliminent des éléments perturbateurs pour améliorer la visualisation des images numériques, ou bien qui suppriment des structures non significatives pour simplifier leur contenu.

2.2.1 Filtrage dans le domaine fréquentiel

En identifiant les structures parasites ou inutiles à des composantes fréquentielles des images numériques, le filtrage peut se faire par sélection des dites composantes dans le domaine fréquentiel : Ainsi le filtrage idéal consiste-t-il à multiplier la transformée de Fourier $F[u, v]$ par une fonction qui vaut 1 si la composante fréquentielle est conservée, 0 si elle est supprimée. Par exemple, le *filtre passe-bas* idéal est défini comme suit :

$$F_{lp}^{(u_M, v_M)}[u, v] = F[u, v] \text{ si } u \leq u_M \text{ et } v \leq v_M \quad (2.8)$$

$$= 0 \text{ sinon} \quad (2.9)$$

De même le *filtre coupe-bande* idéal est défini par :

$$F_{bc}^{(u_m, v_m, u_M, v_M)}[u, v] = 0 \text{ si } u_m < u < u_M \text{ et } v_m < v < v_M \quad (2.10)$$

$$= F[u, v] \text{ sinon} \quad (2.11)$$

La figure 2.1 illustre l'application de ces deux filtres sur une image.

Notons que dans le cas du filtre passe-bas comme du coupe-bande, la composante de fréquence nulle $F(0, 0)$ est conservée. Or $F(0, 0) = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} f[x, y]$.

La somme des niveaux de gris sur toute l'image est donc conservée par ces 2 filtres.

2.2.2 Filtrage par convolution

Comme on l'a vu en introduction de ce chapitre, le filtrage linéaire peut également se calculer directement dans le domaine spatial, par une convolution. Dans le domaine fréquentiel, nous avons assimilé les opérateurs de lissage aux filtres qui laissent la composante de fréquence nulle $F(0, 0)$ inchangée. Un opérateur de convolution q appartenant à la famille des filtres lisseurs aura donc une composante de fréquence nulle $Q(0, 0)$, égale à 1, ce qui signifie que le noyau de convolution q doit être de somme unité pour que l'image lissée conserve la somme des niveaux de gris de l'image originale.

Il existe plusieurs noyaux de convolution lisseurs, correspondant à des fonctions h symétriques, de somme égale à 1. Chacune de ces fonctions est en général associée à un paramètre, qui définit sa *portée spatiale*, et par suite sa «capacité de lissage». Pour pouvoir comparer deux noyaux lisseurs associés à différentes fonctions, il faut pouvoir leur associer à chacun un paramètre qui leur confère une même capacité de lissage. C'est l'objet du *coefficient d'atténuation* présenté ci-après.

Soit $b[x, y]$ un bruit, correspondant à une perturbation aléatoire de moyenne nulle. Supposons qu'on filtre b par un noyau de convolution q . On s'intéresse à présent à la variance de ce bruit filtré. Cette variance peut être mesurée dans le domaine fréquentiel en utilisant $\hat{B}^2[u, v]$, qui est la densité spectrale de puissance de b (Cela correspond au carré du module de la transformée de Fourier pour un signal déterministe, mais b étant aléatoire, \hat{B}^2 ne peut pas être mesuré, mais seulement estimé ou modélisé *a priori*). Donc si l'on note Q la transformée de Fourier de q :

$$\text{var}(b \star q) = \frac{1}{wh} \sum_{u=0}^{w-1} \sum_{v=0}^{h-1} \|Q[u, v]\|^2 \cdot \hat{B}^2[u, v] \quad (2.12)$$

Si l'on fait l'hypothèse que le bruit b n'est pas corrélé d'un pixel à l'autre, sa fonction d'autocorrélation définie par :

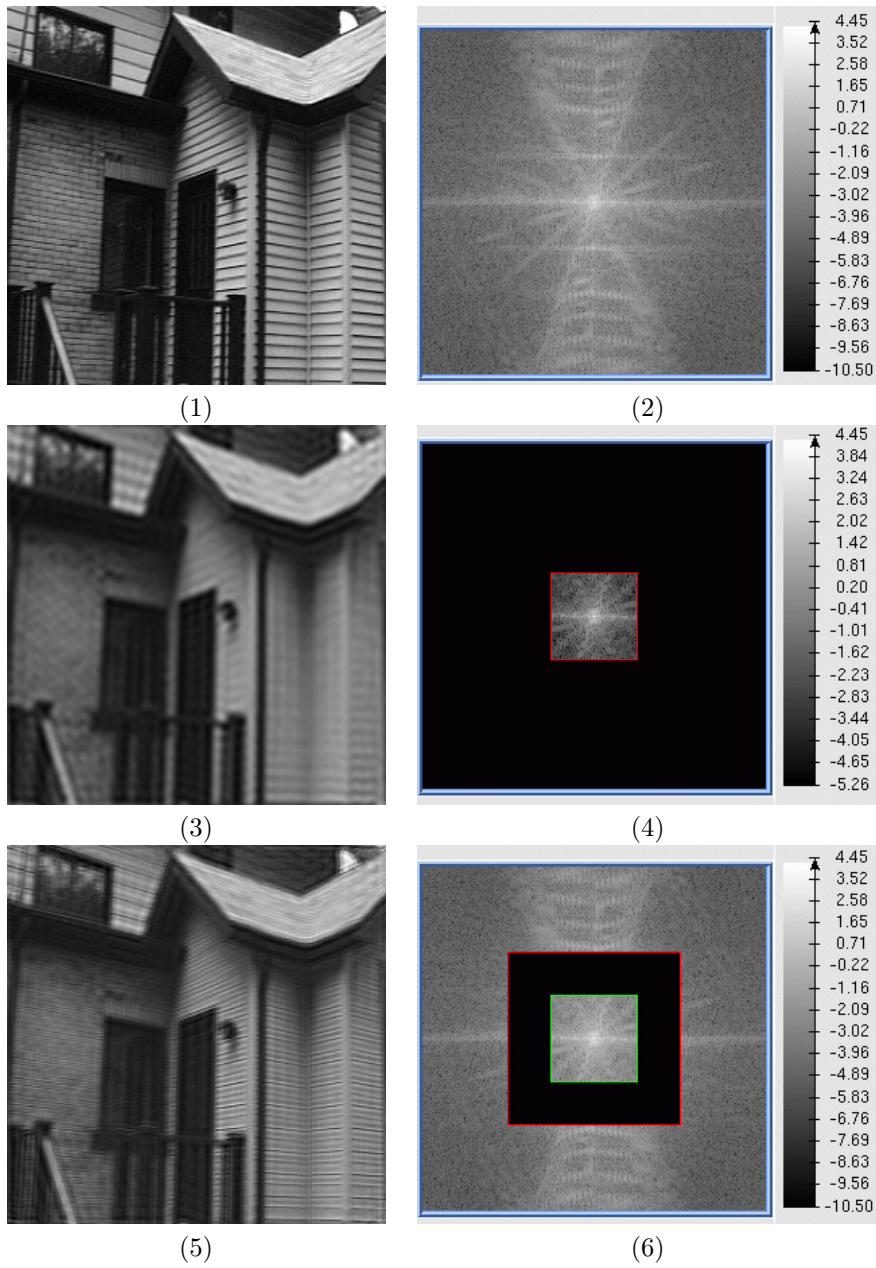


FIGURE 2.1 – Filtres de lissage idéaux. (1) et (2) : une image (256×256) et son spectre d'amplitude. (3) et (4) : filtre passe-bas ($u_M = v_M = 32$). (5) et (6) : filtre coupe-bande ($u_m = v_m = 32, u_M = v_M = 64$)

$$\chi[\tau, \zeta] = \sum \sum b[x, y].b[x + \tau, y + \zeta] \quad (2.13)$$

peut être considérée comme nulle sauf en $(0, 0)$. Or la densité spectrale de puissance correspondant à la transformée de Fourier de la fonction d'autocorrélation, il en résulte que la densité spectrale de puissance d'un tel bruit est constante sur l'ensemble des composantes fréquentielles (bruit «blanc»). Dans ce cas, l'équation 2.12 devient :

$$\text{var}(b \star q) = K_b^2 \cdot \frac{1}{wh} \sum_{u=0}^{w-1} \sum_{v=0}^{h-1} \|Q[u, v]\|^2 \quad (2.14)$$

Où K_b^2 est la densité spectrale (constante) de b . D'après le théorème de Parseval, cette quantité peut également être mesurée dans le domaine spatial :

$$\text{var}(b \star q) = K_b^2 \cdot \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} (q[x, y])^2 \quad (2.15)$$

Le coefficient d'atténuation δ_q d'un noyau de convolution q désigne donc le rapport entre la densité spectrale d'un bruit blanc et la variance de ce même bruit filtré par q , soit :

$$\delta_q = wh \cdot \left(\sum_{u=0}^{w-1} \sum_{v=0}^{h-1} \|Q[u, v]\|^2 \right)^{-1} = \left(\sum_{x=0}^{w-1} \sum_{y=0}^{h-1} (q[x, y])^2 \right)^{-1} \quad (2.16)$$

On peut maintenant présenter et comparer les noyaux de convolutions lisseurs les plus fréquemment rencontrés. S'agissant d'images, ce sont des noyaux de convolution 2d, mais qui sont en raison de leur symétrie, *séparable*. Plus précisément, si une fonction 2d $q[x, y]$ peut s'écrire comme le produit de 2 fonctions 1d : $q[x, y] = q_h[x].q_v[y]$, dans ce cas la convolution par q peut se calculer par la composition de 2 convolutions, l'une verticale, l'autre horizontale :

$$q[x, y] = q_h[x].q_v[y] \Rightarrow f \star q = (f \star \tilde{q}_h) \star \tilde{q}_v \quad (2.17)$$

Avec $\tilde{q}_h[x, y] = q_h[x]$ et $\tilde{q}_v[x, y] = q_v[y]$. Ces noyaux de convolution sont donc entièrement définis par leur expression 1d. Les tableaux 2.1 à 2.4 présentent les principaux noyaux de convolution correspondant aux filtres lisseurs dans le domaine spatial, et leurs propriétés.

Le filtre moyenneur correspond donc à la fonction porte dans le domaine spatial. Sa transformée de Fourier, correspondant aux coefficients de pondération des composantes fréquentielles appliqués par le filtre dans le domaine fréquentiel, est une fonction sinus cardinal. Réciproquement, le filtre passe bas idéal présenté précédemment et correspondant dans le domaine de Fourier à une fonction porte, s'exprime dans le domaine spatial par une convolution par un sinus cardinal.

Le filtre gaussien utilise un paramètre σ qu'on dit écart-type, car il correspond à l'écart-type des valeurs de la fonction gaussienne considérée comme une distribution. Ce filtre est très utilisé en traitement du signal et des images, pour de nombreuses raisons. La première, liée au théorème centrale limite, réside dans le fait que l'application itérée de moyennes locales équivaut asymptotiquement

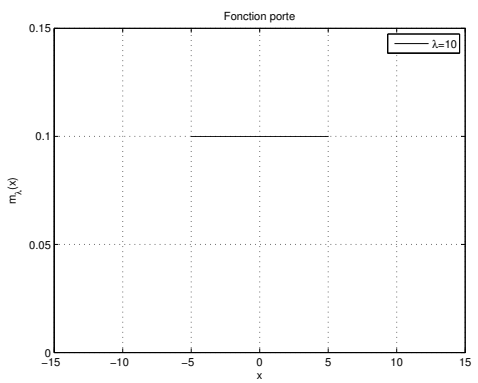
Filtre Moyenneur	
Expression (1d) :	$m_\lambda(x) = \begin{cases} \frac{1}{\lambda} & \text{si } x \in [-\lambda/2, +\lambda/2] \\ 0 & \text{sinon} \end{cases}$
	
Transf. de Fourier (1d) :	$M_\lambda(u) = \frac{\sin(\lambda\pi u)}{\lambda\pi u}$
Expression (2d) :	$m_\lambda(x, y) = \begin{cases} \frac{1}{\lambda^2} & \text{si } (x, y) \in [-\lambda/2, +\lambda/2]^2 \\ 0 & \text{sinon} \end{cases}$
Coeff. d'atténuation (2d)	$\delta_m = \lambda^2$

TABLE 2.1 – Le filtre moyenneur et ses propriétés

à une convolution par une gaussienne. Ensuite, ce qui est particulièrement important en traitement d'images, le noyau gaussien 2d est *isotrope*, c'est-à-dire qu'il commute avec une rotation de l'image. Troisième raison, la transformée de Fourier d'une gaussienne d'écart-type σ est une autre gaussienne, d'écart-type $1/\sigma$. Le filtre gaussien réalise ainsi une sorte d'équilibre entre les répartitions spatiale et fréquentielle, ce qui se traduit par un « meilleur compromis » entre la préservation des structures locales et la réduction du bruit.

Le filtre exponentiel utilise un paramètre $\gamma > 0$, qui détermine sa dispersion spatiale (plus γ est grand, moins le filtre est dispersé spatialement). Par rapport aux autres filtres, le filtre exponentiel préserve mieux les transitions (contours), au prix d'une réduction moins visible du bruit.

La figure 2.2 montre le résultat de l'application des 4 filtres présentés dans cette section sur la même image synthétique bruitée. Les 4 paramètres ont été fixés pour obtenir approximativement le même coefficient d'atténuation. Noter les différences de comportement vis-à-vis des transitions. Remarquer également les ondulations qui apparaissent dans le cas du filtre idéal, correspondant aux fréquences harmoniques des transitions en escalier (frontières des objets) si l'on se place dans le cadre fréquentiel, ou aux lobes secondaires du sinus cardinal si

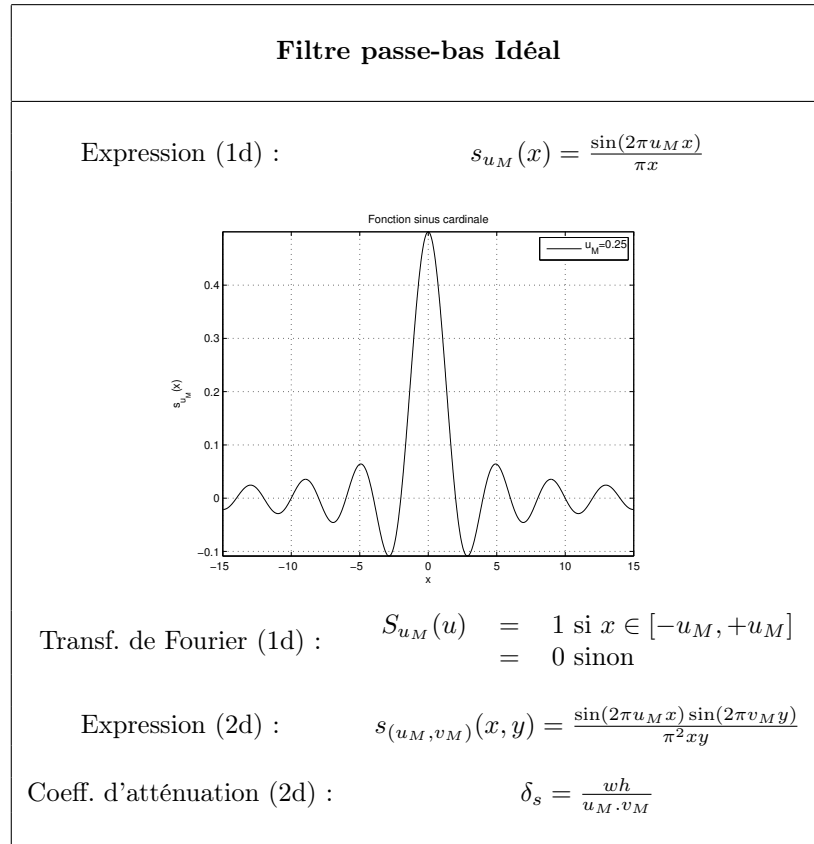


TABLE 2.2 – Le filtre passe-bas idéal dans le domaine spatial et ses propriétés.

l'on se place dans le cadre spatial.

2.2.3 Filtrage non linéaire

Le filtrage linéaire montre des limites dans deux aspects particuliers de l'amélioration des images numériques :

- *Le bruit impulsionnel* : Lorsque l'image fait apparaître des valeurs parasites aberrantes qui ne dépendent pas des valeurs initiales de l'image, le filtrage linéaire diffuse ces valeurs aberrantes sur les pixels voisins.
- *Le respect des frontières* : Le filtrage linéaire lisse les transitions abruptes, rendant ainsi les images floues.

On désigne par filtre non linéaire un opérateur qui remplace la valeur de chaque pixel par une combinaison non linéaire des valeurs de ses pixels voisins. L'objectif des filtres lisseurs non linéaires est d'apporter une solution aux limites exprimées ci-dessus, par l'utilisation d'une (ou de plusieurs) des techniques suivantes :

- *statistiques d'ordre* : Opérateurs basés sur un classement des valeurs des pixels voisins (minimum, maximum, médiane, ...)
- *moyennes robustes* : Calculs de moyennes locales - éventuellement pon-

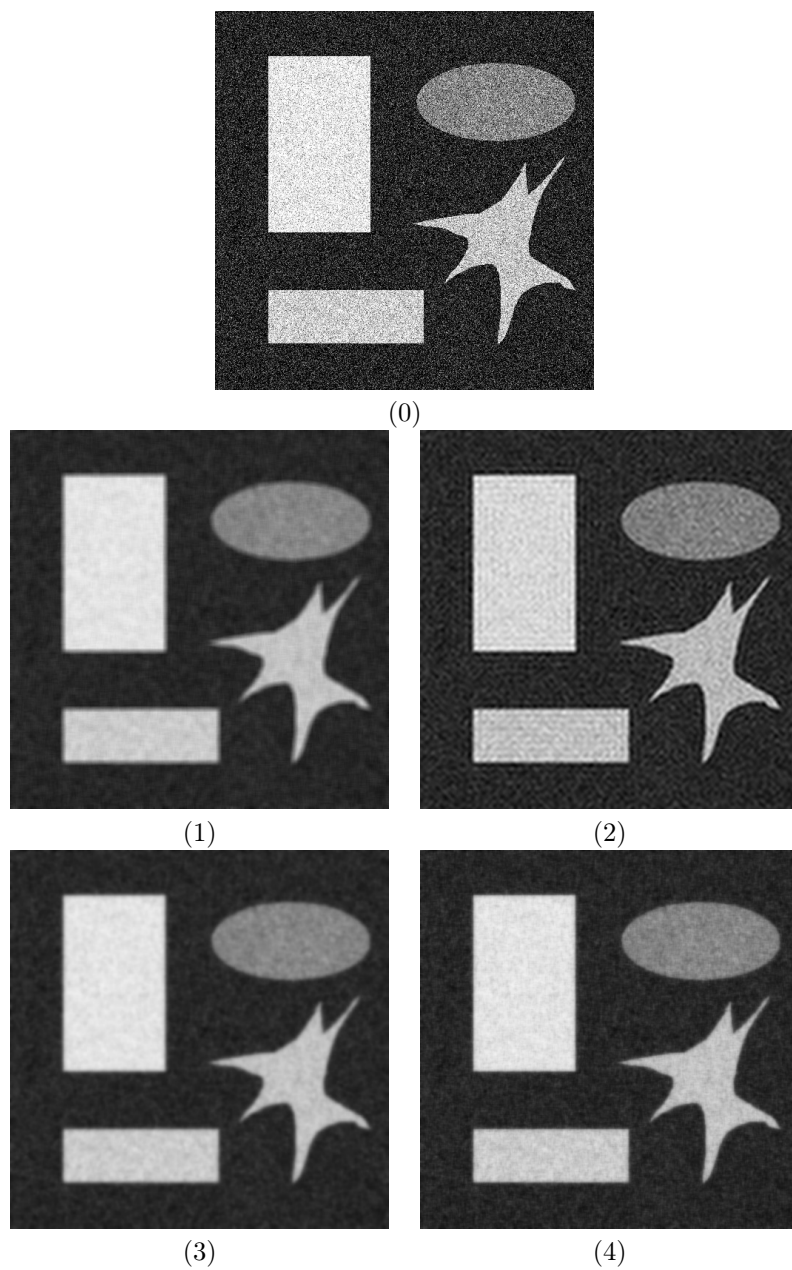


FIGURE 2.2 – Application des filtres de lissage linéaires sur une image synthétique bruitée. (0) Image originale 512×512 . (1) filtre moyenneur ($\lambda = 9$). (2) filtre passe-bas idéal ($u_M = v_M = 57$). (3) filtre gaussien ($\sigma = 2, 5$). (3) filtre exponentiel ($\gamma = 0, 44$).

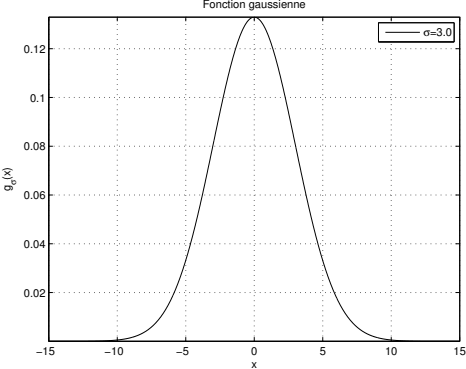
Filtre gaussien	
Expression (1d) :	$g_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$
	
Transf. de Fourier (1d) :	$G_\sigma(u) = \sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2 u^2}$
Expression (2d) :	$g_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$
Coeff. d'atténuation (2d) :	$\delta_g = 4\pi\sigma^2$

TABLE 2.3 – Le filtre gaussien et ses propriétés.

dérées - en excluant du calcul les valeurs marginales.

- *anisotropie* : Opérateurs où le support de calcul du filtre effectivement calculé varie localement en fonction du contenu de l'image.

Il existe un grand nombre de filtres non linéaires exploitant une ou plusieurs de ces propriétés. Nous présentons d'abord les filtres d'ordre, puis le filtre de Nagao, qui présente à la fois un intérêt historique et qui combine les différentes techniques évoquées plus haut. Enfin nous présentons les moyennes non locales, qui font partie des méthodes de référence pour le débruitage d'images.

Filtres d'ordre

Soit B un sous-ensemble fini de \mathbb{Z}^2 , contenant l'origine $(0, 0)$. Soit $N = \text{Card}(B)$. Soit $\mathcal{N}_B(x, y) = \{(x, y) + b; b \in B\}$ l'ensemble des voisins de (x, y) au sens de B . Soit $V_B^f(x, y) \in \mathbb{R}^N$ un vecteur contenant les N valeurs prises par f dans $\mathcal{N}_B(x, y)$, classées par ordre croissant :

$$\forall (x', y') \in \mathcal{N}_B(x, y), \exists i \in \{1, N\} : V_B^f(x, y)(i) = f(x', y') \quad (2.18)$$

$$\forall (i, j) \in \{1, N\}^2, i \leq j \Rightarrow V_B^f(x, y)(i) \leq V_B^f(x, y)(j) \quad (2.19)$$

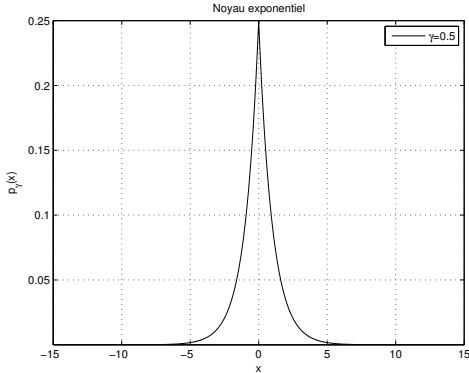
Filtre exponentiel	
Expression (1d) :	$p_\gamma(x) = \frac{\gamma}{2} e^{-\gamma x }$
	
Transf. de Fourier (1d) :	$P_\gamma(u) = \frac{\gamma^2}{\gamma^2 + 4\pi^2 u^2}$
Expression (2d) :	$p_\gamma(x, y) = \frac{\gamma^2}{4} e^{-\gamma(x + y)}$
Coeff. d'atténuation (2d) :	$\delta_p = \frac{16}{\gamma^2}$

TABLE 2.4 – Le filtre exponentiel et ses propriétés.

Le filtre d'ordre μ_B^k , avec $k \in \{1, N\}$, est défini par :

$$\mu_B^k(f)[x, y] = V_B^f(x, y)(k) \quad (2.20)$$

Pour $k = 1$ on parle de filtre min ou *érosion morphologique*. Pour $k = N$ on parle de filtre max ou *dilatation morphologique*. On se reportera au cours de morphologie mathématique pour une étude approfondie de ces opérateurs et leurs applications. Pour $k = N/2$ on parle de filtre médian. Comme la médiane est une statistique robuste vis-à-vis des valeurs marginales, le filtre médian est particulièrement adapté à la suppression du bruit impulsif, comme le montre la figure 2.3.

Filtre de Nagao

Le filtre de Nagao [35] combine statistique linéaire et anisotropie, de la façon suivante : Pour chaque pixel (x, y) , on examine l'ensemble des voisins contenu dans un carré 5×5 centré sur (x, y) : $\mathcal{N}(x, y) = \{(x, y) + b; b \in \{-2, +2\}^2\}$. On considère ensuite 9 ensembles $\mathcal{N}_i(x, y) \subset \mathcal{N}(x, y)$, $i \in \{1, 9\}$, comme représenté sur la figure 2.4.

Pour chaque $\mathcal{N}_i(x, y)$, on calcule la moyenne $m_i(x, y)$ et la variance $v_i(x, y)$. Le résultat du filtre de Nagao consiste à remplacer la valeur de chaque pixel

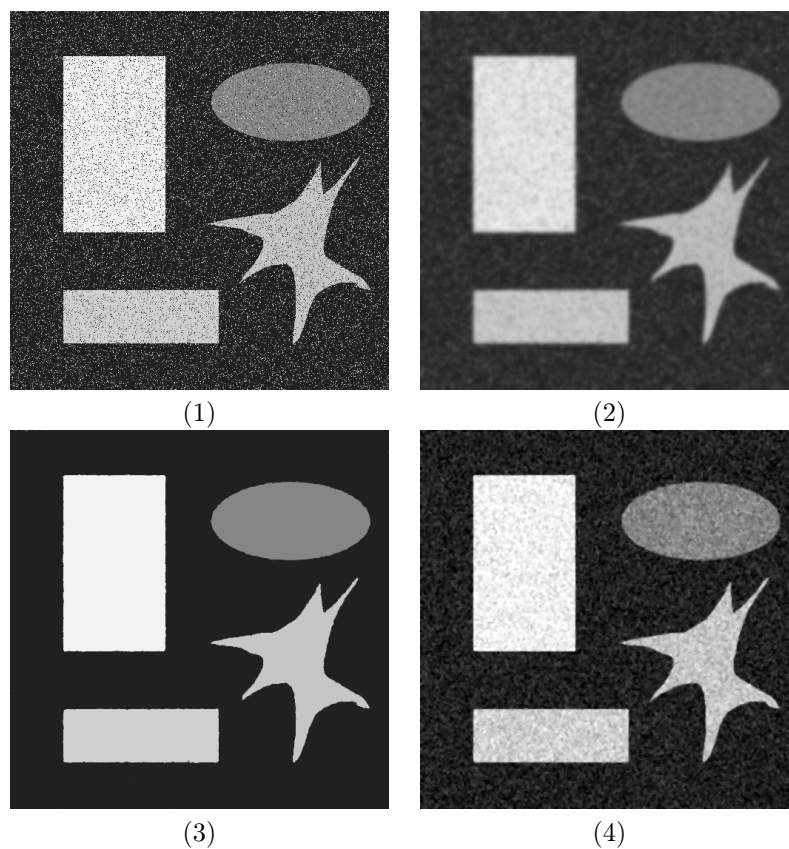


FIGURE 2.3 – Comportement du filtre médian. (1) Image original : bruit impulsionnel (15% des valeurs modifiées arbitrairement). (2) Après filtrage gaussien ($\sigma = 2,5$). (3) Après filtrage médian ($\text{Card}(B) = 21$). (4) Même filtre médian appliqué sur l'image avec bruit additif (voir Figure 2.2).

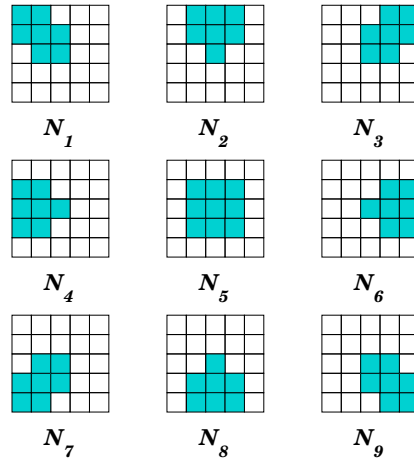


FIGURE 2.4 – Les différents sous-ensembles du voisinage 5×5 utilisés pour le calcul du filtre de Nagao.

par la moyenne $m_i(x, y)$ du domaine $\mathcal{N}_i(x, y)$ qui présente la variance $v_i(x, y)$ minimum. Le comportement du filtre de Nagao, comparé à un filtre linéaire (exponentiel), et à un filtre médian, est illustré Figure 2.5.

Moyennes non locales

Comme on l'a vu précédemment, une image naturelle contient toujours des discontinuités qui engendrent des hautes fréquences spatiales. Il en résulte que ni le filtrage dans le domaine fréquentiel, ni une convolution isotrope ne peut débruiter correctement une image.

Le meilleur moyen de débruiter une image est d'obtenir le plus grand nombre possible d'observations différentes de chaque pixel et de faire des statistiques sur ces observations. Une moyenne temporelle sur une vidéo de scène statique serait donc idéale, mais que faire lorsqu'on n'a qu'une seule image ?

Le principe des moyennes non locales, introduit par Buades, Coll et Morel [5], est de calculer, pour chaque pixel, la moyenne des pixels qui lui ressemblent le plus. Ces opérateurs peuvent s'exprimer dans le cadre des filtres linéaires généralisés :

$$\hat{f}(\mathbf{p}) = \frac{1}{Z(\mathbf{p})} \sum_{\mathbf{q} \in \mathcal{N}(\mathbf{p})} f(\mathbf{q}) \omega(\mathbf{p}, \mathbf{q}) \quad (2.21)$$

où \mathbf{p} et \mathbf{q} sont des pixels de \mathbb{Z}^2 , $\mathcal{N}(\mathbf{p}) \subset \mathbb{Z}^2$ est un voisinage de \mathbf{p} , $\omega(\mathbf{p}, \mathbf{q}) \in \mathbb{R}$ est le poids attaché à \mathbf{q} relativement à \mathbf{p} , enfin Z est la fonction de normalisation, définie par :

$$Z(\mathbf{p}) = \sum_{\mathbf{q} \in \mathcal{N}(\mathbf{p})} \omega(\mathbf{p}, \mathbf{q}) \quad (2.22)$$

Pour une convolution classique qui réalise un filtre linéaire isotrope, on aura $\omega(\mathbf{p}, \mathbf{q}) = g(\|\mathbf{p} - \mathbf{q}\|)$ où g est une fonction décroissante (comme les fonctions

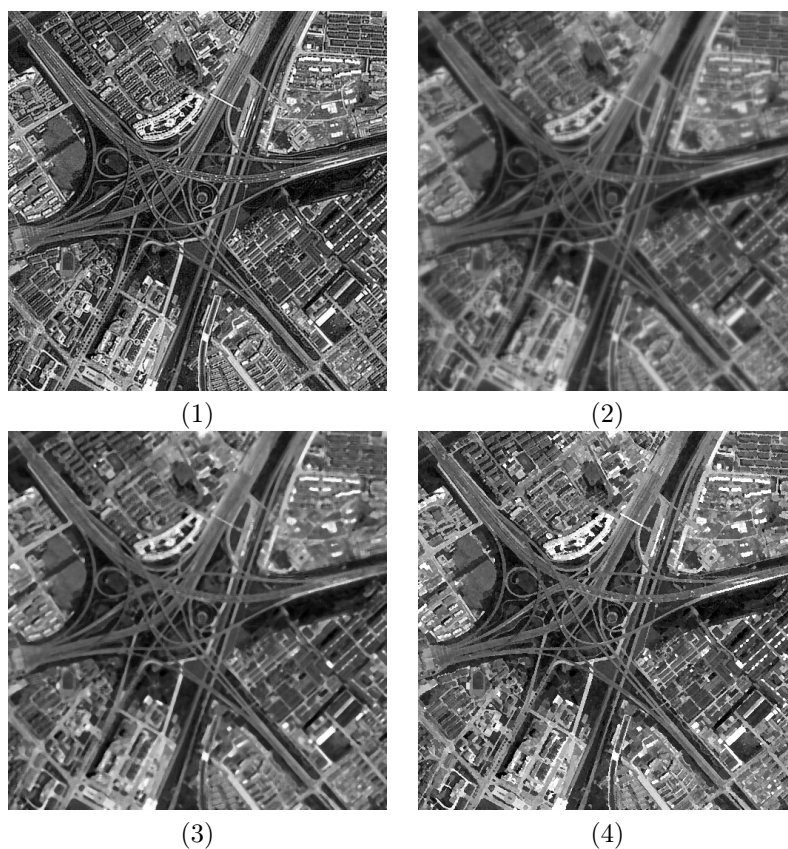


FIGURE 2.5 – Comportement du filtre de Nagao. (1) Image original : Shanghai (détail d'une image SPOT 5 - © CNES 2002). (2) Après filtrage exponentiel ($\gamma = 0,44$). (3) Après filtrage médian ($\text{Card}(B) = 25$). (4) Après filtrage de Nagao.

gaussienne et exponentielle vue précédemment). Dans le cas des moyennes non locales, $\omega(\mathbf{p}, \mathbf{q})$ ne dépend plus de $\|\mathbf{p} - \mathbf{q}\|$, mais de la ressemblance locale entre les pixels \mathbf{p} et \mathbf{q} vis-à-vis de l'image f :

$$\omega(\mathbf{p}, \mathbf{q}) = e^{-\frac{d_f(\mathbf{p}, \mathbf{q})}{h^2}} \quad (2.23)$$

où h est un paramètre de dégénérescence, qui dépend de l'importance du bruit à retirer, et $d_f(\mathbf{p}, \mathbf{q})$, qui mesure la dissimilarité entre \mathbf{p} et \mathbf{q} est généralement définie comme suit :

$$d_f(\mathbf{p}, \mathbf{q}) = \sum_{\mathbf{z} \in \mathcal{W}(\mathbf{O})} k(\|\mathbf{z}\|) (f(\mathbf{p} + \mathbf{z}) - f(\mathbf{q} + \mathbf{z}))^2 \quad (2.24)$$

où $\mathcal{W}(\mathbf{O})$ est un voisinage de \mathbf{O} l'origine de \mathbb{Z}^2 , et k une fonction décroissante réelle (par exemple une gaussienne).

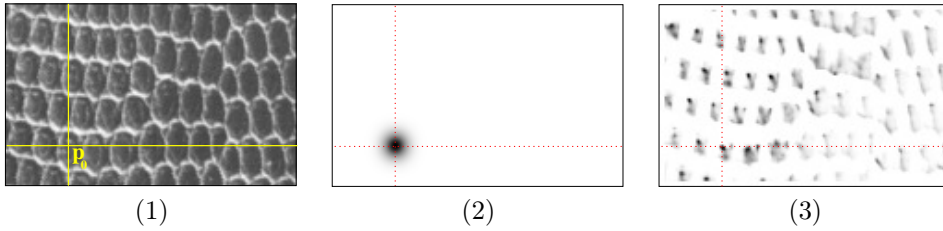


FIGURE 2.6 – La fonction de pondération associée à un pixel \mathbf{p}_0 (1), dans le cas d'une convolution gaussienne (2), et d'une moyenne non locale (3). Ici le niveau de noir est proportionnel au poids du pixel.

La figure 2.6 montre la fonction de pondération associée à un pixel particulier \mathbf{p}_0 d'une texture naturelle. Cette fonction est définie par $\pi_{\mathbf{p}_0}(\mathbf{q}) = \omega(\mathbf{p}_0, \mathbf{q})$. On peut observer que dans le cas de la convolution gaussienne, il est facile de borner $\mathcal{N}(\mathbf{p})$ aux pixels \mathbf{q} tels que le poids $\omega(\mathbf{p}_0, \mathbf{q})$ soit significatif. Au contraire, pour les moyennes non locales, $\mathcal{N}(\mathbf{p})$ n'a aucune raison d'être borné, d'où une complexité de calcul potentiellement très grande.

On peut néanmoins calculer des versions satisfaisantes des moyennes non locales en limitant le voisinage $\mathcal{N}(\mathbf{p})$ à des tailles raisonnables. Par exemple, les résultats de la figure 2.7 ont été obtenus en limitant $\mathcal{N}(\mathbf{p})$ à un voisinage de 17×17 pixels.

2.3 Filtres différentiateurs

2.3.1 Introduction

Les variations locales d'intensité constituent une source primordiale d'information, utilisée dans presque toutes les fonctions d'analyse d'images. De manière générale, le calcul différentiel permet d'étudier le comportement local d'une fonction, sous la forme d'une fonction plus simple (polynômiale) construite à partir des dérivées (Formule de Taylor).

L'image f étant assimilée à une fonction 2d continue et différentiable, ses dérivées premières s'expriment par une fonction vectorielle, le *gradient* :



FIGURE 2.7 – Application des moyennes non locales au débruitage d'image.

$$\nabla f(x_0, y_0) = \left(\frac{\partial f}{\partial x}(x_0, y_0), \frac{\partial f}{\partial y}(x_0, y_0) \right) \quad (2.25)$$

Le vecteur gradient permet d'exprimer algébriquement la règle de dérivation des fonctions composées. Si u est un vecteur unitaire du plan, on calcule la dérivée de f selon u comme suit :

$$\frac{\partial f}{\partial u} = u^\top \cdot \nabla f \quad (2.26)$$

Par suite, la direction et le sens du vecteur gradient correspond à la direction de plus forte pente, qui est orthogonale à la direction de l'isophote (courbe de niveau de gris constant). La norme du gradient, quant à elle, correspond à la valeur de la pente, c'est-à-dire au contraste (voir Figure 2.8).

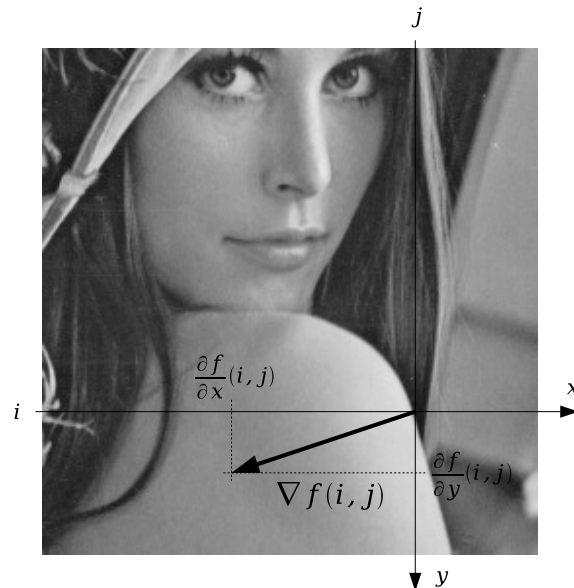


FIGURE 2.8 – Le vecteur gradient.

On s'intéressera également aux dérivées secondes, qui s'expriment par une fonction matricielle, le *hessien* :

$$H_f(x_0, y_0) = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2}(x_0, y_0) & \frac{\partial^2 f}{\partial x \partial y}(x_0, y_0) \\ \frac{\partial^2 f}{\partial x \partial y}(x_0, y_0) & \frac{\partial^2 f}{\partial y^2}(x_0, y_0) \end{pmatrix} \quad (2.27)$$

A l'instar du vecteur gradient, la matrice hessienne permet de calculer les dérivées secondes dans toutes les directions. Si u et v sont deux vecteurs unitaires du plan, on a :

$$\frac{\partial^2 f}{\partial u \partial v} = u^\top \cdot H_f \cdot v \quad (2.28)$$

Géométriquement, la hessienne représente les courbures principales de la fonction f , dont les directions principales et intensités associées sont données

respectivement par les vecteurs et les valeurs propres de la matrice. La somme des valeurs propres, i.e. la trace de la hessienne, est le *laplacien* :

$$\Delta f(x_0, y_0) = \frac{\partial^2 f}{\partial x^2}(x_0, y_0) + \frac{\partial^2 f}{\partial y^2}(x_0, y_0) \quad (2.29)$$

Dans le cadre des images numériques, le problème est : comment estimer ces grandeurs différentielles. Comme dans la section précédente, on considèrera d'abord le cas du filtrage dans le domaine fréquentiel, en suivant une logique symétrique à celle des filtres lisseurs. Puis on s'intéressera à l'estimation des dérivées par des opérateurs de convolution.

2.3.2 Filtrage dans le domaine fréquentiel

Le *filtre passe-haut* idéal est défini comme suit :

$$F_{hp}^{(u_m, v_m)}[u, v] = F[u, v] \text{ si } u \geq u_m \text{ et } v \geq v_m \quad (2.30)$$

$$= 0 \text{ sinon} \quad (2.31)$$

De même le *filtre passe-bande* idéal est défini par :

$$F_{bp}^{(u_m, v_m, u_M, v_M)}[u, v] = F[u, v] \text{ si } u_m < u < u_M \text{ et } v_m < v < v_M \quad (2.32)$$

$$= 0 \text{ sinon} \quad (2.33)$$

La figure 2.9 illustre l'application de ces deux filtres sur une image.

Dans les deux cas, on notera que la composante de fréquence nulle $F(0, 0)$ est annulée. Les images obtenues sont donc de somme nulle dans le domaine spatiale, et comportent par conséquent des valeurs négatives. On interprétera donc les images (3) et (5) de la figure 2.9 en considérant que le niveau de gris moyen représente la valeur zéro, et les niveaux de gris sombres (resp. claires) les valeurs négatives (resp. positives).

2.3.3 Estimation des grandeurs différentielles

Les approximations les plus simples des dérivées premières directionnelles s'obtiennent par simples différences finies, par exemple :

$$f_x[i, j] = f[i, j] - f[i - 1, j] \quad (2.34)$$

$$f_y[i, j] = f[i, j] - f[i, j - 1] \quad (2.35)$$

qui correspondent à des convolutions avec les noyaux $\begin{pmatrix} -1 & \mathbf{1} \end{pmatrix}$ pour f_x , qui est l'approximation de $\frac{\partial f}{\partial x}$, et $\begin{pmatrix} -1 \\ \mathbf{1} \end{pmatrix}$ pour f_y , qui est l'approximation de $\frac{\partial f}{\partial y}$ (le nombre en gras indique l'origine du noyau de convolution).

Notons qu'on a fait ici un choix arbitraire quant au choix de l'origine dans l'opérateur de convolution, qui se traduit par un «déalage» d'un demi-pixel vers la gauche (resp. vers le haut) dans l'estimation de la dérivée horizontale (resp. verticale).

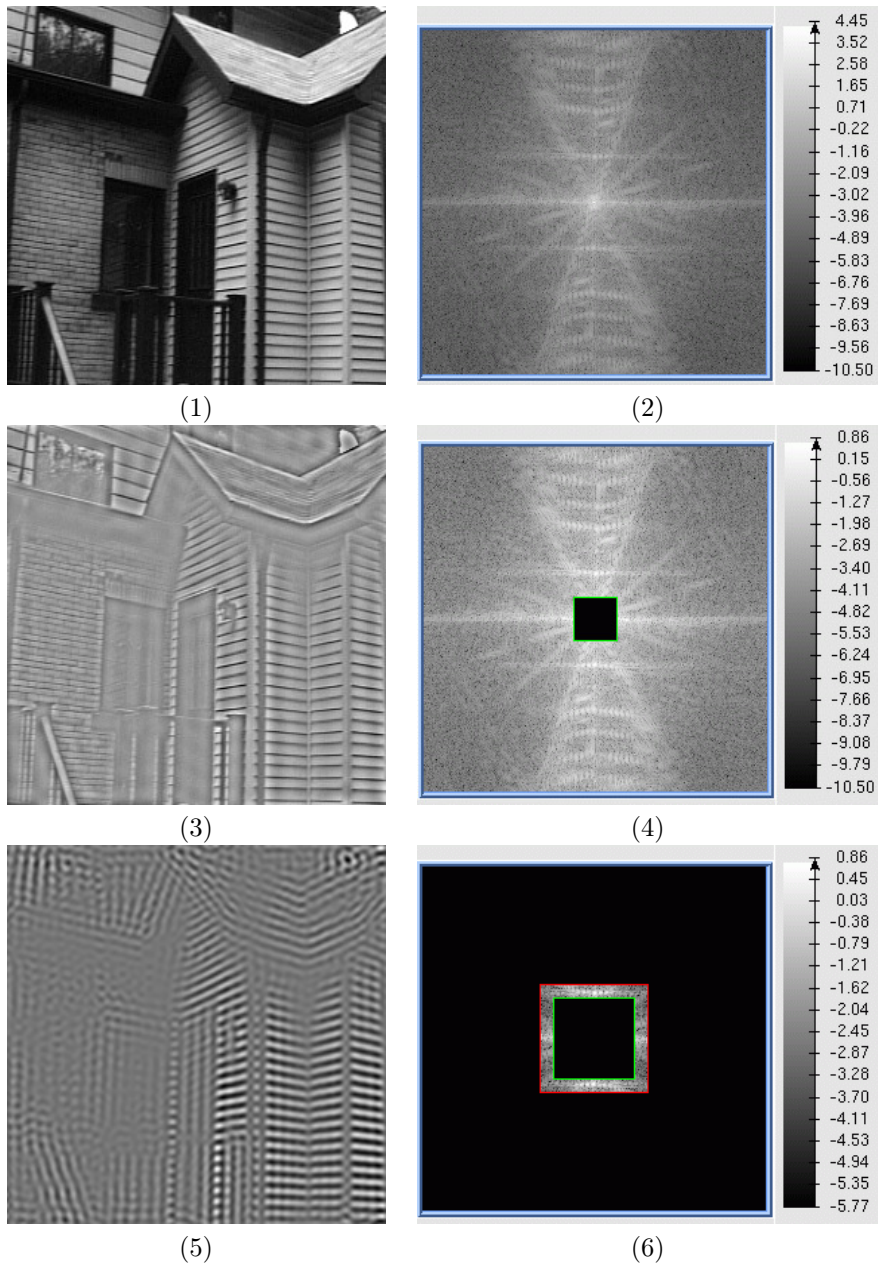


FIGURE 2.9 – Filtres différenciateurs idéaux. (1) et (2) : une image (256×256) et son spectre d'amplitude. (3) et (4) : filtre passe-haut ($u_m = v_m = 16$). (5) et (6) : filtre passe-bande ($u_m = v_m = 30, u_M = v_M = 40$)

Pour cette raison, on utilise plus souvent les opérateurs de convolution $\begin{pmatrix} -1 & \mathbf{0} & 1 \end{pmatrix}$ pour la dérivée horizontale, et $\begin{pmatrix} -1 \\ \mathbf{0} \\ 1 \end{pmatrix}$, pour la dérivée verticale, qui produiront des frontières plus épaisses, mais bien centrées.

Les opérateurs de dérivation sont par nature très sensibles aux irrégularités (et au bruit) dans les images numériques, ce qui est très problématique lorsqu'on veut justement estimer une dérivée pour connaître le comportement de la fonction image au voisinage d'un point. On combine donc en général l'opération de dérivation dans une direction par une opération de lissage dans la direction orthogonale, par exemple en utilisant le noyau $\begin{pmatrix} 1 & \mathbf{2} & 1 \end{pmatrix}$, ou sa transposée. Les opérateurs de convolution utilisant un masque 3×3 les plus utilisés sont ainsi les noyaux de convolution suivants, dits masques de Sobel :

$$h_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & \mathbf{0} & 2 \\ -1 & 0 & 1 \end{pmatrix}; h_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & \mathbf{0} & 0 \\ 1 & 2 & 1 \end{pmatrix} \quad (2.36)$$

On a donc :

$$f_x[i, j] = (f * h_x)[i, j] \quad (2.37)$$

$$f_y[i, j] = (f * h_y)[i, j] \quad (2.38)$$

Ces fonctions scalaires fournissent les coordonnées du vecteur gradient, dont on peut calculer la norme (euclidienne, \mathcal{L}_1 , \mathcal{L}_∞) :

$$\|\nabla f[i, j]\|_2 = \sqrt{f_x[i, j]^2 + f_y[i, j]^2} \quad (2.39)$$

$$\|\nabla f[i, j]\|_1 = |f_x[i, j]| + |f_y[i, j]| \quad (2.40)$$

$$\|\nabla f[i, j]\|_\infty = \max(|f_x[i, j]|, |f_y[i, j]|) \quad (2.41)$$

ainsi que l'orientation :

$$\arg(\nabla f[i, j]) = \arctan\left(\frac{f_y[i, j]}{f_x[i, j]}\right) \quad (2.42)$$

Concernant les dérivées secondes, la composition du noyau de convolution $\begin{pmatrix} -1 & \mathbf{1} \end{pmatrix}$ (resp. sa transposée), avec lui-même donne $\begin{pmatrix} \mathbf{1} & -2 & 1 \end{pmatrix}$ (resp. sa transposée). Mais comme on a enchaîné deux décalages d'un demi-pixel vers la gauche (resp. vers le haut), on recentre le noyau de re-décalant le centre du noyau de convolution d'un pixel à droite (resp. en bas). Les opérateurs de convolution les plus simples pour estimer les dérivées secondes directionnelles,

sont donc $\begin{pmatrix} 1 & -\mathbf{2} & 1 \end{pmatrix}$ pour $\frac{\partial^2 f}{\partial x^2}$, et $\begin{pmatrix} 1 \\ -\mathbf{2} \\ 1 \end{pmatrix}$ pour $\frac{\partial^2 f}{\partial y^2}$.

Il s'ensuit que le laplacien $\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$ pourra être estimé en utilisant le noyau de convolution, dit laplacien en 4-connexité :

$$L_4 = \begin{pmatrix} & 1 & \\ 1 & -\mathbf{4} & 1 \\ & 1 & \end{pmatrix} \quad (2.43)$$

Le laplacien correspond donc à une différence entre une moyenne des voisins du pixel, et le pixel lui-même, qui peut également se calculer par un autre masque 3×3 , dit laplacien en 8-connexité :

$$L_8 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad (2.44)$$

Les figures 2.10 et 2.11 illustrent l'estimation de grandeurs différentielles à partir de convolution par des masques contenus dans un voisinage 3×3 . Ici encore, la plupart de ces images sont à valeurs signées, le zéro est donc ici représenté par le niveau de gris moyen. La norme du gradient est en revanche positive, elle est ici représentée en video inverse : le blanc représente le zéro, et le niveau de noir est proportionnel à la valeur de la norme.

Les figures 2.10 et 2.11 illustrent bien l'intérêt immédiat des dérivées premières et secondes pour l'analyse d'images. En particulier, la notion intuitive de «contours», correspondant aux courbes de changement significatif du niveau de gris dans l'image devient plus facile à formaliser : un contour correspond aux lieux des maxima du module du gradient, ou aux lieux des changements de signe du laplacien.

Néanmoins, l'estimation des dérivées par des calculs limités au voisinage immédiat du pixel s'avère en général insuffisante, car elle fait ressortir les détails locaux, qui ne sont pas forcément significatifs au regard de l'application, et peut négliger une structure importante, mais dont l'étendue spatiale est trop grande pour être correctement révélée par les opérateurs 3×3 . Ce problème fait l'objet des traitements multi-échelles, présentés dans le chapitre suivant.

2.4 Restauration

Dans cette partie, on s'intéresse à la suppression d'un défaut de type convolutif, qui se traduit en pratique par une image floue, qui peut être due à un défaut de mise au point de l'optique, ou à un mouvement rapide du porteur de la caméra (flou de bougé). La résolution de ce problème est difficile dans le cas général lorsqu'on ne connaît pas les paramètres du flou observé (i.e. le noyau de convolution). C'est le problème dit de la déconvolution aveugle, qui ne sera pas traité ici.

En revanche lorsque le noyau de convolution est connu, on peut chercher à calculer une convolution inverse, qui se traduira par une division dans le domaine fréquentiel, c'est l'objet du filtrage inverse. En réalité, l'inversion directe dans le domaine de Fourier engendre de fortes instabilités dès que le défaut convolutif s'accompagne d'un bruit additif, même imperceptible. L'inversion doit donc être conditionnée en fonction d'une estimation de ce bruit additif, c'est l'objet du filtrage de Wiener.

2.4.1 Filtrage inverse

Supposons donc d'abord que l'image observée g soit le fruit de la dégradation d'une image f par un noyau de convolution d connu, et qu'on néglige le bruit additif :

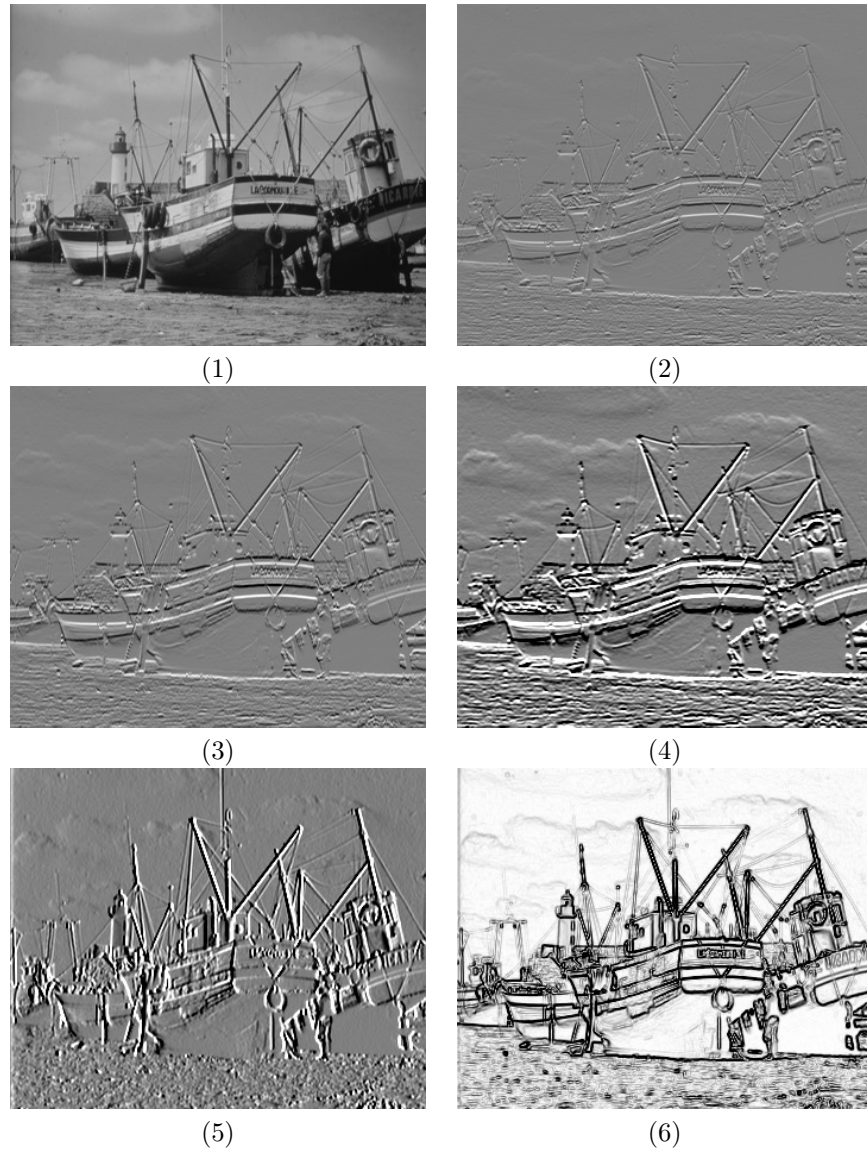


FIGURE 2.10 – Estimation de grandeurs différentielles à partir de convolution dans un voisinage 3×3 . (1) Image originale «boats». Dérivée partielle verticale estimée par convolution avec : (2) $(-1 \ 1)^T$ (3) $(-1 \ 0 \ 1)^T$ (4) Filtre de Sobel vertical h_y . (5) Dérivée partielle horizontale estimée par convolution avec le filtre de Sobel horizontal h_x . (6) Norme euclidienne du gradient de Sobel.



FIGURE 2.11 – Estimation du laplacien par convolution 3×3 . (1) Laplacien 4-connexe (2) Laplacien 8-connexe.

$$g(x) = f(x) \star d(x) \quad (2.45)$$

Soit, dans le domaine fréquentiel :

$$G(u) = F(u) \times D(u) \quad (2.46)$$

D'où l'estimée de f dans le domaine fréquentiel :

$$\hat{F}(u) = \frac{G(u)}{D(u)} \quad (2.47)$$

Et donc finalement, en notant r_d la transformée de Fourier inverse de $D(u)^{-1}$:

$$\hat{f}(x) = g(x) \star r_d(x) \quad (2.48)$$

Mais évidemment, $D(u)$ n'est pas toujours inversible, et d'un point de vue numérique, de très fortes instabilités se produiront lorsque la valeur de $D(u)$ est faible, d'où la définition de la pseudo-inverse, ou solution principale au sens de Bracewell [3] :

$$\hat{F}(u) = G(u) \times R(u), \text{ avec : } R(u) = \begin{cases} 0 & \text{si } D(u) < \varepsilon \\ \frac{1}{D(u)} & \text{sinon} \end{cases} \quad (2.49)$$

Où le seuil ε doit être déterminé en fonction de la précision de la représentation numérique.

La figure 2.12 illustre le problème de la restauration et du filtrage inverse dans le cas d'une image ayant subi un flou de bougé horizontal. Lorsque l'image, après convolution, est représentée en grande précision (ici en *float* 32 bits), le filtrage inverse simple peut permettre de retrouver l'image initiale, pour peu qu'aucune composante de la transformée de Fourier discrète ne soit strictement nulle. en revanche, le simple fait de quantifier l'image sur une représentation classique (ici en *byte* 8 bits), rend l'application du filtrage inverse simple inopérante, comme le montre la figure 2.12(5).

On notera que la différence entre les représentations *byte* et *float*, qui n'est pas perceptible dans le domaine spatial est visible sur le spectre grâce à la

représentation logarithmique du module de la transformée de Fourier. On voit en effet que le défaut de bougé, correspondant à un filtre moyenneur horizontal dans le domaine spatial, est bien visible par les lobes de son sinus cardinal horizontal dans le domaine fréquentiel, comme on le voit sur la figure 2.12(2), ce qui n'est plus le cas sur le spectre de l'image quantifiée, voir figure 2.12(4).

Finalement, le résultat du filtre pseudo-inverse sur l'image quantifiée est visible sur la figure 2.12(6).

2.4.2 Filtrage de Wiener

Cette section est adaptée de [30], chapitre «Restauration». Comme on l'a vu dans la section précédente, la déconvolution ne peut pas se réduire à l'inversion d'un opérateur linéaire, et il faut prendre en compte le terme de bruit additif :

$$g(x) = f(x) \star d(x) + b(x) \quad (2.50)$$

Le filtrage de Wiener [49] propose une solution à l'estimation de f par la minimisation d'une expression quadratique comportant un terme de régularisation :

$$\hat{f} = \arg \min_k \int (g(x) - k(x) \star d(x))^2 + (k(x) \star q(x))^2 \quad (2.51)$$

Le premier terme, $(g(x) - k(x) \star d(x))^2$ correspond au filtrage inverse, tandis que le second terme $(k(x) \star q(x))^2$ est une contrainte linéaire de régularisation. En passant dans le domaine de Fourier, si l'on impose à chaque composante fréquentielle de minimiser sa contribution à la somme, on obtient l'équation suivante :

$$\hat{F} = \arg \min_K |G - KD|^2 + |KQ|^2 \quad (2.52)$$

Le problème de minimisation est alors résolu en annulant la dérivée première de l'expression selon K :

$$\frac{\partial (|G - KD|^2 + |KQ|^2)}{\partial K} = 0 \quad (2.53)$$

et donc

$$-2D^*(G - DK) + 2Q^2K = 0 \quad (2.54)$$

soit

$$K(DD^* + Q^2) = D^*G \quad (2.55)$$

et en remplaçant K par \hat{F} , on obtient finalement l'équation du filtrage de Wiener :

$$\hat{F} = \frac{D^*}{DD^* + Q^2} \times G \quad (2.56)$$

avec D^* le conjugué de D , Q^2 le terme de régularisation.

On peut noter que si Q est nul partout, on retrouve le filtrage inverse. Dans le cas général, lorsque D est très faible, c'est le terme Q^2 qui devient prépondérant

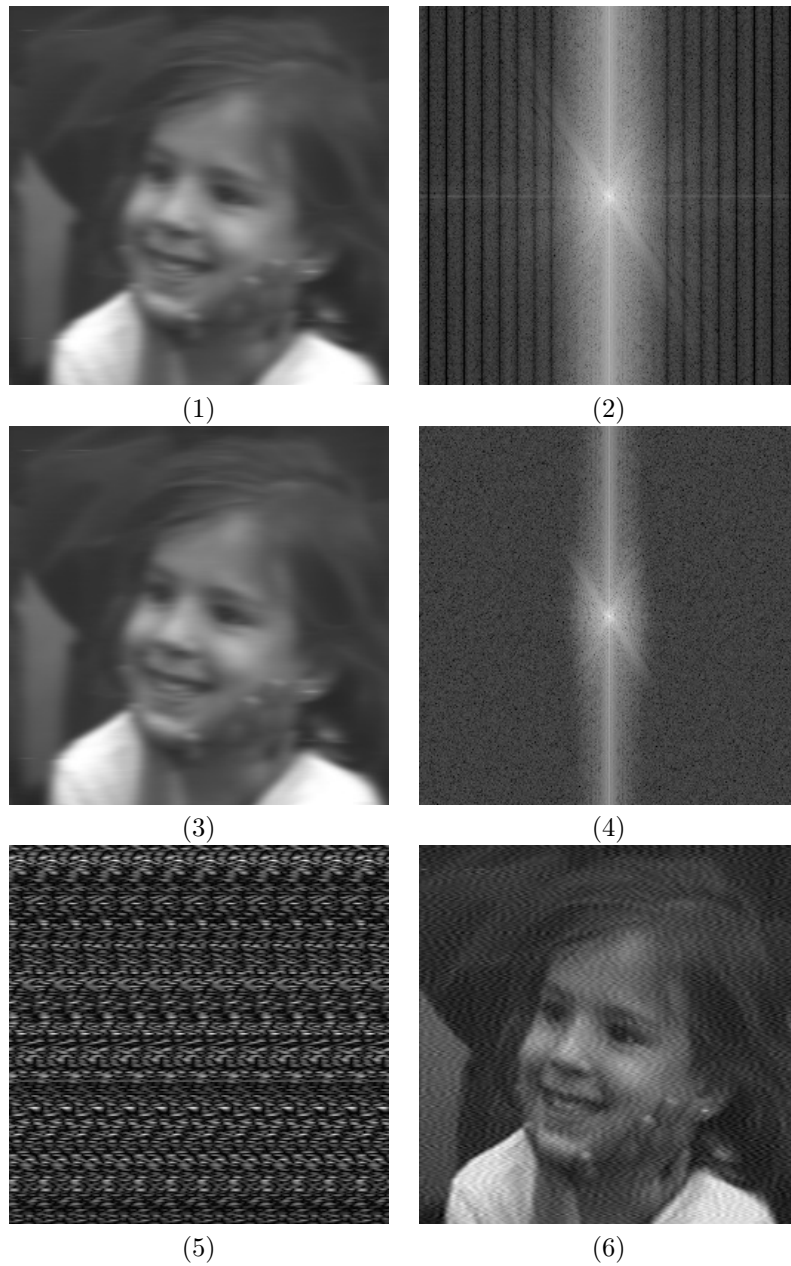


FIGURE 2.12 – Image avec flou de bougé horizontal. (1) Image représentée en grande précision (float), et (2) sa transformée de Fourier. (3) Image représentée en précision normale (byte), et (4) sa transformée de Fourier. (5) Filtrage inverse obtenu à partir de la transformée de Fourier de l'image en byte. (6) Filtrage pseudo-inverse obtenu à partir de la transformée de Fourier de l'image en byte. (©Jordán et Lotufo 1997, tiré de [22])

et qui permet de réaliser un compromis entre déconvolution et amplification du bruit.

Le principe consiste donc à fixer Q^2 en fonction d'une estimation de l'intensité relative du bruit par rapport à l'image. Idéalement, on pourrait définir un coefficient Q variant pour chaque composante fréquentielle, en fonction du rapport des modules de chaque composante :

$$Q_a^2(u) = \frac{|B(u)|^2}{|F(u)|^2} \quad (2.57)$$

En pratique on utilise plutôt un coefficient Q constant sur l'ensemble des fréquences, défini comme le rapport des moyennes des modules au carré sur l'ensemble des fréquences, estimées pour le bruit et pour l'image originale :

$$Q_b^2 = \frac{\langle |B(u)|^2 \rangle}{\langle |F(u)|^2 \rangle} \quad (2.58)$$

La figure 2.13 illustre le filtrage de Wiener sur la même image que précédemment. Le module du filtre de Wiener obtenu à partir de D en utilisant un coefficient Q constant est montré figure 2.13(3). Par multiplication de ce filtre avec la transformée de Fourier de l'image à traiter, dont on voit le module sur la figure 2.13(2), puis par transformée de Fourier inverse, on obtient l'image traitée, figure 2.13(4).

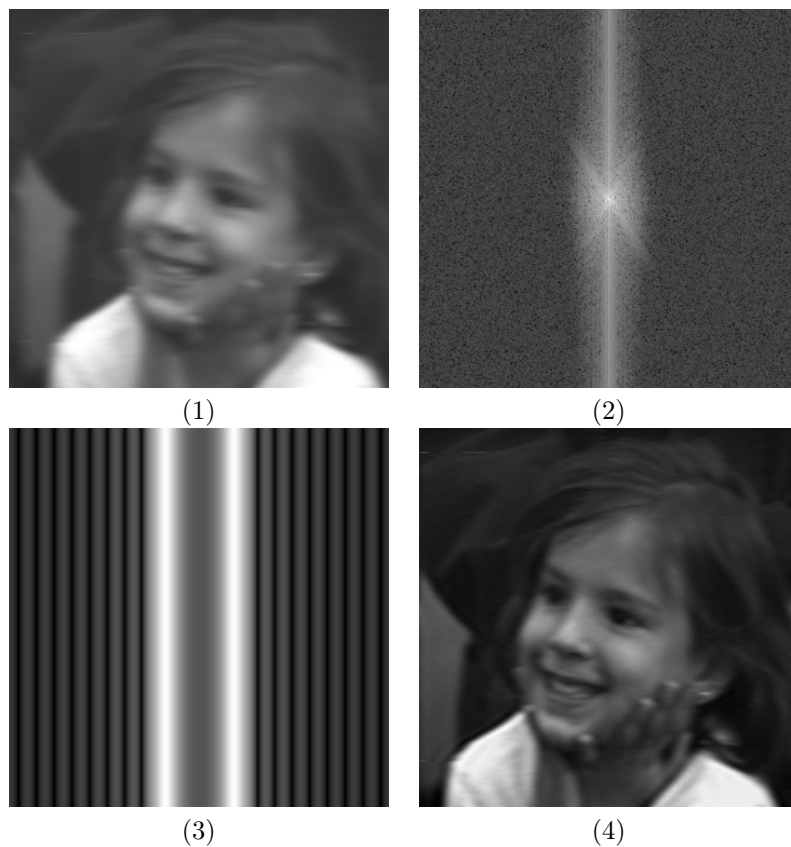


FIGURE 2.13 – Image avec flou de bougé horizontal. (1) Image représentée en précision normale (byte), et (2) sa transformée de Fourier. (3) Transformée de Fourier du Filtre de Wiener, avec Q constant. (4) Filtrage de Wiener de l'image floue (byte) initiale. (©Jordán et Lotufo 1997, tiré de [22])

