Deconvolution algorithms: denoising images

Julien Blanchon, Nestor Carmona, Minh Vuong

I. Introduction

Let's consider a problem of the form:

$$y = h \star x + n \tag{1}$$

with: x the original image, h a Gaussian kernel of a given intensity, n a white noise of a given intensity and y the resulting blurred image; $\star$ the convolution product. The goal of this project is to find an approximation $\tilde{x}$ of x.

A. Approaches

The first approach consists of computing the Fourier Transform of y. Thus, eq.1 can be rewritten as follows:

$$\hat{x} = \hat{h}^{-1}(\hat{y} - \hat{n}) \tag{2}$$

Therefore, $\tilde{x}$ is equal to the inverse Fourier transform of eq.2. However, there is no guarantee $\hat{h}$ is inversible and, if it is, its coefficients are close to zero, thus its inverse's coefficients tend to infinity.

The second approach is to implement Wiener's method. For more details, see Sect.II.

Finally, the third approach, see Sect.III is to transform eq.1 into an optimisation problem with regularisation.

B. Metrics

Since evaluating an image's quality is subjective, we need some metrics to assess the methods' performance:

Definition 1.1 (Peak Signal to Noise Ratio):

$$PSNR(x, \tilde{x}) = 10 \log_{10}\left(\frac{MAX_{x_i}}{MSE(x, \tilde{x})}\right) \tag{3}$$

II. Wiener's filter

The Wiener filtering consists in minimizing the following function:

$$F(x) = \frac{1}{2}||h \star x - y||_2^2 + \frac{\lambda}{2}||x||_2^2 \tag{4}$$

The minimizer $\tilde{x}$ is given, for some g, by:

$$\tilde{x}(t) = g(t) \star y(t) \iff \tilde{X}(f) = G(f)Y(f) \tag{5}$$

Wiener gives a solution in the frequency domain:

$$G = \frac{H^*}{|H|^2 + \lambda} \tag{6}$$

with λ a parameter to estimate.

- Advantages: no need for iterations, gives a good approximation for the first step of the algorithms in Sect.III.
- Disadvantages: despite optimising for λ, the restored image is still noisy.

III. Optimisation problem

Eq.1 can be rewritten as the following optimisation problem:

$$F(x) = \underbrace{\frac{1}{2}||h \star x - y||_2^2}_{f(x)} + \underbrace{\lambda||Kx||_1}_{g(Kx)} \tag{7}$$

with K a linear operator. In this project, we used the following operators:

- W_{ortho}: orthogonal wavelet transform.
- W_{TI}: translation invariant wavelet transform.
- ∇: discrete gradient operator.

Let's remind that W_{ortho} and W_{TI} both verify $W^*W = Id$. Also, Moreau's identity:

$$prox_f(x) + prox_{f^*}(x) = x \tag{8}$$

We use eq.8 to compute the dual proximal operators.

Finally, depending on the properties verified by f and g, such as convexity, differentiability, etc., different solvers can be applied.

In the next page, we showcase a summary of the solvers, hypothesis and regularizations.

IV. Summary Tables

Name	**Hypothesis**	**Parameters**	**Iteration**
FB	f differentiable ∇f *L-Lipschitz* g convex $F(x) = \underbrace{f(x)}_{grad_f} + \underbrace{g(x)}_{prox_g}$	$\gamma \in]0, \frac{1}{L}[$	$x_{n+1} = prox_{\gamma g}(x_n - \gamma \nabla f(x_n))$
FISTA	f differentiable ∇f *L-Lipschitz* g convex $F(x) = \underbrace{f(x)}_{grad_f} + \underbrace{g(x)}_{prox_g}$	$\gamma \in]0, \frac{1}{L}[$ $\alpha \geq 3$	$\begin{cases} x_{n+1} = prox_{\gamma g}(y_n - \gamma \nabla f(y_n)) \\ y_n = x_n + \frac{n}{n+\alpha}(x_n - x_{n-1}) \end{cases}$
Douglas Rachford	f and g convex $F(x) = \underbrace{f(x)}_{prox_f} + \underbrace{g(x)}_{prox_g}$	$\mu \in]0, 1[$ $\gamma > 0$	$\begin{cases} y_{n+1} = prox_{\gamma g}(x_n) \\ z_{n+1} = prox_{\gamma f}(2y_{n+1} - x_n) \\ x_{n+1} = x_n + 2(1-\mu)(z_{n+1} - y_{n+1}) \end{cases}$ $u_n = prox_{\gamma g}(x_n)$ converges to $\tilde{x}$
PPXA	f and g convex $F(x) = \underbrace{f(x)}_{prox_f} + \underbrace{g(x)}_{prox_g}$	$\mu \in]0, 1[$ $\gamma > 0$	$\begin{cases} x^1_{n+1} = x^1_n + 2(1-\mu)\left(prox_{\gamma f}(x^2_n) - \frac{x^1_n + x^2_n}{2}\right) \\ x^2_{n+1} = x^2_n + 2(1-\mu)\left(prox_{\gamma g}(x^1_n) - \frac{x^1_n + x^2_n}{2}\right) \end{cases}$ $x_n = \frac{x^1_n + x^2_n}{2}$ converges to $\tilde{x}$
Chambolle Pock	f and g convex $F(x) = \underbrace{f(x)}_{prox_f} + \underbrace{g(Kx)}_{prox_{g^*}}$	K: linear op. $L = \vert\vert\vert K \vert\vert\vert$ $\sigma\tau L^2 < 1$	$\begin{cases} y_{n+1} = prox_{\sigma g^*}(y_n + \sigma K \overline{x_n}) \\ z_{n+1} = prox_{\tau f}(x_n - \tau K^* y_{n+1}) \\ \overline{x_{n+1}} = 2x_{n+1} - x_n \end{cases}$
Condat	f differentiable ∇f *L-Lipschitz* h convex $F(x) = \underbrace{f(x)}_{grad_f} + \underbrace{h(Kx)}_{prox_{h^*}}$	K: linear op. $\rho \in (0, 1]$ $\sigma, \tau > 0$ $\tau(\frac{L}{2} + \sigma \vert\vert\vert K^* K \vert\vert\vert) < 1$	$\begin{cases} \overline{x_{n+1}} = x_n - \tau \nabla f(x_n) - \tau K^*(u_n) \\ x_{n+1} = \rho \overline{x_{n+1}} + (1-\rho) x_n \\ \overline{u_{n+1}} = prox_{\sigma h^*}(u_n + \sigma K(2\overline{x_{n+1}} - x_n)) \\ u_{n+1} = \rho \overline{u_{n+1}} + (1-\rho) u_n \end{cases}$

Table I: Table with different solvers for the optimization problem 7

Name	**Notation**	**Dual of K**	**Proximal operator of g**
Orthogonal Wavelet	$g(W_{ortho}x) = \lambda \vert\vert W_{ortho}x \vert\vert_1$	W^*_{ortho}: inverse orthogonal wavelet transform	$prox_g$ = soft-threshold
Translation Invariant Wavelet	$g(W_{TI}x) = \lambda \vert\vert W_{TI} \vert\vert_1$	W^*_{TI}: inverse TI wavelet transform	$prox_g$ = soft-threshold
Total Variation	$g(\nabla x) = \lambda \vert\vert \nabla x \vert\vert_1$	∇^*=-*div*	$prox_g$ = soft-threshold

Table II: Table with different types of regularizations.