Rapport OPT202 Project 2: Image deblurring

Alexandre LAPRERIE, Antoine GERMAIN

26 March 2024

1 Introduction

1.1 Goal of the project

In this project, we will consider the signal processing problem of deblurring a blurred image. This problem will be formulated as an optimization problem with an l_1 regularization penalty.In order to do so, we will code a forward-backward method with Nesterov's acceleration.

1.2 Mathematical background of deblurring an image

An image can be thought of a a serie of pixels, each pixel having three dimensions. In this project, we consider only one dimension pixels, so each pixel is just a number between 0 and 255.

Let $z \in \mathbb{R}_n$ be the n pixels mapped into a column vector. Blurring causes a linear transformation of the pixels into other pixels that can be written as:

$$z' = Cz, \quad z = W^H x$$

Reconstructing the true image implies solving an inverse problem. Let b be the pixels that we observe of the blurred image, and $CW^H x$ our model of how an healthy image is distorted to account for blur.

Ideally, one would solve the convex least-squares problem

$$x^* \in \underset{x \in \mathbb{R}_n}{\textbf{argmin}} \left(\frac{1}{2} \|CW^H x - b\|_2^2 \right), \quad z^* = W^H x^*$$

As C is mostly empty one can add a regularization term to the cost to favor special properties of the solution. In this project, we will add $\epsilon \|x\|_2^2$ to obtain sparsity property.

2 Mathematical study of the problem

We consider the following problem:

$$x^* \in \underset{x \in \mathbb{R}_n}{\textbf{argmin}} \left(\frac{1}{2} \|CW^H x - b\|_2^2 + \epsilon \|x\|_1 \right), \quad z^* = W^H x^* \tag{1}$$

Let $A = CW^H$ and $f_1(x) = \|Ax - b\|_2^2$, $f_2(x) = \epsilon \|x\|_1$. Then (1) is equivalent to :

$$x^* \in \underset{x \in \mathbb{R}_n}{\operatorname{argmin}} \left(\frac{1}{2} f_1(x) + f_2(x) \right), \quad z^* = W^H x^*$$

This problem is convex as $\mathbb{R}_n$ is and f_1 and f_2 are. Moreover, we can prove that $f_1 \in S_{0,L}^{1,1}$:

The gradient of f_1 is $\nabla f_1(x) = A^T(Ax - b)$

$$\text{So } \|\nabla f_1(x) - \nabla f_1(y)\|_2 = \|A^T(Ax-b) - A^T(Ay-b)\|_2$$
$$= \|A^T A(x-y)\|_2$$
$$\leq \|A^T A\|_2 \|x-y\|_2$$

Therefore $L = \|A^T A\|_2$.
The proximal gradient method iterates the following steps for $k = 0, 1, 2...$:

1. Compute the gradient of f_1 at x^k: $\nabla f_1(x^k)$
2. Perform a gradient step: $v^k = x^k - \alpha \nabla f_1(x^k)$
3. Apply the proximal operator to v^k: $x^{k+1} = \text{prox}_{\alpha f_2}(v^k)$

Given $f_2(x) = \epsilon \|x\|_1$, the proximal operator is defined component wise for each component as:

$$\begin{aligned}
x^{k+1} &= \text{prox}_{\alpha f_2}(v^k) \\
&= \text{prox}_{\alpha \|.\|}(x^k - \alpha \nabla f_1(x^k)) \\
&= \underset{x \in \mathbb{R}_n}{\text{argmin}} \left(\|x\|_1 + \frac{1}{2\alpha x} \|x - x_k\| + \alpha \nabla f_1(x^k) \right)
\end{aligned}$$

$$\iff$$

$$\alpha \partial \|x\| + (x - v_{k+1}) = 0$$

$$\iff$$

$$\alpha \partial |[z]_i| + [z]_i = [v_{k+1}]_i$$

Therefore:

$$[x^{k+1}]_i = [x]_1^* = \text{sign}([v^k]_i) \max(|[v^k]_i| - \alpha\epsilon, 0) = \text{sign}([v^k]_i)(|[v^k]_i| - \alpha\epsilon, 0)_+$$

We can then implement the ISTA and FISTA methods.

3 Resolution of the problem

3.1 Blurring of the image

In this project, we will apply our algorithms on this image:

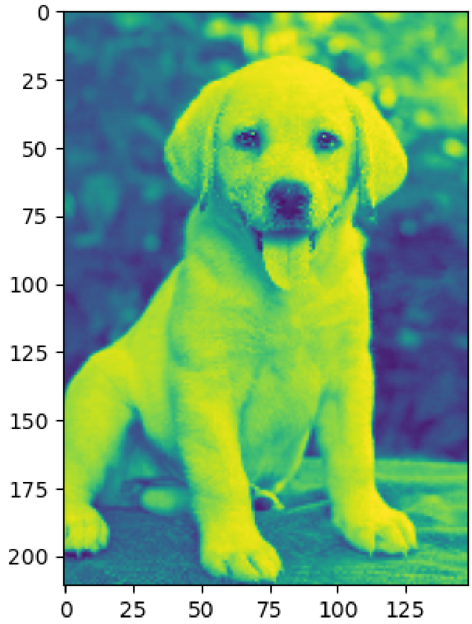
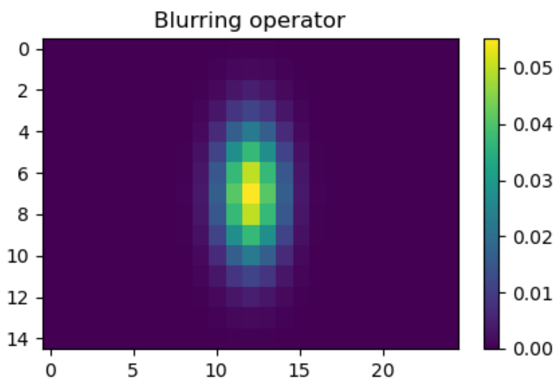
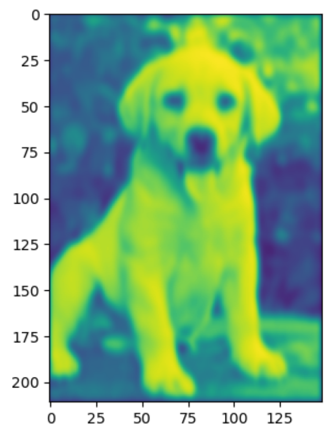


Figure 1: Image used in this project.

Here is the blurring operator used by default in the following blurring processes, and the blurred image resulting:



(a) Blurring operator.



(b) Blurred image.

3.2 First results

With $\alpha = \frac{1}{L}$ and $\epsilon = 0.1$, we obtain this deblurred image:

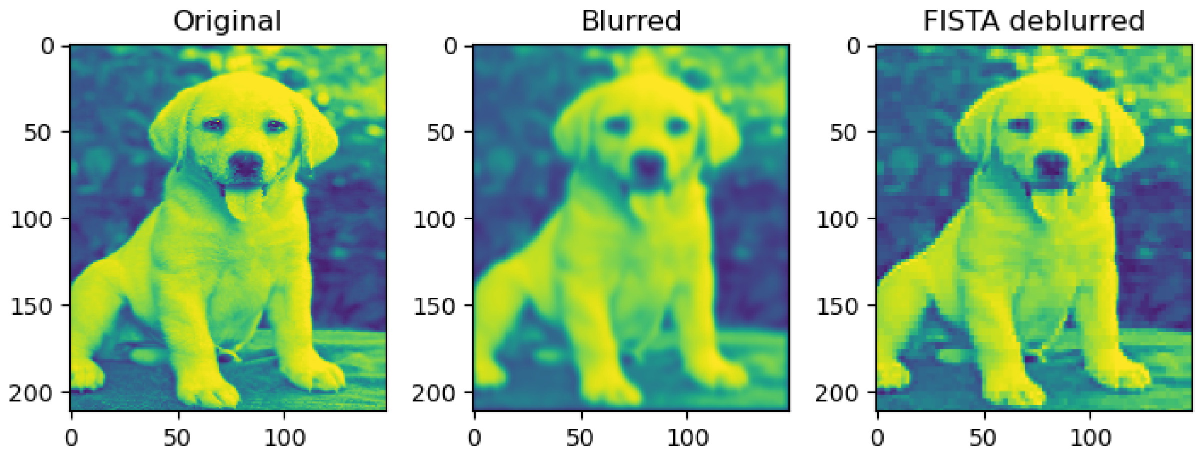


Figure 3: Original, blurred and deblurred image.

The difference in convergence between ISTA and FISTA can then be clearly seen on this graph:

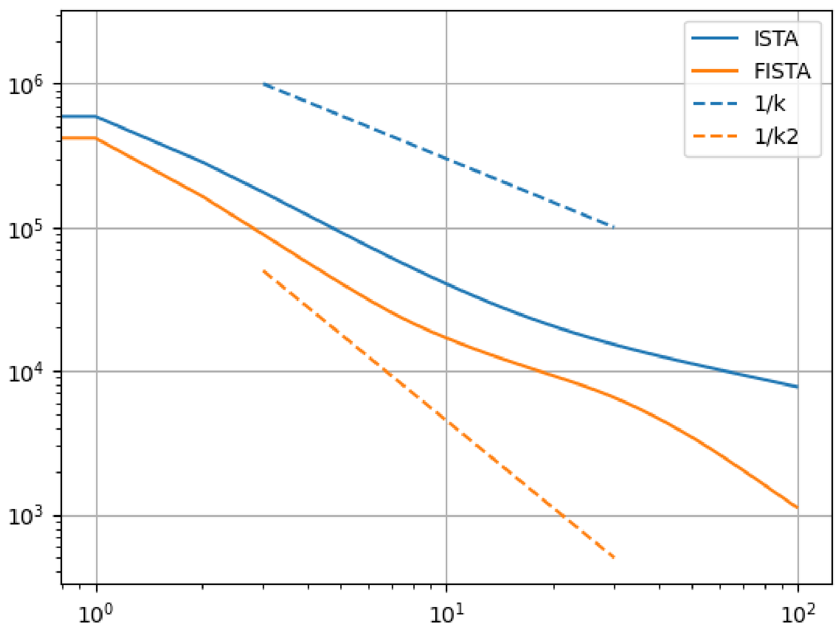


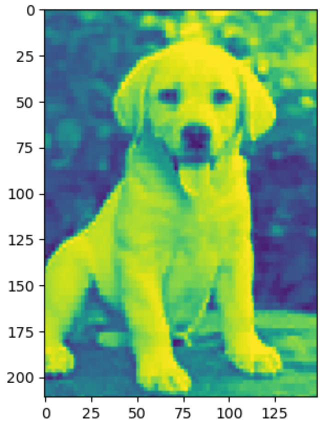
Figure 4: Convergence of ISTA and FISTA.

The rate of convergence is as we could expect by theory.

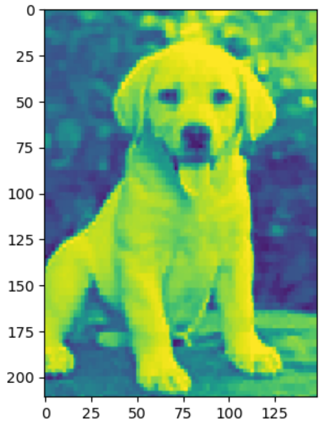
3.3 Parameters impact

3.3.1 Impact of ϵ

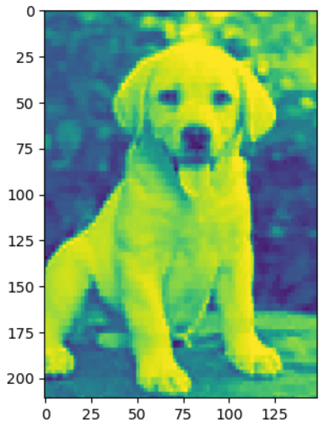
We can try different values for ϵ and see the difference in deblurring, with $\alpha = \frac{1}{L}$ and the default blurring operator:



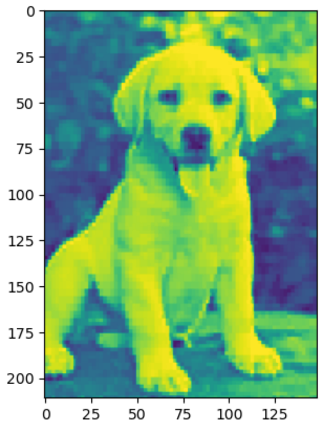
(a) Deblurred image with $\epsilon = 1$.



(b) Deblurred image with $\epsilon = 0.1$.



(a) Deblurred image with $\epsilon = 0.01$.



(b) Deblurred image with $\epsilon = 0.001$.

ϵ does not seem to drastically modify the resulting deblurred image. However, we can see the impact of ϵ when looking at the convergence rates:

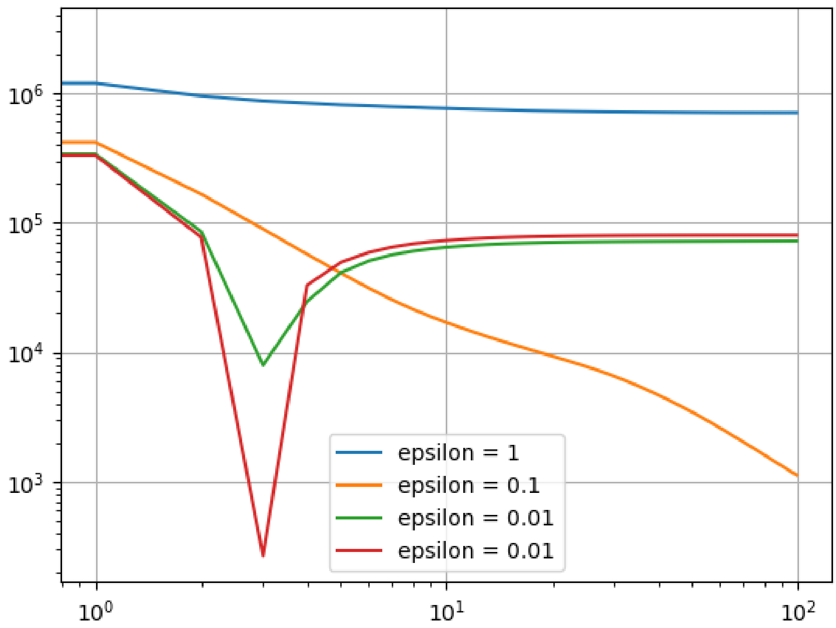
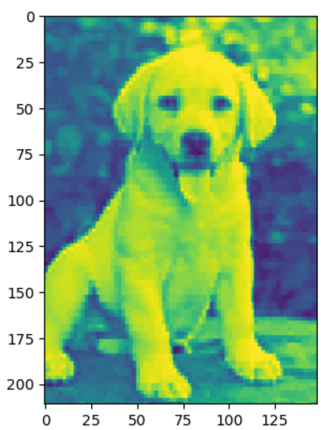


Figure 7: Convergence for FISTA for different values of ϵ.

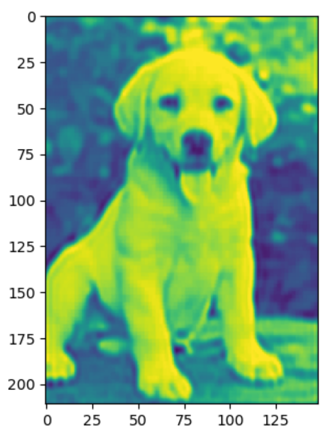
We can conclude that $\epsilon = 0.1$ is the optimal value, which we will continue to use for the next experiments.

3.3.2 Impact of α

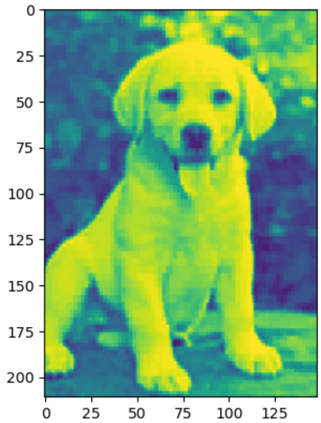
We can also experiment different values for α and see the difference in the deblurred image, with $\epsilon = 0.1$ and the default blurring operator:



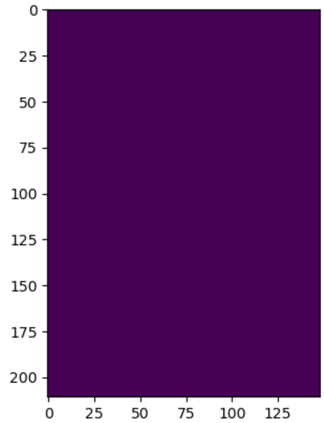
(a) Deblurred image with $\alpha = \frac{1}{L}$.



(b) Deblurred image with $\alpha = \frac{1}{k}$.

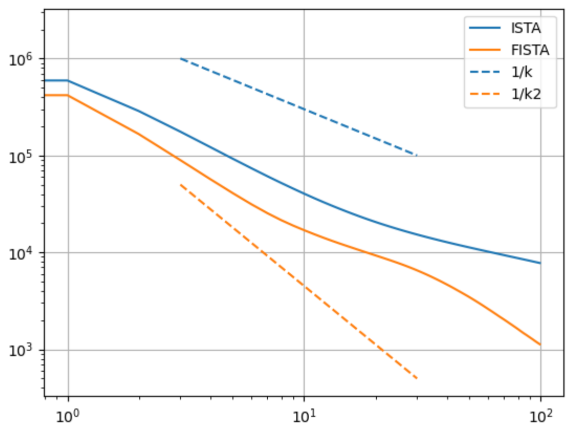


(a) Deblurred image with $\alpha = 0.99^k$.

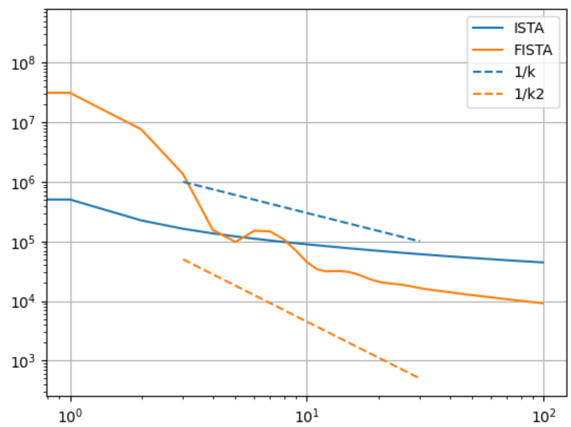


(b) Deblurred image with $\alpha = 1.01^k$.

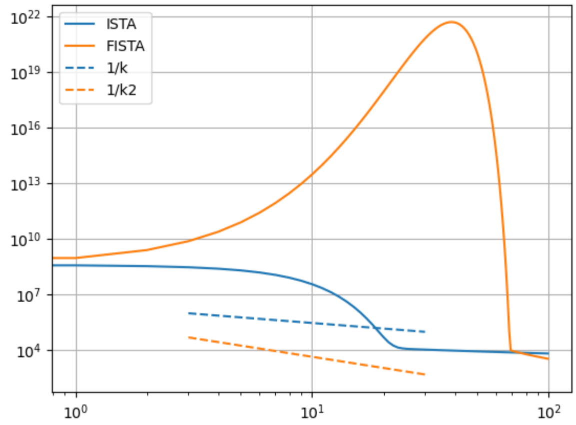
As we could expect, if α diverges to infinity, the deblurred image is not relevant. The result appears better with $\alpha = \frac{1}{k}$, which is coherent with theory. We can then draw the graphs of ISTA/FISTA convergences:



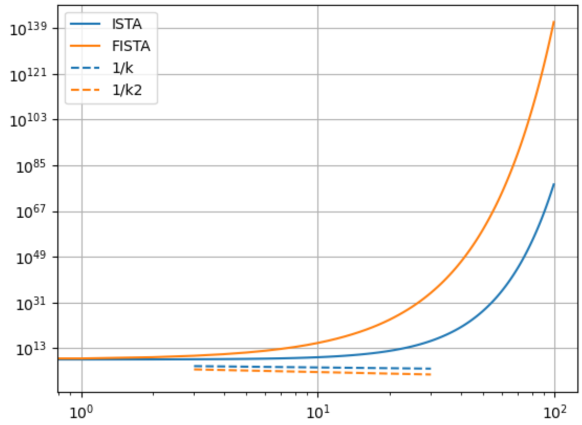
(a) Convergence for ISTA and FISTA with $\alpha = \frac{1}{L}$.



(b) Convergence for ISTA and FISTA with $\alpha = \frac{1}{k}$.



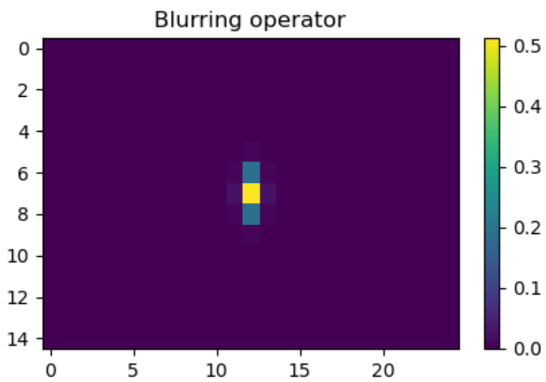
(a) Convergence for ISTA and FISTA with $\alpha = 0.99^k$.



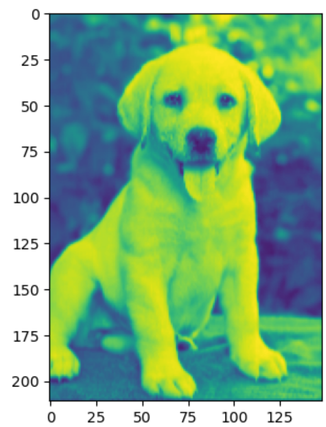
(b) Convergence for ISTA and FISTA with $\alpha = 1.01^k$.

3.3.3 Impact of the blurring operator

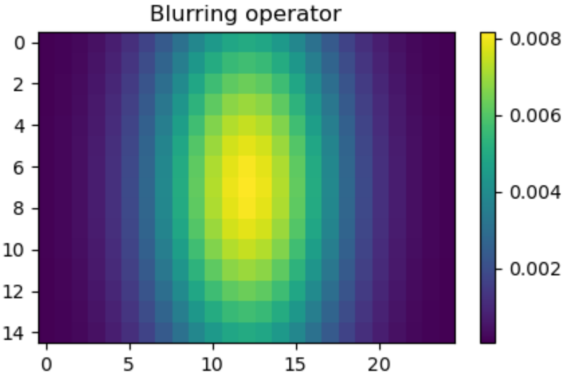
By modifying the blurring operator, we drastically change the results in terms of both deblurred image and convergence. First, we can change the width of the gaussian:



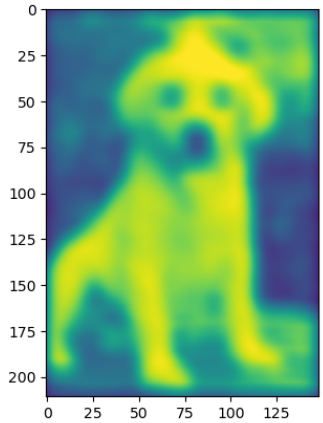
(a) Thinner blurring operator.



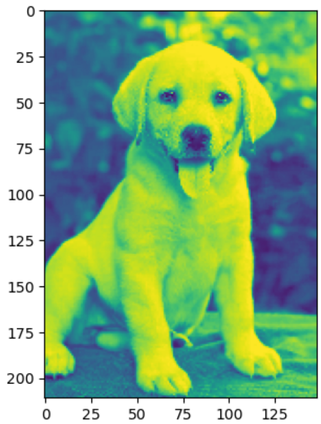
(b) Blurred image.



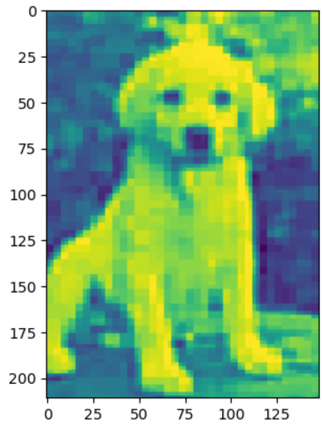
(a) Wider blurring operator.



(b) Blurred image.

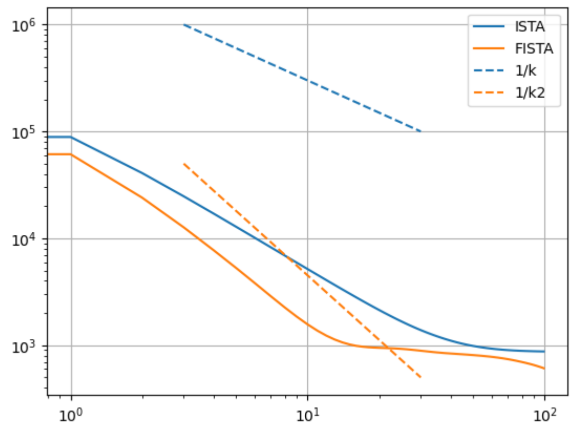


(a) Deblurred image with thin blurring operator.

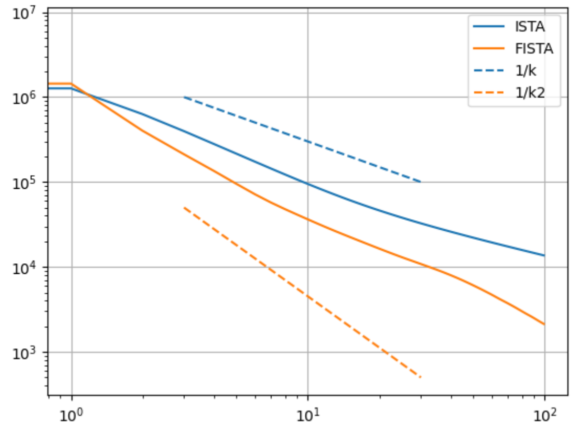


(b) Deblurred image with wide blurring operator.

As one could expect, the thin blurring operator (weighted mainly in the center) slightly modifies the image and result in a very precise deblurred image, and the wide operator (more evenly weighted) results in the opposite. The impact on convergence is also coherent with what we could expect.

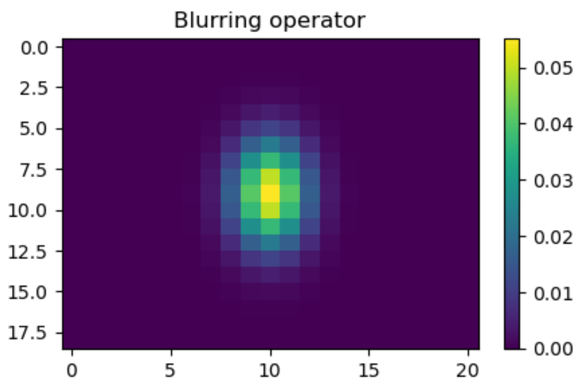


(a) Convergence ISTA/FISTA with thin blurring operator.

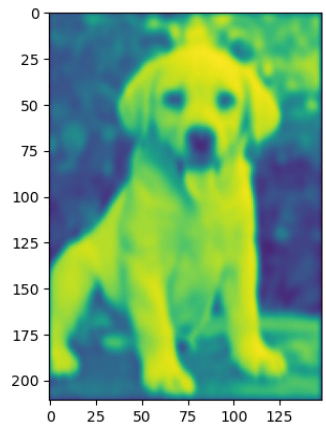


(b) Convergence ISTA/FISTA with wide blurring operator.

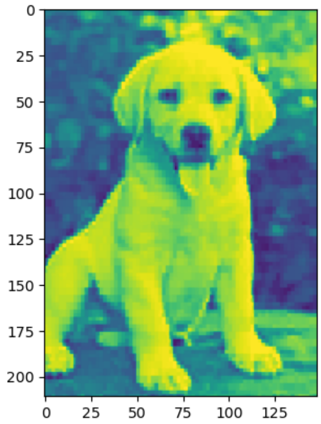
Finally, we can modify the sampling size of the operator by reducing it:



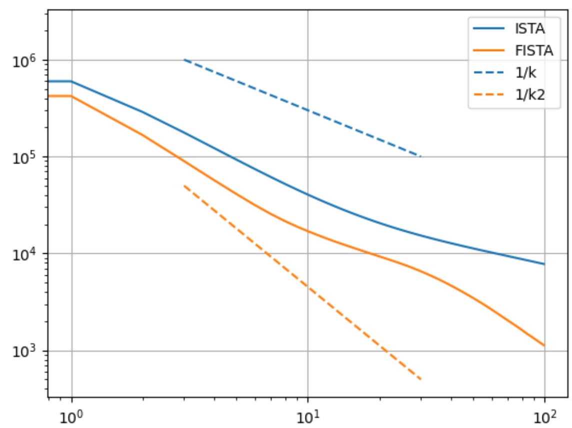
(a) Reduced sampling blurring operator.



(b) Blurred image.



(a) Deblurred image with reduced sampling blurring operator.



(b) Convergence ISTA/FISTA with reduced sampling blurring operator.

This results in no major differences compared to the default settings, other than a differently blurred image. However, the blurring operator being smaller, A has fewer rows (almost three times less, but as much columns) and the computing is thus faster.

4 Conclusion

Overall, the parameters of blurring, gradient step and penalization all have the impact we could expect from them. The ISTA and FISTA methods also act as theoretically planned, particularly in terms of convergence.