

Compte rendu : Déconvolution d'images

Elise Bachellerie
bachelle@insa-toulouse.fr
Philippe Conesa
conesa@insa-toulouse.fr
Manon Santrisse
santrisse@insa-toulouse.fr

5^{eme} année, spécialité Mathématiques Appliquées

Encadrant :
Charles DOSSAL

Date : 9 janvier 2023

Table des matières

1	Introduction	1
2	Déconvolution de Wiener	1
3	Algorithmes de minimisation	2
3.1	Forward-Backward	2
3.2	FISTA	2
3.3	Douglas Rachford	3
4	Déconvolution en ondelette	3
4.1	Résolution numérique par Forward-Backward	3
4.2	Résolution numérique par FISTA	4
4.3	Résolution numérique par Douglas-Rachford	5
5	Transformée en ondelettes invariante par translation (Translation invariant wavelet transform)	6
5.1	Résolution numérique par Forward-Backward	7
5.2	Résolution numérique par FISTA	8
6	Déconvolution avec la régularisation TV (Total Variation)	8
6.1	Algorithme Primal Dual : Chambolle-Pock	9
6.2	Algorithme Primal Dual : Condat	10

1 Introduction

Au cours de ce projet nous avons testé plusieurs façons de déconvoluer une image. Suivant l'image à déconvoluer, il est important de choisir la bonne formulation variationnelle. En effet, certaines sont plus adaptées à des images sparse (régularisation en ondelettes) tandis que d'autres sont adaptées à des images homogènes c'est-à-dire constantes par morceaux (régularisation TV - Total Variation).

Dans la suite de ce rapport, nous vous présenterons les différentes déconvolutions que nous avons testées.

2 Déconvolution de Wiener

Dans un premier temps, nous avons appliqué le filtrage de Wiener qui est une technique de déconvolution de base mais très peu performante. Elle consiste à minimiser la fonction suivante :

$$F(x) = \frac{1}{2} \|h \star x - y\|_2^2 + \frac{\lambda}{2} \|x\|_2^2 \quad (1)$$

On a alors la transformée de Fourier de x le minimiseur de F qui s'écrit comme suit :

$$\hat{x}(\omega) = \hat{y}(\omega) \frac{\overline{\hat{h}(\omega)}}{|\hat{h}(\omega)|^2 + \lambda} \quad (2)$$

Ici, notre but est de trouver le λ optimal en fixant le kernel h et l'intensité du bruit σ^2 . Pour cela nous avons créé un dashboard et fait varier λ . Comme nous pouvons le voir sur la figure 1, en fixant $\sigma^2 = 10$, nous avons obtenu le meilleur PSNR pour $\lambda = 0.055$.

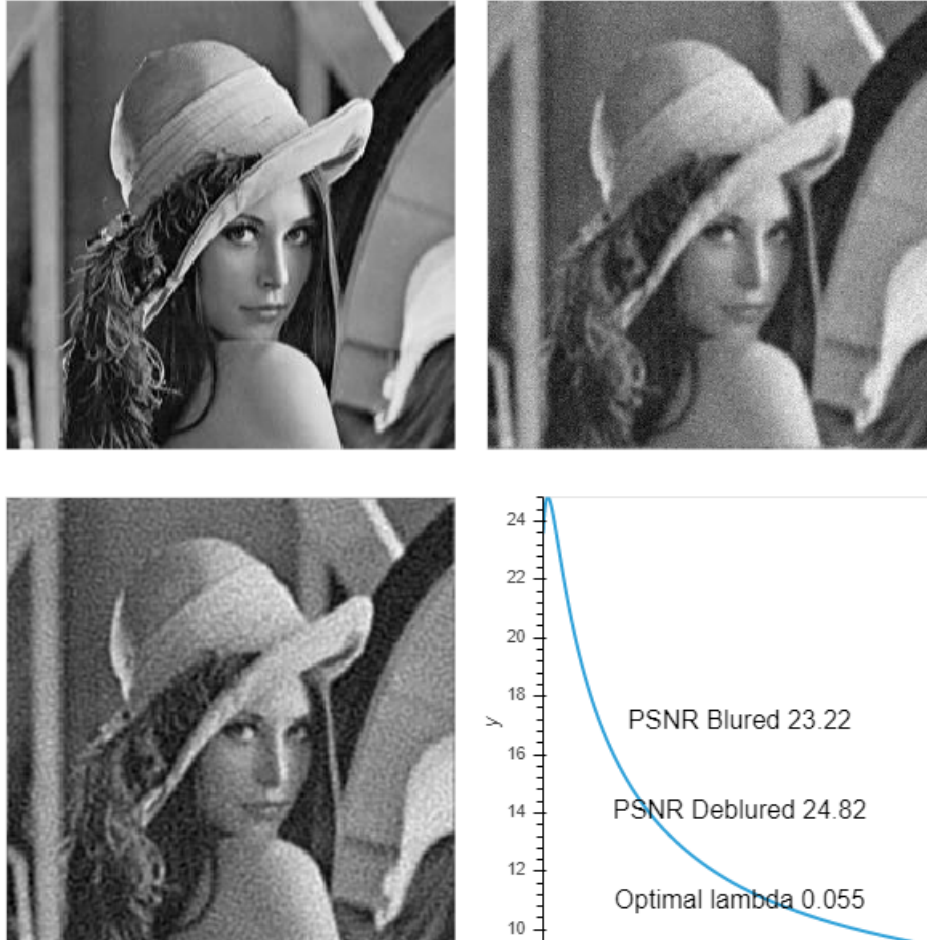


FIGURE 1 – λ optimal pour le filtre de Wiener avec $\sigma^2 = 10$

3 Algorithmes de minimisation

Lors de ce projet, nous avons cherché à résoudre le problème de minimisation suivant :

$$\min_{x \in E} F(x) = \min_{x \in E} f(x) + g(x) \quad (3)$$

où

- $F=f+g$ une fonction convexe composite, car elle est la somme de deux fonctions convexes ;
- f une fonction différentiable à gradient L -Lipschitz ;
- g une fonction convexe dont on sait calculer l'opérateur proximal.

Pour cela, nous avons eu recours aux algorithmes Forward-Backward, FISTA et Douglas Rachford. Ces algorithmes sont utilisés dans le cas où g est une norme L_1 .

3.1 Forward-Backward

L'algorithme Forward-Backward s'écrit :

$$x_{n+1} = \text{prox}_{hg}(x_n - \gamma \nabla f(x_n)) = Tx_n \quad \text{avec } T := \text{prox}_{hg} \circ (Id - \gamma \nabla f) \quad (4)$$

La suite $(x_n)_{n \in \mathbf{N}}$ converge vers un minimiseur de F si $\gamma < \frac{2}{L}$.

L'algorithme Forward-Backward peut apparaître lent et n'offre pas toujours les meilleurs résultats, il faut balayer pour chercher des paramètres optimaux et s'assurer de la convergence.

On cherche à obtenir de meilleurs résultats, en termes de qualité de reconstruction de l'image, mais aussi de vitesse de convergence de l'algorithme. Ainsi, on va mettre en place une seconde stratégie : l'accélération de Nesterov ou FISTA, qui est basée sur l'algorithme Forward-Backward, mais qui comporte un terme inertiel, censé accélérer l'algorithme, même s'il n'est donc pas réellement un algorithme de descente.

3.2 FISTA

L'algorithme FISTA pour Fast Iterative Shrinkage Thersholding Algorithm applique l'opérateur proximal sur la partie non différentiable de la formulation variationnelle ($T = \text{prox}_{\gamma g} \circ (Id - \gamma \nabla f)$) en appliquant un terme inertiel.

Pour l'opérateur T , on applique une descente de gradient en un point décalé de x_n avec un pas $\gamma < \frac{1}{L}$:

$$x_{n+1} = T(x_n + \alpha_n(x_n - x_{n-1}))$$

Avec $\alpha_n = \frac{n-1}{n+\alpha-1}$. La suite $(x_n)_{n \in \mathbf{N}}$ converge vers un minimiseur x^* de la fonctionnelle F (F sci, conexe et coecive) pour $\alpha > 3$. Dans la pratique, on choisirat α très proche de 3.

Pour l'algorithme FISTA, dans notre cas où F est une fonction composite avec g correspondant à une norme L_1 , l'opérateur proximal est un seuillage doux.

Si l'algorithme Forward-Backward est utilisé pour résoudre le problème de minimisation de notre fonction composite, il est intéressant d'appliquer l'algorithme FISTA qui est une accélération de la première méthode. Ainsi par comparaison, la méthode Forward-Backward converge telle que $F(x_n) - F(x^*) = \mathcal{O}\left(\frac{1}{n\gamma}\right)$, alors que pour $\alpha > 3$, la méthode FISTA converge plus rapidement telle que : $F(x_n) - F(x^*) = \mathcal{O}\left(\frac{1}{n^2}\right)$.

Théoriquement sur des fonctions fortement convexes, pour l'algorithme Forward Backward, la suite de termes généraux $F(x_n) - F(x^*)$ où x^* est un minimiseur, décroît géométriquement ce qui n'est pas le cas de FISTA présentée ici qui sont au mieux polynomiales.

3.3 Douglas Rachford

Dans cette partie, on veut résoudre un problème de la forme :

$$\min_{x \in E} F(x) = \min_{x \in E} f(x) + g(x)$$

où f et g sont convexes sur X , un espace euclidien.

La forme générale de l'algorithme de Douglas Rachford est la suivante :

$$\begin{cases} u_n &= \text{prox}_{\gamma f}(x_{n-1}) \\ z_n &= \text{prox}_{\gamma g}(2 * u_n - x_{n-1}) \\ x_n &= x_{n-1} + z_n - u_n \end{cases}$$

où $\text{prox}_{\gamma f}(x) = \arg \min_{y \in X} \frac{1}{2} \|x - y\|^2 + \gamma F(y)$

L'algorithme d'optimisation de Douglas-Rachford est un algorithme itératif utilisé pour résoudre des problèmes d'optimisation convexe. Il s'agit d'une méthode de résolution de problèmes d'optimisation qui consiste à trouver le minimum (ou le maximum) d'une fonction en suivant une série d'étapes itératives.

L'algorithme de Douglas-Rachford est basé sur la méthode de projection sur un convexe, consistant à projeter un point sur l'ensemble de définition d'une fonction convexe de manière à minimiser la distance entre le point de départ et le point projeté. Pour cela, on utilise l'opérateur proximal d'une fonction convexe (ici f ou g). Cet opérateur calcule la projection d'un point sur l'ensemble de définition de la fonction auquel il est associé.

L'algorithme de Douglas-Rachford utilise cette méthode pour résoudre des problèmes d'optimisation en itérant sur la projection de points sur l'ensemble de définition de fonctions convexes jusqu'à ce qu'une solution optimale soit trouvée.

4 Déconvolution en ondelette

Dans cette partie, nous avons cherché à résoudre le problème de minimisation suivant :

$$\min_x \frac{1}{2} \|h \star x - y\|_2^2 + \lambda \|Wx\|_1 \quad (5)$$

où W est une transformée orthogonale en ondelettes.

L'utilisation de cette formulation variationnelle est pertinente si les coefficients en ondelette de l'image cible sont sparse. Pour résoudre ce problème de minimisation, nous avons utilisé les trois algorithmes présentés précédemment.

4.1 Résolution numérique par Forward-Backward

Nous appliquons dans un premier temps la méthode Forward-Backward, où $f(x) = \frac{1}{2} \|h \star x - y\|_2^2$ et $g(x) = \lambda \|Wx\|_1$, une norme L_1 appliquée à la transformée orthogonale en ondelettes. Pour appliquer l'algorithme Forward-Backward, on définit tout d'abord le gradient de la fonction f , obtenue avec des transformations de Fourier discrète. L'opérateur proximal de g correspond quant à lui à un Seuillage Doux en Ondelettes.

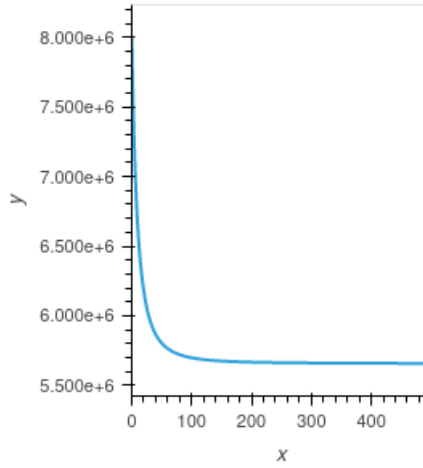


FIGURE 2 – Fonction coût algorithme Forward-Backward avec une déconvolution en ondelette avec 500 itérations

Le PSNR obtenu pour cet algorithme avec un λ égal à 5 est de 25.799. L'image bruitée de départ avait quant à elle un PSNR de 23.211. Le PSNR est bien amélioré donc l'image est débruitée.

4.2 Résolution numérique par FISTA

De la même manière que pour la méthode Forward-Backward, pour appliquer l'algorithme FISTA, le gradient de la fonction f est nécessaire et l'opérateur proximal de g correspond à un seuillage doux en ondelettes.

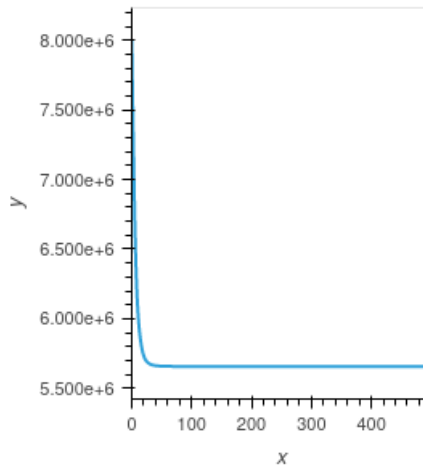


FIGURE 3 – Fonction coût algorithme FISTA avec une déconvolution en ondelette avec 500 itérations

Le PSNR obtenu pour cet algorithme avec un λ égal à 5 est de 25.657. L'image bruitée de départ avait quant à elle un PSNR de 23.211. Le PSNR est bien amélioré donc l'image est débruitée.

On note qu'avec cette méthode, le problème de minimisation est résolu plus rapidement, cependant le PSNR est moins bon qu'avec une descente de gradient telle que Forward-Backward.

La figure 4 permet d'observer les images débruitées avec ces deux algorithmes résolvant le problème de minimisation (5).

Image bruitée



Image Déconvolution Wiener optimale



Image débruitée Forward-Backward



Image débruitée FISTA



FIGURE 4 – Comparaison avec l'image bruitée, la déconvolution de Wiener et la déconvolution en ondelette obtenue avec les algorithmes Forward-Backward et FISTA

4.3 Résolution numérique par Douglas-Rachford

A présent, regardons les résultats numériques de l'algorithme Douglas-Rachford.

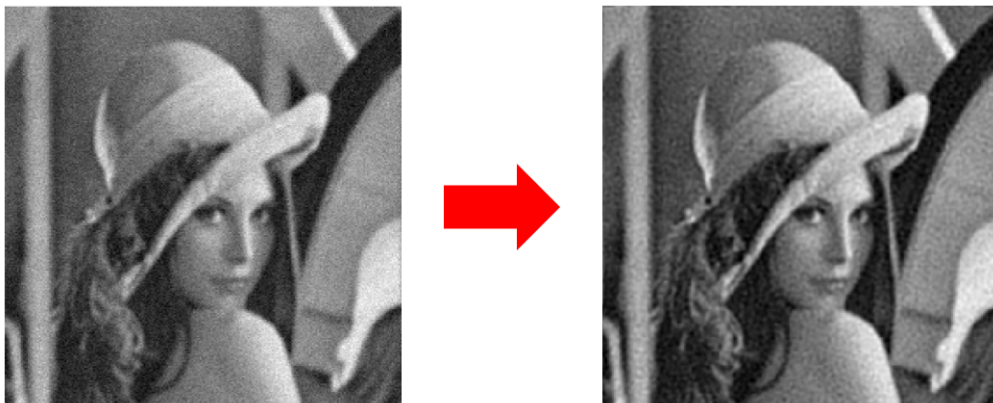


FIGURE 5 – Défloutage à l'aide de l'algorithme Douglas-Rachford avec 500 itérations, $\gamma = 1$, $\rho = 0.001$

Tout d'abord, on prend une image que l'on vient bruite, donnant cet aspect granuleux. En appliquant l'algorithme de Douglas-Rachford avec 500 itérations et en prenant $\gamma = 1$ et $\rho = 0.001$ nous obtenons une image débruitée, avec beaucoup moins de grains (figure 5). Les contours des objets et du modèle sont mieux dessinés et précis. Les tons de noir et blanc sont également plus intenses.

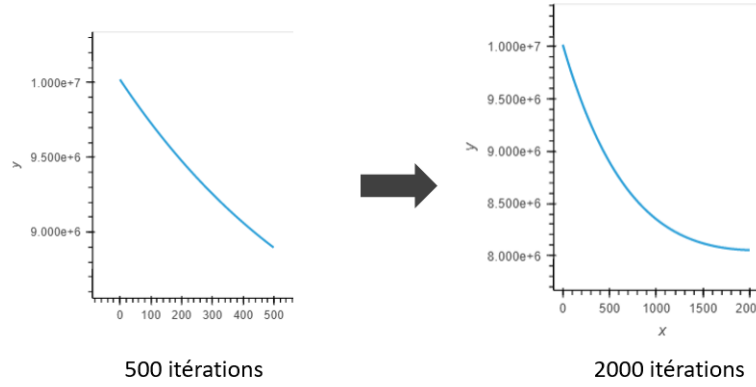


FIGURE 6 – Comparaison des fonctions coût entre 500 itérations et 2000 itérations

Par la suite, nous comparons les fonctions coût entre 500 itérations et 2000 itérations (figure 6). On observe bien que l'algorithme converge vers le minimum de la fonction. Cette tendance est plus visible avec 2000 itérations.

On essaie ensuite une autre photographie, que l'on vient également flouter (figure 7). L'image de départ est très floue : les contours des objets, du paysage et du modèle sont flous. En appliquant l'algorithme Douglas-Rachford avec 500 itérations, $\gamma = 1$, $\rho = 0.001$, l'image est très bien défloutée. On retrouve des contours précis.



FIGURE 7 – Défloutage à l'aide de l'algorithme Douglas-Rachford avec 500 itérations, $\gamma = 1$, $\rho = 0.001$

5 Transformée en ondelettes invariante par translation (Translation invariant wavelet transform)

La déconvolution peut être améliorée en utilisant une transformée en ondelettes invariante par translation (IT). Nous pouvons résoudre le problème d'optimisation suivant :

$$\min_x \frac{1}{2} \|h \star x - y\|_2^2 + \lambda \|W_{TI}x\|_1 \quad (6)$$

où W_{TI} est une transformée orthogonale en ondelettes invariante par translation. Cette transformation implique que $W_{TI}^* W_{TI} = Id$ même si $W_{TI} W_{TI}^* \neq Id$.

Pour cette formulation variationnelle seuls les méthodes Forward-Backward et FISTA sont appliquées. Pour ces deux résolutions numériques, le gradient de la fonction f est nécessaire et l'opérateur proximal de la fonction g avec $g(x) = \lambda \|W_{TI}x\|_1$ est un seuillage doux des coefficients en ondelettes :

$$\text{prox}_{\|W_{TI}\cdot\|_1}(x) = W_{TI}^* \circ \text{SoftThresh} \circ W_{TI}(x). \quad (7)$$

5.1 Résolution numérique par Forward-Backward



FIGURE 8 – Comparaison image bruitée et image débruitée avec l'algorithme Forward-Backward pour une déconvolution en ondelettes invariante par translation pour 400 itérations

Avec Forward-Backward, pour la déconvolution en ondelette invariante par translation, le PSNR est de 27.829 ($\lambda = 0.5$).

5.2 Résolution numérique par FISTA

Image bruitée

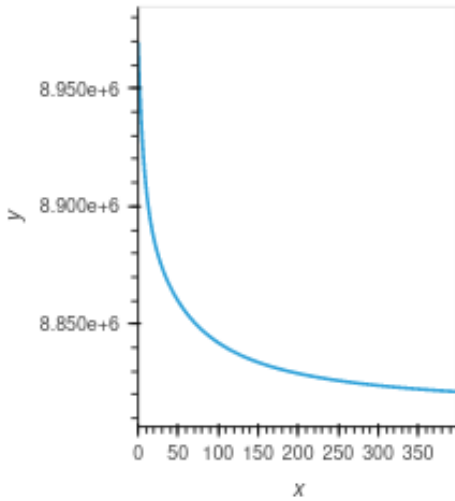


Image débruitée translation invariante FISTA

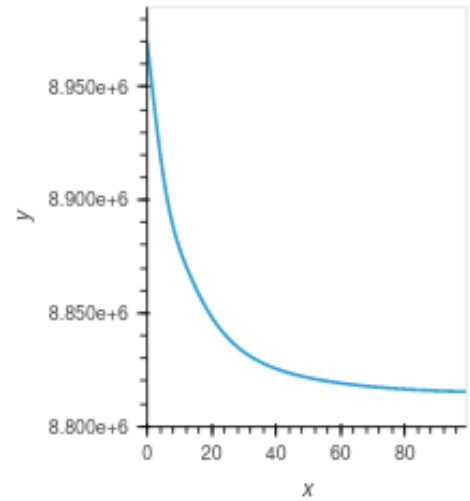


FIGURE 9 – Comparaison image bruitée et image débruitée avec l’algorithme FISTA pour une déconvolution en ondelettes invariante par translation pour 100 itérations

Avec FISTA, pour la déconvolution en ondelette invariante par translation, le PSNR est de 28.104 ($\lambda = 0.5$).



(a) Forward-Backward pour 400 itérations



(b) FISTA pour 100 itérations

FIGURE 10 – Fonction coût pour la déconvolution en ondelette invariante par translation

6 Déconvolution avec la régularisation TV (Total Variation)

La formulation variationnelle à minimiser est la suivante :

$$\min_x F(x) = \min_x \frac{1}{2} \|h \star x - y\|_2^2 + \lambda \|\nabla x\|_1 \quad (8)$$

6.1 Algorithme Primal Dual : Chambolle-Pock

Pour appliquer l'algorithme Primal Dual, le problème d'optimisation est différent :

$$\min_{x \in E} F(x) = \min_{x \in E} f(x) + g(Kx) \quad (9)$$

où K est un opérateur linéaire, f et g sont deux fonctions convexes propres, chacune possédant un opérateur proximal. Ainsi, la fonction F ne correspond pas à une fonction différentiable f et une fonction g non différentiable, dont l'opérateur proximal est nécessaire comme vu précédemment. Ici les deux opérateurs proximaux des fonctions f et g seront nécessaires.

L'algorithme est le suivant :

$$\begin{cases} y_{n+1} &= \text{prox}_{\sigma g^*}(y_n + \sigma K \bar{x}_n) \\ x_{n+1} &= \text{prox}_{\tau f}(x_n - \tau K^* y_{n+1}) \\ \bar{x}_{n+1} &= 2x_{n+1} - x_n \end{cases}$$

Avec $\text{prox}_{\sigma g^*}(x) = x - \sigma \text{prox}_{\sigma^{-1}g}(\sigma^{-1}x)$.

Dans notre cas,

$$F(x) = \frac{1}{2} \|h \star x - y\|_2^2 + \|\nabla x\|_1 \quad (10)$$

donc $f(x) = \frac{1}{2} \|h \star x - y\|_2^2$ et $g(Kx) = \|\nabla x\|_1$, où l'opérateur K correspond à l'opérateur ∇ tel que $K^* = -\text{div}$.

L'opérateur proximal de la fonction f est obtenu à l'aide de transformée de Fourier discrète. L'opérateur proximal de la fonction g est lié à l'opérateur proximal de la norme l_1 tel que :

$$\text{prox}_{\|\cdot\|_1}(x) = \begin{cases} x + \lambda & \text{si } x \leq -\lambda \\ 0 & \text{si } x \in [-\lambda, \lambda] \\ x - \lambda & \text{si } x \geq \lambda \end{cases} \quad (11)$$

Cet algorithme permet en particulier de pouvoir appliquer cette méthode sur des fonctions composites avec deux fonctions non différentiables. On note en particulier le caractère générique de cet algorithme, il peut être appliqué pour n'importe quelles fonctions différentiables ou non, il suffit de connaître leurs opérateurs proximaux. Cependant le choix de σ est très important pour la convergence de cet algorithme et le choix de ce paramètre dépend du point de départ choisi pour l'algorithme.



FIGURE 11 – Comparaison image bruitée et image débruitée avec Chambolle-Pock pour une déconvolution TV

Sur la figure 11, le PSNR final de l'iage débruitée était de 27.355. On représente sur la figure 12, l'image débruitée obtenue avec l'algorithme de Chambolle-Pock pour une déconvolution TV avec le plus haut PSNR. Pour un bruit de 6.80, le PSNR le plus haut est 28.08 pour $\lambda = 0.39$, $\tau = 0.25$ et $\gamma = 0.013$.



FIGURE 12 – Comparaison image bruitée et image débruitée avec Chambolle-Pock pour une déconvolution TV

6.2 Algorithme Primal Dual : Condat

L'algorithme de Condat permet la minimisation de problème du type :

$$F(x) = f(x) + g(x) + \sum_{m=1}^M h_m(L_m x) \quad (12)$$

avec

- f une fonction différentiable à gradient L -Lipschitz
- g et h_m qui ont des opérateurs proximaux connus
- L_k un opérateur linéaire

Dans notre cas, nous avons fixé les fonctions comme suit :

$$f(x) = \frac{1}{2} \|h \star x - y\|_2^2 \quad (13)$$

$$g = 0 \quad (14)$$

$$h_1(x) = \|x\|_1 \text{ avec } M = 1 \text{ et } L_1 x = \nabla x \quad (15)$$

Nous cherchons donc à minimiser un problème de la forme :

$$F(x) = \frac{1}{2} \|h \star x - y\|_2^2 + \|\nabla x\|_1 \quad (16)$$

L'algorithme de Condat est défini comme :

$$\begin{cases} \tilde{x}_{k+1} = x_k - \tau \nabla f(x_k) - \tau \operatorname{div}(u_k) \\ x_{k+1} = \rho \tilde{x}_{k+1} + (1 - \rho)x_k \\ \tilde{u}_{k+1} = \operatorname{Proj}_{\mathcal{B}_{\infty}(\frac{1}{\sigma})}(u_k + \sigma \nabla(2\tilde{x}_{k+1} - x_k)) \\ u_{k+1} = \rho \tilde{u}_{k+1} + (1 - \rho)u_k \end{cases} \quad (17)$$

quelque soit $\rho \in (0, 1)$ et τ et σ assez petits.

L'algorithme Primal Dual proposé par Condat a plusieurs avantages. Tout d'abord il a recours à une descente de gradient explicite sur la partie différentielle. De plus, dans le cas où $M > 1$, les $(u_k^m)_{m \leq M}$ sont indépendants et leur calcul peut être fait séparément. Cela permet une parallélisation des calculs qui peut accélérer grandement le temps de calcul total.

Nous avons fait varier les paramètres de l'algorithme de Condat grâce à un dashborad et nous avons trouvé les paramètres qui permettent de maximiser le PSNR et donc d'obtenir une meilleure image débruitée. Sur la figure 13, nous pouvons voir qu'en choisissant les paramètres $\lambda = 0.1$, $\tau = 0.01$, $\sigma = 1$, $\rho = 1$ et niter = 2000 l'algorithme de Condat est très efficace pour déconvoluer la photo de Lenna. En effet, on passe d'un PSNR de 24,38 à 28,54.



FIGURE 13 – Déconvolution par l'algorithme de Condat