

TP 1

Lissage et détection de contours

PAUL BOUTON | TONY BUTHOD-GARÇON

23 mars 2017

1 Lissage linéaire

1.1 FFT et filtrage fréquentiel

Dans cette partie, nous allons étudier le comportement du filtre gaussien en fonction de la valeur de σ , du bruit de l'image, et de son niveau. Pour mesurer la qualité du filtrage, nous procéderons à une comparaison visuelle, mais aussi à une comparaison numérique à l'aide du PSNR.

Tout d'abord, commençons par comparer la qualité du filtre pour différents types de bruit (gaussien, speckle, poivre et sel). Nous utilisons ici les images formes2bb10, formes2pets1, formes2sp1.

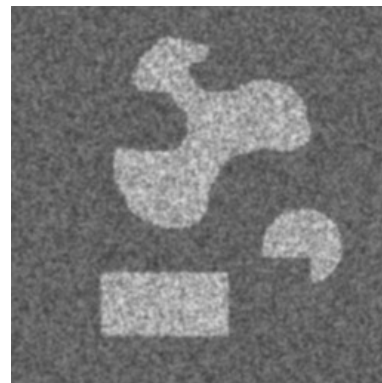
$\sigma = 1.0$



Gaussian. PSNR = 34.65



Poivre et sel. PSNR = 28.81

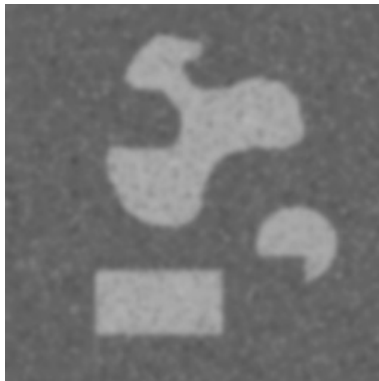


Speckle. PSNR = 27.23

$\sigma = 2.0$



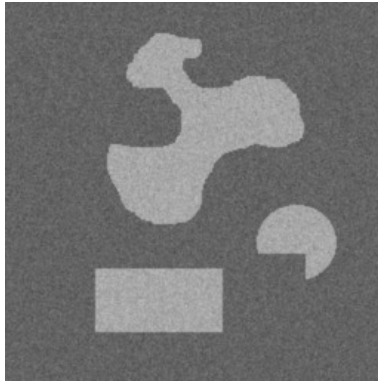
Gaussian. PSNR = 32.85



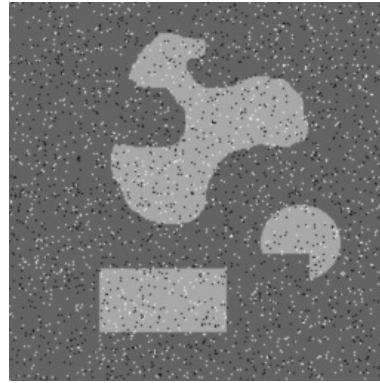
Poivre et sel. PSNR = 30.86



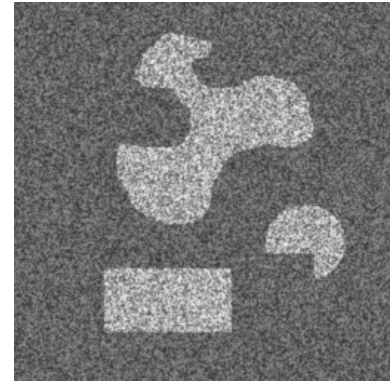
Speckle. PSNR = 30.39

$\sigma = 0.5$ 

Gaussian. PSNR = 32.73



Poivre et sel. PSNR = 23.90

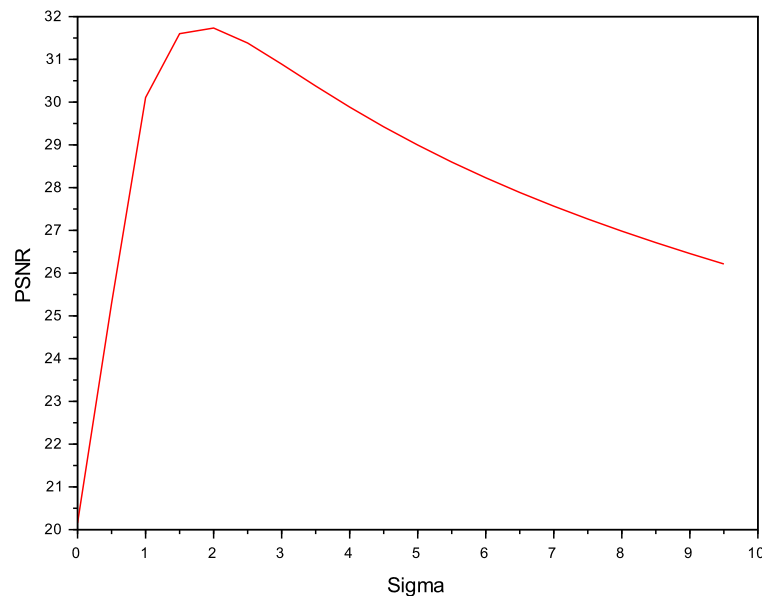


Speckle. PSNR = 21.93

Visuellement, le filtre gaussien fournit de meilleurs résultats pour une image entachée d'un bruit gaussien. En effet, dans le cas où $\sigma = 2.0$, le bruit gaussien n'est presque plus visible. A l'inverse, le bruit poivre et sel et speckle sont encore visibles. Ce résultat est conforté par l'analyse du PSNR. Celui-ci est maximum lorsque le bruit est gaussien.

Cependant, ce filtre a tendance à lisser les contours si le paramètre σ est élevé comme le montre les images filtrées avec $\sigma = 2$: Les images filtrées semblent floutées et de mauvaise qualité.

Cet effet peut être mis en avant en regardant l'évolution du PSNR en fonction du paramètre σ .



On voit alors que le PSNR est maximal pour une certaine valeur de σ . Lorsque que sigma est trop petit, le bruit n'est presque pas atténué, et lorsque σ est trop grand, le lissage des contours devient trop important.

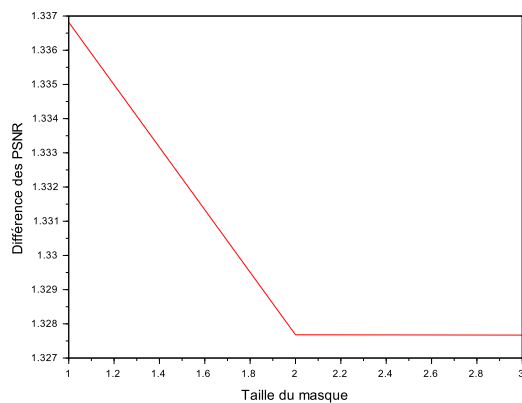
1.2 Convolution spatiale

Dans cette partie, nous allons étudier le filtre linéaire gaussien en fonction de la valeur de σ et par rapport au masque gaussien précédent. Ce filtre s'applique directement sur l'image par

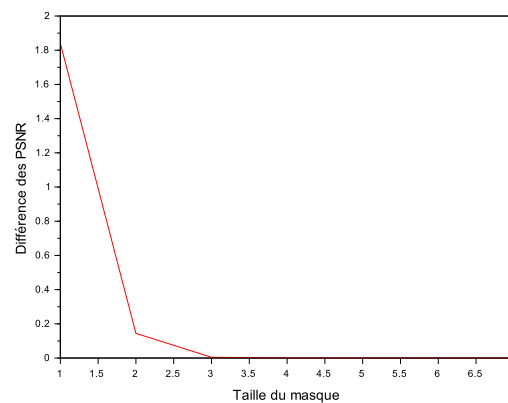
convolution sans passer par une transformée de Fourier. Nous allons voir quels sont ses avantages et inconvénients.

Pour gérer les effets de bords, l'image est périodisée. Ce choix est pertinent dans le cas où l'objet photographié se trouve au milieu de la photo. Dans le cas où l'objet est coupé sur un côté de l'image, ce choix est moins pertinent, et il sera préférable de prendre l'image miroir pour gérer les effets de bord.

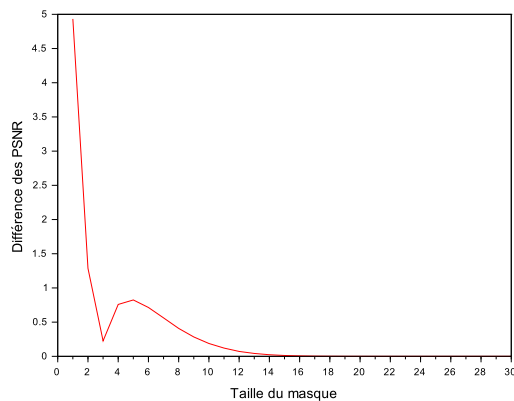
Pour différentes valeurs fixées de $\sigma \in \{0.5, 1, 5, 10\}$, le PSNR est calculé pour les deux filtres (filtre gaussien et filtre gaussien linéaire). Le but est alors de trouver la plus petite taille du masque pour laquelle le PSNR du filtre gaussien linéaire est suffisamment proche du PSNR du filtre gaussien. On trace la différence des PSNR pour l'image 'formes2bb50.pgm' pour différentes valeurs de σ , on obtient alors les graphiques suivants :



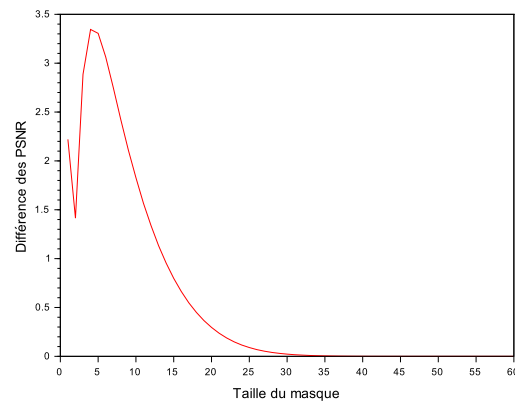
$\sigma = 0.5$



$\sigma = 1$



$\sigma = 5$



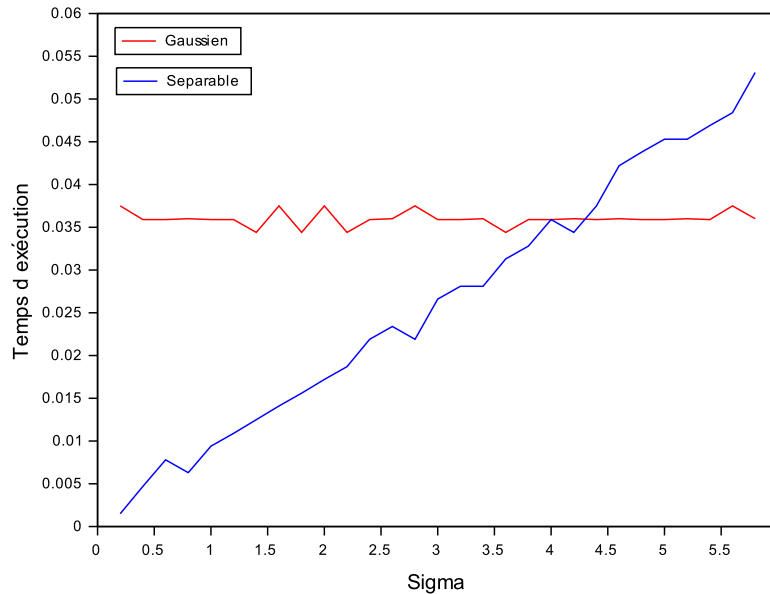
$\sigma = 10$

On voit alors que la différence des PSNR se rapproche de 0 quand la taille du masque augmente. Ainsi, pour $\sigma = 0.5$ la taille du masque n est environ 2 pour que la différence soit proche de 0. Pour $\sigma = 1$, $n \approx 3.5$. Pour $\sigma = 5$, $n \approx 17$. Et pour $\sigma = 10$, $n \approx 37$. On peut alors établir la loi empirique que la taille du masque doit être de l'ordre de 3.5 fois σ pour que les PSNR soient identiques.

1.3 Complexité et comparaison des 2 méthodes

Dans cette partie, nous allons comparer le temps d'exécution des différentes méthodes pour un résultat identique.

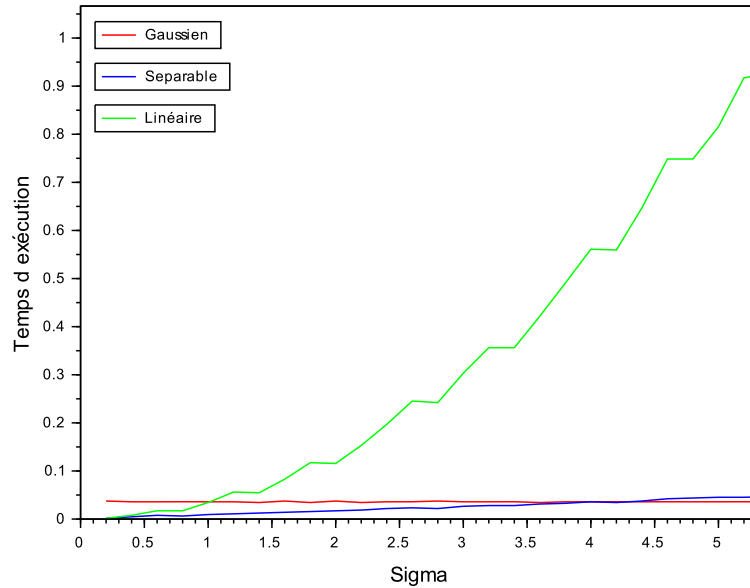
Sur le graphique suivant, on peut voir l'évolution du temps de calcul en fonction de σ et en prenant la relation $n = 3.5\sigma$ pour la taille du masque.



On peut donc voir que le filtre gaussien par fft possède un temps d'exécution constant quelque soit σ . A l'inverse, le filtre linéaire spatial possède un temps d'exécution croissant, puisque la taille du masque augmente avec σ .

Ainsi, pour $\sigma < 4.25$, il est préférable d'utiliser un filtre linéaire séparable, et lorsque $\sigma > 4.25$, il est préférable d'utiliser le filtre gaussien sur le domaine fréquentiel.

On peut aussi comparer les différences de performances entre un filtre séparable et un filtre non séparable. Sur le graphique suivant, le temps d'exécution d'un filtre non-séparable est plus de 10 fois supérieur à celui d'un filtre séparable. On peut donc voir l'importance d'un tel filtre sur le temps d'exécution.



2 Détection de contours

Lors de ce TP, nous avons implémenté deux techniques de détection de contours. La première utilise le gradient, un opérateur différentiel du premier ordre, la seconde, quant à elle, repose sur le laplacien, un opérateur différentiel du second ordre.

2.1 Méthodes mises en œuvre

Pour chacune des deux méthodes, nous avons choisi d'implémenter un filtre spécifique. Pour le gradient, notre choix s'est porté sur le filtre de Prewitt, associé au masque suivant :

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

Pour la détection de contours par laplacien, nous avons retenu le masque ci-après :

$$\begin{pmatrix} -0 & 1 & 0 \\ -1 & -4 & 1 \\ -0 & 1 & 0 \end{pmatrix}$$

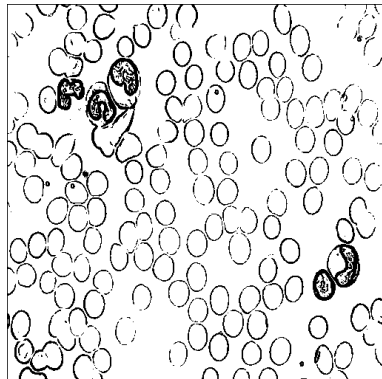
Afin d'améliorer nos résultats, nous avons procédé à un filtrage gaussien préalable à la détection de nos contours. Ainsi, nous pouvons jouer sur deux paramètres afin d'améliorer la précision de nos résultats :

-Le coefficient σ de notre filtre gaussien : Plus ce dernier sera élevé, plus notre image sera débruitée afin la détection mais plus nos contours seront flouté et donc épais. Il conviendra donc d'ajuster ce paramètre afin d'obtenir le meilleur compromis entre détection et précision.

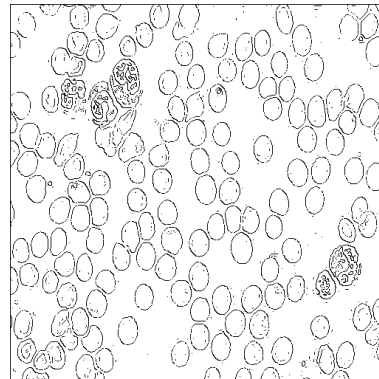
-Le seuil de détection : Pour le gradient, il s'agit de la valeur du module au dessus de laquelle on considère avoir détecté un contours. Cette dernière correspond à un pourcentage de la valeur maximale du module du gradient sur l'ensemble de notre image. En effet, tout les contours n'ayant pas la même intensité, on ne peut pas seulement retenir la valeur maximale et couper les autres valeurs. Pour le laplacien, la détection du passage par zéro s'avère plus délicate. Effectivement, ce passage ne se situe pas forcément sur notre grille de pixel qui est un milieu discret. Il faut donc tester le signe du produit de tout les pixels adjacents afin de détecter notre 0. Cependant, si on se contente de retenir les pixels dont le produit est négatif (et qui sont donc de signe opposé), on est beaucoup trop sensible au bruit (un bruit même faible, provoquera un passage par 0 du laplacien). Il est donc nécessaire de tester le module du produit en plus de son signe afin de détecter des passages par 0 "brusques" qui correspondant à des contours de notre image. Après plusieurs expérimentations, nous avons fixé ce module seuil à 1 : Tout produit négatif de module inférieur à 1 sera donc considéré comme un contours.

2.2 Comportement face au bruit

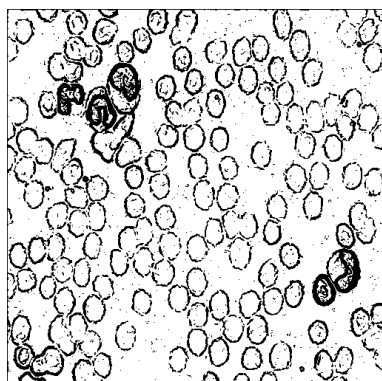
Si les deux techniques sont similaires, elles n'ont pas le même comportement vis à vis du bruit. Afin d'illustrer cela, prenons comme exemple le traitement des photos de globules rouges proposées. Ci dessous, nous avons successivement appliqué nos deux filtres de l'image la moins bruitée à la plus bruitée (sans bruit, bb10, bb25 et bb50) en choisissant nos deux paramètres (σ et seuil) de manière optimale :



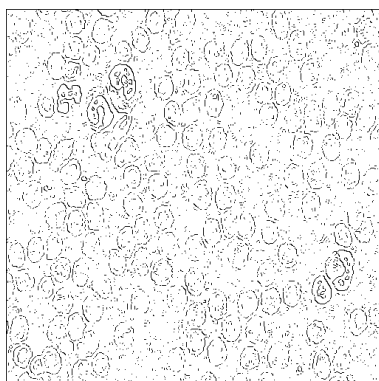
Gradient : $seuil = 0.2\%$, $\sigma = 0.0$.



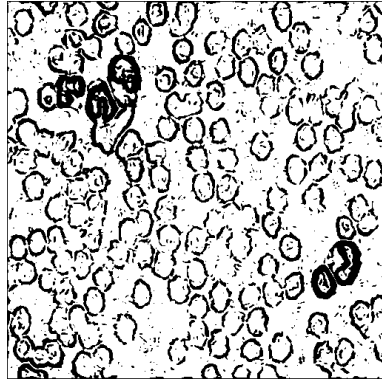
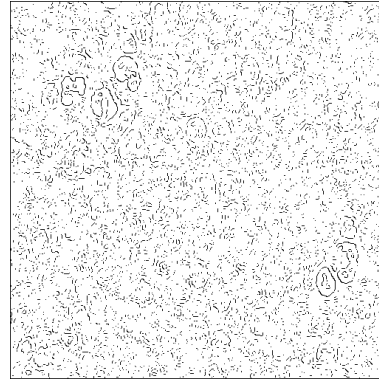
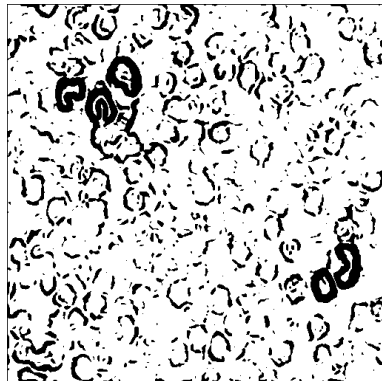
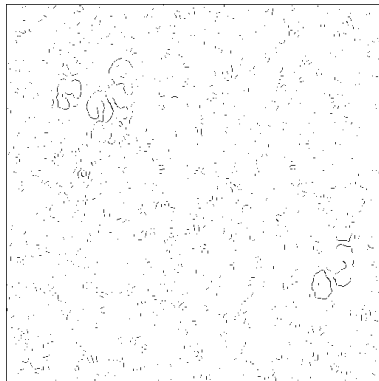
Laplacien : $\sigma = 1.3$



Gradient : $seuil = 1.5\%$, $\sigma = 1.0$



Laplacien : $\sigma = 1.9$

Gradient : $seuil = 1.8\%$, $\sigma = 2.0$.Laplacien : $\sigma = 2.3$ Gradient : $seuil = 2.5\%$, $\sigma = 3.0$ Laplacien à droite : $\sigma = 3.0$

On s'aperçoit que pour des images non bruitées, le laplacien fournit des résultats bien plus précis que le gradient : Les contours sont plus fins et plus détaillés. Ainsi, pour la première image (qui est non bruitée), le laplacien parvient à détecter les membranes cellulaires internes ce qui n'est pas le cas du gradient. Cependant, dès que l'on rajoute du bruit, même en très faible quantité, le laplacien devient très imprécis voir inutilisable (images 3 et 4) Le gradient quant à lui, permet une détection tout à fait acceptable même en présence de bruit : Même si on observe un élargissement des contours dû à un filtrage de plus en plus fort, on parvient quand même à garder la forme générale des contours.

En définitive, il faudra donc adapter le choix de notre filtre à la nature de notre image (bruitée ou non) et choisir le degrés de filtrage préalable le plus adapté.